Distributing Power-Law Graphs using Labeling Schemes

Casper Petersen, Noy Rotbart,
Jakob Grue Simonsen and Christian Wulff-Nilsen

Department of Computer Science, University of Copenhagen
Universitetsparken 5, 2100 Copenhagen
{cazz,noyro,simonsen,koolooz}@diku.dk

ABSTRACT

A plethora of the graphs underlying social and web networks have been modelled as power-law graphs, and storing power-law graphs, has in turn become a fundamental and well-studied topic. Due to their size, such graphs are not only compressed, but may be stored distributed across several computers.

We study the storage of power-law graphs in a completely distributed manner, using the well-known algorithmic technique of *adjacency labeling schemes*. This theoretical tool assigns labels to the vertices of a graph such that adjacency between two vertices can be inferred using only the information in their corresponding labels. Thus, there is no need for a centralized data structure to hold the graph, and adjacency queries are resolved locally between two vertices. The quality of a labeling scheme is determined by the size of the largest label size it produces, so that each vertex is guaranteed to hold at most that number of bits.

As the number of vertices of high degree is low in power-law graphs, a natural idea is to store the adjacency relation between low and high degree vertices the only in the vertices of low degree. We show that, using a careful selection of the *threshold* between vertices of high and low degree, this strategy alone produces not only good labeling schemes in practice, but that it is theoretically almost optimal. The examination of the labeling scheme in practice is done by an experimental evaluation using both synthetic data and real-world networks with up to 30 million nodes. The theoretical proof is done by proving near matching upper and lower bounds, both for deterministically and probabilistically constructed power-law graphs. Finally, we use the technique to produce a theoretically interesting distance labeling scheme for power-law graphs.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Theory

Keywords

Labeling schemes, Power-law graphs , Distributed data structures

1. INTRODUCTION

A body of work on large, real-world networks deals with the difficulties of storing them and to effectively resolve queries on them; examples of techniques are compression [15, 14] and dissemination the underlying graphs of these networks over several machines [36, 52, **?**]. Another approach is to disseminate the structural information of the graph to its vertices. This *peer-to-peer* strategy allows inferring the graph's local topology using only local information stored in each vertex without using costly access to large, global data structures, and can be particularly useful to address privacy concerns and ensure a high survivability rate [19].

We posit that a useful tool for such a peer-to-peer strategy is the notion of a *labeling scheme*: an algorithm that assigns a bit string–a *label*–to each vertex so that a query between any two vertices can be deduced solely from their respective labels. Labeling schemes are extremely well-studied in the algorithmic literature [39, 33, 18, 31, 41, 42, 40, 21, 28, 49, 8]; the main objective is to minimize the *maximum label size*: the maximum number of bits used in a label of any vertex. Among applications for labeling schemes are XML search engines [27], mapping services [1], and internet routing [44]. Adjacency labeling schemes for numerous important graph families are by now well understood. general graphs require a label size of $n/2 + O(1)$ [47, 8], while trees, planar graphs, and bounded degree graphs enjoy labels of logarithmic size [9, 31, 3].

Routing labeling schemes for power-law graphs have been investigated by Brady and Cowen [18], and by Chen et al. [22]. Labeling schemes for other properties than adjacency have been investigated for various classes of graphs, e.g., distance [33], and flow [39]. Dynamic labeling schemes were studied by Korman and Peleg [41, 42, 40] and recently by Dahlgaard et. al [28]. Experimental evaluation for some labeling schemes for various properties on general graphs have been performed by Caminiti et. al [21], Fischer [30] and Rotbart et. al [49].

One class of graphs extensively used for modelling real-world networks is *power-law graphs*: roughly, n-vertex graphs where the number of vertices of degree k is proportional to n/k^{α} for some positive α. Power-law graphs (also called scale-free graphs in the literature) have been used to model the Internet AS-level graph [50, 4], and many other types of network (see, e.g., [46, 26] for overviews). The adequacy of fit of power-law graph models to actual data, as well as the empirical correctness of the conjectured mechanisms giving rise to power-law behaviour, have been subject to criticism (see, e.g., [2, 26]), but in spite of such criticism, and because their degree distribution affords a reasonable approximation of the degree distribution of many networks, the class of power-law graphs remains a popular tool in network modelling. In this paper, we perform the first theoretical and practical study of adjacency labeling schemes for classes of graphs whose statistical properties–in particular their *degree distribution*– more closely resemble that of real-world networks.

1.1 Our contribution

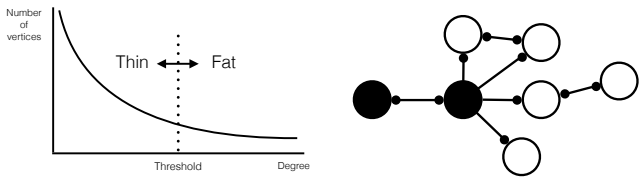
We first define two families of graphs, one that contains and one that is contained by the standard definitions of power-law graphs in the literature. Using those we contribute the following results for the family of power-law graphs:

An $O(\sqrt[\alpha]{n}(\log n)^{1-1/\alpha})$ adjacency labeling scheme. The scheme is based on two ideas: (i) a labeling *strategy* that partitions the vertices of G into high ("fat") and low degree ("thin") vertices based on a threshold degree, and (ii) a threshold *prediction* that depends only on the coefficient α of a power-law curve fitted to the degree distribution of G. Real-world power-law graphs rarely exceed 10^{10} vertices, implying a label size of at most 10^5 bits, well within the processing capabilities of current hardware. Our scheme may be appealing in practice, both due to its simplicity and the reasonable size of its labels. Using the same ideas, we get an asymptotically near-tight $O(\sqrt{n \log n})$ adjacency labeling scheme for sparse graphs.

A lower bound of $\Omega(\sqrt[\alpha]{n})$ for any adjacency labeling scheme. We use our restrictive subclass of power-law graphs and show that it requires label size $\Omega(\sqrt[\alpha]{n})$ for n-vertex graphs. This lower bound shows that our upper bound above is asymptotically optimal, bar a $(\log n)^{1-1/\alpha}$ factor. By the connections between adjacency labeling schemes and universal graphs, we also obtain upper and lower bounds for induced universal graphs for power-law graphs.

An $o(n)$ distance labeling scheme. Using similar strategy to the adjacency labeling scheme, and a small modification, we get this result.

An experimental investigation of our labeling scheme. Using both real-world (23K-3M vertices) and synthetic (300K-1M vertices) data sets, we observe that: (i) Our threshold *prediction* performs close to optimal when using the labeling *strategy* above. (ii) our labeling scheme achieves maximum label size several orders of magnitude smaller than the state-of-the-art labeling schemes for more general graph families.



(a) (b)

Figure 1: Two illustrations of the main idea: Figure (a) demonstrates the threshold assignment, figure (b) demonstrates the label assignment, in which fat (black) nodes do not store adjacency to thin (white) nodes.

2. PRELIMINARIES

Throughout the paper, we consider n-vertex, undirected, finite graphs. For real $c > 0$, a graph is *c-sparse* if it has at most cn edges and *sparse* if it is c-sparse for some constant c. For $0 < c \leq n-1$, the set of c-sparse graphs with n vertices is denoted by $\mathcal{S}_{c,n}$. If $\mathcal{F}$ is a set of graphs, $\mathcal{F}_n$ denotes the subset of graphs in $\mathcal{F}$ having exactly n vertices. The degree of a vertex v in a graph is denoted by $\Delta(v)$, and for non-negative integers k, the set of vertices in a graph G of degree k is denoted by V_k. The length of a binary string $x \in \{0,1\}^*$ is denoted by $|x|$.

Let $\mathcal{F}$ be a set of graphs. An *adjacency labeling scheme* (from hereon just *labeling scheme*) for $\mathcal{G}$ is a pair consisting of an *encoder* and a *decoder*. The encoder is an algorithm that receives $G \in \mathcal{G}$ as input and outputs a bit string $\mathcal{L}(v) \in \{0,1\}^*$ called the *label* of v. The decoder is an algorithm that receives any two labels $\mathcal{L}(v), \mathcal{L}(u)$ as input and outputs `true` iff u and v are adjacent in G and `false` otherwise. Note that the graph G is not an input to the decoder. The *size* of a labeling scheme is the map size : $\mathbb{N} \longrightarrow \mathbb{N}$ such that size(n) is the maximum length of any label assigned by the encoder to any vertex in any graph $G \in \mathcal{F}_n$. The *degree distribution* of a graph $G = (V, E)$ is the mapping $\text{ddist}_G(k) : \mathbb{N}_0 \longrightarrow \mathbb{Q}$ defined by $\text{ddist}_G(k) := \frac{|V_k|}{n}$.

3. POWER-LAW GRAPHS

In the literature *power-law* graphs are usually defined as the class of n vertex graphs G such that $\text{ddist}_G(k)$ is proportional to $k^{-\alpha}$ for some real number $\alpha > 1$. Ideally, and ignoring rounding, $\text{ddist}_G(k) = Ck^{-\alpha}$ for all k for constant C. As the degree distribution of a graph must be a probability distribution, we have $\sum_{k=1}^{\infty} Ck^{-\alpha} = C\sum_{k=1}^{\infty} k^{-\alpha} = 1$, hence $C = 1/\zeta(\alpha)$ where ζ is the Riemann zeta function. However, in the literature, concessions are usually made that relax the restrictions on $\text{ddist}_G(k)$, for example that the power-law property need only hold for high-degree vertices ("above a cutoff"), or that $\text{ddist}_G(k)$ is only *approximately* equal to $Ck^{-\alpha}$, with some approximation error that falls off with n. To ensure that our results hold for all these variations of power-law graphs, we define two families of graphs $\mathcal{P}_h$ and $\mathcal{P}_l$ with $\mathcal{P}_l \subsetneq \mathcal{P}_h$. Family $\mathcal{P}_h$ is rich enough to contain the graphs whose degree distribution is approximately, or perfectly, power-law distributed, and our upper bound on the label size for our labeling scheme holds for any graph

in $\mathcal{P}_h$. Family $\mathcal{P}_l$ is used to show our lower bound and is restrictive enough that most definitions of power-law graph occurring in the literature will contain it.

In the following, let $i_1 = \Theta(\sqrt[\alpha]{n})$ be the smallest integer such that $\lfloor Cn/i_1^\alpha \rfloor \leq 1$, and let $C' \geq (\frac{C}{\alpha-1} + \frac{i_1}{\sqrt[\alpha]{n}} + 5)^\alpha + \frac{C}{\alpha-1}$ be a constant; we shall use C' in the remainder of the paper.

DEFINITION 1. *Let $\alpha > 1$ be a real number and let $\chi : \mathbb{N} \to \mathbb{N}$ be a function. $\mathcal{P}_{h,\chi,\alpha}$ is the family of graphs G such that if $n = |V(G)|$ then for all integers k between $\chi(n)$ and $n-1$, $\sum_{i=k}^{n-1} |V_i| \leq C'(\frac{n}{k^{\alpha-1}})$. We shall usually suppress χ and α, writing merely $\mathcal{P}_h$.*

The number C_2 captures the notion of a cutoff as defined in [26] (Sec. 3.1); the intuition is that the power law distribution need only apply for nodes of degree higher than C_2, rather than for all degrees. Setting $C_2 = 1$ corresponds to the case where the entire range of degrees follows a power-law distribution, hence even for small values of C_2, $\mathcal{P}_h$ morally contains all graphs with power-law degree distribution. We will later prove upper bounds that hold for *all* C_2 bounded by some function; in particular for the upper bound for adjacency labelling schemes, the bound holds for C_2 as high as $\sqrt[\alpha]{n/\log n}$.

The class $\mathcal{P}_l$ contains graphs where the number of vertices of degree k must be $C\frac{n}{k^\alpha}$ rounded either up or down and the number of vertices of degree k is non-increasing with k. Note that the function $k \mapsto C\frac{1}{k^\alpha}$ is strictly decreasing.

DEFINITION 2. *Let $\alpha > 1$ be a real number and let $C = 1/\zeta(\alpha)$ where ζ is the Riemann zeta function. $\mathcal{P}_{l,\alpha}$ is the set of graphs $G = (V, E)$ such that*

1. $\lfloor Cn \rfloor - i_1 - 1 \leq |V_1| \leq \lceil Cn \rceil$,
2. $\lfloor C\frac{n}{2^\alpha} \rfloor \leq |V_2| \leq \lceil C\frac{n}{2^\alpha} \rceil + 1$,
3. *for every i with $3 \leq i \leq n$: $|V_i| \in \{\lfloor C\frac{n}{i^\alpha} \rfloor, \lceil C\frac{n}{i^\alpha} \rceil\}$, and*
4. *for every i with $2 \leq i \leq n-1$: $|V_i| \geq |V_{i+1}|$.*

We usually suppress α, writing just $\mathcal{P}_l$.

Note that we allow slightly more noise in the sizes of V_1 and V_2 than in the remaining sets; without it, it seems tricky to prove a better lower bound than $\Omega(\sqrt[\alpha+1]{n})$ on label sizes.

We show the following properties of $\mathcal{P}_l$.

PROPOSITION 1. *The maximum degree in an n-vertex graph in $\mathcal{P}_l$ is at most $\left(\frac{C}{\alpha-1} + 2\right) \sqrt[\alpha]{n} + i_1 + 3 = \Theta(\sqrt[\alpha]{n})$.*

PROOF. Let $n > 0$ be an integer and let $k' = \lfloor \sqrt[\alpha]{n} \rfloor$. Furthermore, let $S_{k'} = \sum_{i=1}^{k'} |V_i|$, that is $S_{k'}$ is the number of vertices of degree at most k'. Let $S_{k'}^- = (\sum_{i=1}^{k'} \lfloor Cni^{-\alpha} \rfloor) - i_1 - 1$. Then $S_{k'} \geq S_{k'}^-$. We now bound $S_{k'}^-$ from below. For every i with $1 \leq i \leq k'$,

$$\begin{aligned}
S_{k'}^- + k' &= -i_1 - 1 + \sum_{i=1}^{k'} \left(\lfloor Cni^{-\alpha} \rfloor + 1 \right) \geq \\
&\quad - i_1 - 1 + \sum_{i=1}^{k'} Cni^{-\alpha} = -i_1 - 1 + Cn \sum_{i=1}^{k'} i^{-\alpha} \\
&\geq n \left(1 - C \sum_{i=k'+1}^{\infty} i^{-\alpha} \right) - i_1 - 1 \\
&\geq n \left(1 - C \int_{k'}^{\infty} x^{-\alpha} dx \right) - i_1 - 1 \\
&= n \left(1 - C \left[\frac{1}{\alpha - 1} x^{-\alpha+1} \right]_{\infty}^{k'} \right) - i_1 - 1 \\
&= n \left(1 - \frac{C}{\alpha - 1} \left(\lceil \sqrt[\alpha]{n} \rceil \right)^{-\alpha+1} \right) - i_1 - 1 \\
&\geq n \left(1 - \frac{C}{\alpha - 1} \left(\sqrt[\alpha]{n} \right)^{-\alpha+1} \right) - i_1 - 1 \\
&= n - \frac{Cn}{\alpha - 1} n^{-1+\frac{1}{\alpha}} - i_1 - 1 \\
&= n - \frac{C}{\alpha - 1} \sqrt[\alpha]{n} - i_1 - 1,
\end{aligned}$$

giving $S_{k'} \geq S_{k'}^- \geq n - \frac{C}{\alpha-1} \sqrt[\alpha]{n} - \lceil \sqrt[\alpha]{n} \rceil - i_1 - 1$. There are thus at most $\frac{C}{\alpha-1} \sqrt[\alpha]{n} + \lfloor \sqrt[\alpha]{n} \rfloor + i_1 + 1$ vertices of degree strictly more than $k' = \lceil \sqrt[\alpha]{n} \rceil$. Since for every $1 \leq i \leq n-1$: $|V_i| \geq |V_{i+1}|$, it follows that the maximum degree of any graph in $\mathcal{P}_l$ is at most $\left(\frac{C}{\alpha-1} + 2\right) \sqrt[\alpha]{n} + i_1 + 3$. □

PROPOSITION 2. *For $\alpha > 2$, all graphs in $\mathcal{P}_l$ are sparse.*

PROOF. By Proposition 1, the maximum degree of an n-vertex graph in $\mathcal{P}_l$ graph is at most $k' \triangleq \left(\frac{C}{\alpha-1} + 2\right) \sqrt[\alpha]{n} + i_1 + 3$, whence the total number of edges is at most $\frac{1}{2} \sum_{k=1}^{k'} k|V_k|$. By definition, $|V_k| \leq \lceil \frac{Cn}{k^\alpha} \rceil \leq \frac{Cn}{k^\alpha} + 1$ for $k \neq 2$ and $|V_2| \leq \lceil \frac{Cn}{2^\alpha} \rceil + 1$, and thus

$$\begin{aligned}
\frac{1}{2} \sum_{k=1}^{k'} k|V_k| &\leq 1 + \frac{1}{2} \sum_{k=1}^{k'} k \left(\frac{Cn}{k^\alpha} + 1 \right) \\
&\leq 1 + \frac{k'(k'+1)}{4} + Cn \sum_{k=1}^{\infty} k^{-\alpha+1} \\
&= O(n^{2/\alpha}) + Cn\zeta(\alpha - 1) = O(n).
\end{aligned}$$

□

PROPOSITION 3. *For any $C_2 > 0$ and $\alpha > 1$, $\mathcal{P}_{l,\alpha} \subseteq \mathcal{P}_{h,C_2,\alpha}$.*

PROOF. Let $d = \lfloor (\frac{C}{\alpha-1} + 2) \sqrt[\alpha]{n} + i_1 + 3 \rfloor$. For any graph in $\mathcal{P}_l$ with n vertices and for any k, $|V_k| \leq Ck^{-\alpha}n + 1$ and by Proposition 1, $|V_k| = 0$ when $k > d$.

Let k be an arbitrary integer between C_2 and $n-1$. We need to show that $\sum_{i=k}^{n-1} |V_i| \leq C'(\frac{n}{k^{\alpha-1}})$. It suffices to show this

for $k \leq d$. We have:

$$\begin{aligned}
\sum_{i=k}^{n-1} |V_i| &\leq \sum_{i=k}^{d} (Ci^{-\alpha}n + 1) = d - k + 1 + Cn \sum_{i=k}^{d} i^{-\alpha} \\
&\leq \left(\frac{C}{\alpha - 1} + \frac{i_1}{\sqrt[\alpha]{n}} + 5\right) \sqrt[\alpha]{n} + Cn \int_k^d x^{-\alpha} dx \\
&\leq \left(\frac{C}{\alpha - 1} + \frac{i_1}{\sqrt[\alpha]{n}} + 5\right) \sqrt[\alpha]{n} + Cn \left[\frac{1}{\alpha - 1} x^{-\alpha+1}\right]_{\infty}^{k} \\
&\leq \left(\left(\frac{C}{\alpha - 1} + \frac{i_1}{\sqrt[\alpha]{n}} + 5\right)\left(\frac{\sqrt[\alpha]{n} d^{\alpha-1}}{n}\right) + \frac{C}{\alpha - 1}\right) nk^{-\alpha+1} \\
&\leq \left(\frac{C}{\alpha - 1} + \frac{i_1}{\sqrt[\alpha]{n}} + 5\right)\left(\frac{C}{\alpha - 1} + \frac{i_1}{\sqrt[\alpha]{n}} + 5\right)^{\alpha-1} nk^{-\alpha+1} \\
&+ \left(\frac{C}{\alpha - 1}\right) nk^{-\alpha+1} \\
&\leq C' nk^{-\alpha+1},
\end{aligned}$$

as desired. □

3.1 Comparison to other deterministic models

Numerous probabilistic and deterministic definitions of power-law graphs are given in the literature. A recent deterministic model, called shifted power law distribution [29] has recently proven to capture a vast number of such definitions, both in theory and experimentally in [17]. We show that our definition of $\mathcal{P}_h$ contains graphs that adhere to the model, which is defined as follows. Let $c_1 > 0$ be a constant. A graph G is *power law bounded* for parameters $\alpha > 1$ and $t \geq 0$ if for every integer $d \geq 0$, the number of vertices of G of degree in $[2^d, 2^{d+1})$ is at most

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

As experimentally verified in [17], the value of t is typically very small. If $t = O(1)$, the bound above becomes $O(n \sum_{i=2^d}^{2^{d+1}-1} i^{-\alpha})$. In this case, our family $\mathcal{P}_h(C_2, \alpha)$ is rich enough to contain these power law bounded graphs for sufficiently large C' and any choice of C_2 and α. This follows since for any power law bounded graph with n vertices, $|V_i| = O(ni^{-\alpha})$ so for any integer k between 1 and $n-1$, $\sum_{i=k}^{n-1} |V_i| = O(\frac{n}{k^{\alpha-1}})$. It follows that our upper bound also applies to power law bounded graphs. It is possible to extend our upper bound to super-constant t where the bound is stronger the smaller t is; we omit the details. Conversely, our family $\mathcal{P}_l$ is restrictive enough that $\mathcal{P}_l$ is contained in the family of power law bounded graphs when $t = O(1)$, and the lower bound we derive thus also holds in that setting.

4. THE LABELING SCHEMES

We now construct algorithms for labeling schemes for c-sparse graphs and for the family $\mathcal{P}_h$. Both labeling schemes partition vertices into *thin* vertices which are of low degree and *fat* vertices of high degree. The *degree threshold* for the scheme is the lowest possible degree of a fat vertex. We start with c-sparse graphs.

THEOREM 1. *There is a $\sqrt{2cn \log n} + 2 \log n + 1$ labeling scheme for $\mathcal{S}_{c,n}$.*

PROOF. Let $G = (V, E)$ be an n-vertex c-sparse graph. Let $\tau(n)$ be the degree threshold for n-vertex graphs; we choose $\tau(n)$ below. Let k denote the number of fat vertices of G, and assign each fat vertex a unique identifier between 1 and k. Each thin vertex is given a unique identifier between $k+1$ and n.

For a $v \in V$, the first part of the label $\mathcal{L}(v)$ is a single bit indicating whether v is thin or fat followed by a string of $\log n$ bits representing its identifier. If v is thin, the last part of $\mathcal{L}(v)$ is the concatenation of the identifiers of the neighbors of v. If v is fat, the last part of $\mathcal{L}(v)$ is a *fat bit string* of length k where the ith bit is 1 iff v is incident to the (fat) vertex with identifier i.

Decoding a pair $(\mathcal{L}(u), \mathcal{L}(v))$ is now straightforward: if one of the vertices, say u, is thin, u and v are adjacent iff the identifier of v is part of the label of u. If both u and v are fat then they are adjacent iff the ith bit of the fat bit string of $\mathcal{L}(u)$ is 1 where i is the identifier of v. Both decoding processes can be computed in $O(\log n)$ time using standard assumptions.

Since $|E| \leq cn$, we have $k \leq 2cn/\tau(n)$. A fat vertex thus has label size $1 + \log n + k \leq 1 + \log n + 2cn/\tau(n)$ and a thin vertex has label size at most $1 + \log n + \tau(n) \log n$. To minimize the maximum possible label size, we solve $2cn/x = x \log n$. Solving this gives $x = \sqrt{2cn/\log n}$ and setting $\tau(n) = \lceil x \rceil$ gives a label size of at most $1 + \log n + (\sqrt{2cn/\log n} + 1) \log n \leq 1 + 2\log n + \sqrt{2cn \log n}$. □

By Proposition 2, graphs in $\mathcal{P}_l$ are sparse for $\alpha > 2$. This gives a label size of $O(\sqrt{n \log n})$ with the labeling scheme in Theorem 1. We now show that this label can be significantly improved, by constructing a labeling scheme for $\mathcal{P}_h$ which contains $\mathcal{P}_l$.

THEOREM 2. *There is a $\sqrt[\alpha]{C'n}(\log n)^{1-1/\alpha} + 2\log n + 1$ labeling scheme for $\mathcal{P}_h$.*

PROOF. The proof is very similar to that of Theorem 1. We let $\tau(n)$ denote the degree threshold. If we pick $\tau(n) \geq \sqrt[\alpha]{n/\log n}$ then by Definition 1 there are at most $C'n/\tau(n)^{\alpha-1}$ fat vertices. Defining labels in the same way as in Theorem 1 gives a label size for thin vertices of at most $1 + \log n + \tau(n) \log n$ and a label size for fat vertices of at most $1 + \log n + C'n/\tau(n)^{\alpha-1}$. We minimize by solving $x \log n = C'n/x^{\alpha-1}$, giving $x = \sqrt[\alpha]{C'n/\log n}$. Setting $\tau(n) = \lceil x \rceil$ gives a label size of at most $\sqrt[\alpha]{C'n}(\log n)^{1-1/\alpha} + 2\log n + 1$. □

4.1 A labeling scheme for random graphs

There are schemes using randomness to "grow" graphs that, with high probability, have approximate power-law degree distribution for a range of degrees. One such scheme is the preferential attachment model [13]. For graphs obtained from such models, their degree sequences are instead probability distributions. We now show that applying our labeling scheme for $\mathcal{P}_h$ to random graphs with the power law distribution results in a small expected worst-case label size.

Using the definition of Mitzenmacher [46], a random variable X is said to have the *power law* distribution (w.r.t. $\alpha > 1$) if

$$Pr[X \geq x] \sim cx^{-\alpha+1},$$

for a constant $c > 0$, i.e., $\lim_{x\to\infty} Pr[X \geq x]/cx^{-\alpha+1} = 1$.

Let $\epsilon > 0$ be fixed. Consider a graph G picked from a family $\mathcal{F}$ of random graphs whose degree sequences have the power law distribution. Order the vertices of G arbitrarily as $v_1, \ldots, v_n$. For $i = 1, \ldots, n$, let indicator variable X_i be 1 iff v_i has degree at least $d = \sqrt[\alpha]{n/\log n}$. There is a constant $N_0 \in \mathbb{N}$ (depending on ϵ) such that if $n \geq N_0$ then for all i,

$$E[X_i] = \Pr[X_i = 1] \leq (1+\epsilon)cd^{-\alpha+1}.$$

With the same labeling scheme as for $\mathcal{P}_h$ with degree threshold $\tau(n) = d$, denote by E_n the expected label size of an n-vertex graph from $\mathcal{F}$. Then for all $n \geq N_0$,

$$\begin{aligned} E_n &= \sum_{x=0}^{n} \Pr\left[\sum_{i=1}^{n} X_i = x\right] O((x + d\log n)) \\ &= O\left(d\log n + E\left[\sum_{i=1}^{n} X_i\right]\right) \\ &= O\left(d\log n + \sum_{i=1}^{n} E[X_i]\right) \\ &= O\left(d\log n + nd^{-\alpha+1}\right) \\ &= O\left(\sqrt[\alpha]{n}(\log n)^{1-1/\alpha}\right). \end{aligned}$$

THEOREM 3. *Let $\mathcal{F}$ be a family of graphs with degree sequences having the power law distribution w.r.t. $\alpha > 1$. Then there is a labeling scheme for $\mathcal{F}$ such that the expected worst-case label size of any graph $G \in \mathcal{F}$ is $O(\sqrt[\alpha]{n}(\log n)^{1-1/\alpha})$ where n is the number of vertices of G.*

Some generative models use randomness to, roughly, grow a graph with approximate power-law distribution for a range of degrees, but with no theoretical guarantee that the result will be a random graph as above. The Barabási-Albert (BA) model is the most well-known such generative model; the BA model, roughly, grows a graph in a sequence of time steps by inserting a single vertex at each step and attaching it to $\mathcal{E}$ existing vertices with probability weighted by the degree of each existing vertex [13]. The BA model generates graphs that asymptotically have a power-law degree distribution ($\alpha = 3$) for low-degree nodes [16]. However, graphs created by the BA model have low arboricity [1] [35]; we use that fact to devise the following highly efficient labeling scheme for any graph generated by the BA model.

PROPOSITION 4. *The family of graphs generated by the BA model has an $O(\mathcal{E}\log n)$ adjacency labeling scheme.*

PROOF. Let $G = (V, E)$ be an n-vertex graph resulting by the construction by the BA model with some parameter m (starting from some graph $G_0 = (V_0, E_0)$ with $|V_0| \ll n$). While it is not known how to compute the arboricity of a graph efficiently, it is possible in near-linear time to compute a partition of G with at most twice[2] the number of forests in comparison to the optimal [10]. We can thus decompose the graph to $2\mathcal{E}$ forests in near linear time and label each forest using the recent $\log n + O(1)$ labeling scheme for trees [6], and achieve a $2\mathcal{E}(\log n + O(1))$ labeling scheme for G. □

[1]the arboricity of a graph is the minimum number of spanning forests needed to cover its edges.

If the encoder operates at the same time as the creation of the graph, Proposition 4 can be strengthened to yield an $m\log n$ labeling scheme, by storing the identifiers of the ϵ vertices to the node introduced. Theorem 4 and Proposition 4 strongly suggest that local properties of power-law graphs are very different from those of a randomly generated graph using the BA model. In contrast, other generative models such as Waxman's [54], N-level Hierarchical [20]. and Chung and Liu's [24] (Chapter 3) do not seem to have an obvious smaller label size than the one in Proposition 2.

5. LOWER BOUNDS

We now derive lower bounds for the label size of any labeling schemes for both $\mathcal{S}_{c,n}$ and $\mathcal{P}_l$. Our proofs rely on Moon's [47] lower bound of $\lfloor n/2 \rfloor$ bits for labeling scheme for general graphs. We first show that the upper bound achieved for sparse graphs is close to the best possible. The following proposition is essentially a more precise version of the lower bound suggested by Spinrad [51].

PROPOSITION 5. *Any labeling scheme for $\mathcal{S}_{c,n}$ requires labels of size at least $\left\lfloor \frac{\sqrt{cn}}{2} \right\rfloor$ bits.*

PROOF. Assume for contradiction that there exists a labeling scheme assigning labels of size strictly less than $\lfloor \frac{\sqrt{cn}}{2} \rfloor$. Let G be an n-vertex graph. Let G' be the graph resulting by adding $\left\lfloor \frac{n^2}{c} \right\rfloor - n$ isolated vertices to G, and note that now G' is c-sparse. The graph G is an induced subgraph of G'. It now follows that the vertices of G have labels of size strictly less than $\left\lfloor \frac{\sqrt{c\lfloor n^2/c \rfloor}}{2} \right\rfloor \leq n/2$ bits. As G was arbitrary, we obtain a contradiction. □

5.1 Lower bound for power-law graphs

In the remainder of this section we are assuming that $\alpha > 2$ and prove the following:

THEOREM 4. *For any n, any labeling scheme for n-vertex graphs of $\mathcal{P}_{h,C_2,\alpha}$ requires label size $\Omega(\sqrt[\alpha]{n})$.*

More precisely, we present a lower bound for $\mathcal{P}_l$ which is contained in $\mathcal{P}_h$. Let $n \in \mathbb{N}$ be given and let $H = (V(H), E(H))$ be an arbitrary graph with i_1 vertices where $i_1 = \Theta(\sqrt[\alpha]{n})$ is defined as in Section 3. We show how to construct a graph $G = (V, E)$ in $\mathcal{P}_l$ with n vertices that contains H as an induced subgraph. Observe that a labeling of G induces a

[2]More precisely, for any $\epsilon \in (0, 1)$ there exist an $O(|E(G)|/\epsilon)$ algorithm [43] that computes such partition using at most $(1+\epsilon)$ times more forests than the optimal.

labeling of H. As H was chosen arbitrarily and as any labeling scheme for k-vertex graphs requires $\lfloor i_1/2 \rfloor$ label size in the worst case, Theorem 4 follows if we can show the existence of G.

We construct G incrementally where initially $E = \emptyset$. Partition V into subsets $V_1, \ldots, V_n$ as follows. The set V_1 has size $\lfloor Cn \rfloor - i_1$. For $i = 2, \ldots, i_1 - 1$, V_i has size $\lfloor Cn/i^\alpha \rfloor$. Letting $n' = \sum_{i=1}^{i_1-1} |V_i|$, we set the size of V_i to 1 for $i = i_1, \ldots, i_1 + n - n' - 1$ and the size of V_i to 0 for $i = i_1 + n - n', \ldots, n$, thereby ensuring that the sum of sizes of all sets is n. Observe that $\sum_{i=1}^{i_1} \lfloor Cn/i^\alpha \rfloor \leq n$ so that $n' \leq n - i_1$, implying that $n - n' \geq i_1$. Hence we have at least i_1 size 1 subsets $V_{i_1}, \ldots, V_{i_1+n-n'-1}$ in each of which the vertex degree allowed by Definition 2 is at least i_1.

Let $v_1, \ldots, v_{i_1}$ be an ordering of $V(H)$, form a set $V_H \subseteq V$ of i_1 arbitrary vertices from the sets $V_{i_1}, \ldots, V_{i_1+n-n'-1}$, and choose an ordering $v'_1, \ldots, v'_{i_1}$ of V_H. For all $i, j \in \{1, \ldots, i_1\}$, add edge (v'_i, v'_j) to E iff $(v_i, v_j) \in E(H)$. Now, H is an induced subgraph of G and since the maximum degree of H is $i_1 - 1$, no vertex of V_i exceeds the degree bound allowed by Definition 2 for $i = 1, \ldots, n$.

We next add additional edges to G in three phases to ensure that it is an element of $\mathcal{P}_l$ while maintaining the property that H is an induced subgraph of G. For $i = 1, \ldots, n$, during the construction of G we say that a vertex $v \in V_i$ is *unprocessed* if its degree in the current graph G is strictly less than i. If the degree of v is exactly i, v is *processed*.

***Phase* 1.** Let $V' = V \backslash (V_1 \cup V_H)$. Phase 1 is as follows: while there exists a pair of unprocessed vertices $(u, v) \in V' \times V_H$, add (u, v) to E.

When Phase 1 terminates, H is clearly still an induced subgraph of G. Furthermore, all vertices of V_H are processed. To see this, note that the sum of degrees of vertices of V_H when they are all processed is $O(i_1^2) = O(n^{2/\alpha})$ which is $o(n)$ since $\alpha > 2$. Furthermore, prior to Phase 1, each of the $\Theta(n)$ vertices of V' have degree 0 and can thus have their degrees increased by at least 1 before being processed.

***Phase* 2.** While there exists a pair of unprocessed vertices $(u, v) \in V' \times V'$, add (u, v) to E. At termination, at most one vertex of V' remains unprocessed. If such a vertex exists we process it by connecting it to $O(\sqrt[\alpha]{n})$ vertices of V_1; as $|V_1| = \Theta(n)$ there are enough vertices of V_1 to accomodate this. Furthermore, prior to adding these edges, all vertices of V_1 have degree 0, and hence the bound allowed for vertices of this set is not exceeded.

***Phase* 3.** We add edges between pairs of unprocessed vertices of V_1 until no such pair exists. If no unprocessed vertices remain we have the desired graph G. Otherwise, let $w \in V_1$ be the unprocessed vertex of degree 0. We add a single edge from w to another vertex w' of V_1, thereby processing w and moving w' from V_1 to V_2. Note that the sizes of V_1 and V_2 are kept in their allowed ranges due to the first two conditions in Definition 2. This proves Theorem 4.

6. A DISTANCE LABELING SCHEME

In this section we propose a distance labeling scheme for power law graphs.

The *distance* between two nodes in an undirected graph is the length of the shortest path connecting the two nodes, if it exists, and ∞ if no such path exists.

Let $f : \mathbb{N} \longrightarrow \mathbb{N}$ be a map such that $f(n) \leq n - 1$ for all n. An $f(n)$-*distance labelling scheme* is a labelling scheme such that, for any graph G, its decoder given labels $\mathcal{L}(u)$ and $\mathcal{L}(v)$ of two nodes u and v will output the distance between u and v if the distance is at most $f(|V(G)|)$, and output "no" if the distance is strictly greater than $f(|G|)$. If $f(n) = n-1$, an $f(n)$distance labelling scheme is simply called a *distance labelling scheme*.

For sparse graphs, Alstrup et al. [7] obtain a distance labelling scheme with maximum label size $O(\frac{n}{D} \log^2 D)$ where $D = (\log n)/(\log \frac{m+n}{n})$ and m is the number of edges in the graph. Using similar methods, Gawrychowski et al. obtain an upper bond of [34] $O(\frac{n}{D} \log D)$ with sublinear decoding time. Few general results on lower bounds exist. The lower bound of $\Omega(\sqrt{n})$ for adjacency given in the present paper is trivially also a lower bound for distance; for total label size, the best known lower bound remains $\Omega(n^{3/2})$ as proved by Gavoille et al. [32].

We now devise an $f(n)$-distance labelling scheme for $\mathcal{P}_{h,C_2,\alpha}$ working for any $C_2 \geq n^{1/(\alpha-1+f(n))}$. The scheme has shorter labels than any known labelling schemes for small distances. As the class of graphs with power law degree distribution is a subset of $\mathcal{P}_{h,C_2,\alpha}$, and as power-law graphs in general have very small expected distances, the labelling scheme should work well for practical purposes in power-law graphs.

LEMMA 1. *For any computable $f : \mathbb{N} \longrightarrow \mathbb{N}$ such that $f(n) \leq n - 1$ for all n, and for any $C_2 \geq \sqrt[\alpha-1+f(n)]{n}$ there is an $f(n)$-distance labelling scheme for $\mathcal{P}_{h,C_2,\alpha}$ that assigns labels of length at most $O(n^{f(n)/(f(n)+1)} \log f(n))$.*

PROOF. As for adjacency labelling, the scheme is based on *thin* and *fat* nodes. Let G be a graph in $\mathcal{P}_{h,C_2,\alpha}$ Call a node of G *fat* if it has degree at least $n^{1/(\alpha-1+f(n)}$ and *thin* otherwise. The label of each node v now contains (i) a table of distances to all fat nodes (if the distance is more than $f(n)$, it is simply ignored), (ii) a table of distances to all thin nodes w that are at most distance $f(n)$ away from v where the shortest path between v and w does not pass through any fat node, and (iii) a single bit signifiying whether the node is fat or thin. Clearly, as $f(n)$ is computable and distances in G are computable, there is a computable encoder assigning labels. A decoder can now compute the distance between any two nodes u, v as follows: If both u or v are fat, the distance can be directly read off part (i) of the label of any node. If at least one of u and v is fat, the distance can be read off part (i) of the label of the thin node. If both nodes are thin, the decoder can check if the distance is in part (ii) of the label of either node; if the distance is not present, either

the distance is strictly greater than $f(n)$, or the shortest path between u and v passes through a fat node; in this case, the decoder may brute-force check the distances from u and v to each fat node, and simply output the smallest sum of these two distances.

Furthermore, as all nodes of G are either thin or fat, it is clearly possible for an encoder to compute all distances less than or equal to $f(n)$ between any pair of nodes. Note that as all distances we care for are bounded above by $f(n)$, each such distance can be stored using at most $\log f(n)$ bits.

As $G = G(V, E)$ is in $\mathcal{P}_{l,C_2}$, we have

$$\sum_{i=C_2}^{n-1} |V_i| \leq \sum_{i=n^{\frac{1}{\alpha-1+f(n)}}}^{n-1} |V_i| \leq C' \left(\frac{n}{\left(n^{\frac{1}{\alpha-1+f(n)}}\right)^{\alpha-1}} \right)$$
$$\leq C' n^{1-(\alpha-1)/\alpha-1+f(n)} = C' n^{f(n)/(\alpha-1+f(n))}$$

Thus, a table of distances to all fat nodes takes up at most $O\left(n^{\frac{f(n)}{\alpha-1+f(n)}} \log f(n)\right)$ bits.

Similarly, for each node v there are at most $\left(n^{1/(\alpha-1+f(n))}\right)^{f(n)} = n^{f(n)/(\alpha-1+f(n))}$ nodes at distance at most $f(n)$ away from v where the shortest path consists only of thin nodes. Hence, the associated table of distances takes up at most $O(n^{f(n)/(\alpha-1+f(n))} \log n)$ bits.

In total, each label thus has size at most $O(n^{f(n)/(f(n)+1)} \log n)$ bits. □

For $f(n) = \log n$, Lemma 1 yields a labelling scheme having label size at most $O\left(n^{(\log n)/(\alpha-1+\log n)} \log \log n\right)$. Unsurprisingly, as we are only considering distances up to $f(n)$, this label size is asymptotically smaller than for the labelling schemes working for all distances in *sparse* graphs, e.g. the largest label sizes of [34] for sparse graphs is $O(n\frac{\log \log n}{\log n})$. For power law random graphs, Chung and Lu show in [23] that, subject to mild conditions, the diameter of power law graphs with $\alpha > 2$ is almost surely $\Theta(\log n)$. We thus expect our labelling scheme to have superior performance for such graphs.

7. EXPERIMENTAL STUDY

We now perform an experimental evaluation of our labeling scheme on a number of power-law networks. The source code for our experiments can be found at:
`www.diku.dk/\~simonsen/suppmat/podc15/powerlaw.zip`

7.1 Experimental Framework

Recall that our labeling scheme separates the nodes according to a selected threshold from the range $0 \ldots \Delta$,[3] which we select as a function of the power-law parameter α. The following observation is the key to assess our labeling scheme's quality. Suppose we chose a threshold n_0 for a graph G, and call the maximum label size of a thin node, $T(n_0)$ and the maximum label size of a fat node $F(n_0)$. The size of our labeling scheme for the graph G is the larger of these two values. The critical observation is that, as our selection of threshold n_0 increases, $T(n_0)$ monotonically increases and $F(n_0)$ monotonically decreases. Our strategy thus arrives to optimality if we choose n_0 that minimises the value $F(n_0) - T(n_0)$, in other words, where the curves of both functions intersect. In the remainder of this section we call such threshold the *empirical* threshold.

In contrast, we set the threshold in our labeling scheme as $\lceil \sqrt[\alpha]{Cn/(\alpha-1)} \rceil$, which we denote as the *predicted* threshold. It is an approximation to the theoretically optimal threshold choice when degree distributions follow the power-law curve $k \mapsto Cn/k^\alpha$ perfectly, using integration as used in Proposition 3.

Note that a slow encoder can always arrive at the empirical threshold by a binary search which would incur an $O(\log n)$ factor on its running time.

Performance Indicators. We use the following to determine the quality of our labeling scheme: *Performance Indicator i:* We measure the difference in label sizes using the predicted and empirical thresholds. We also measure the difference between the predicted and empirical threshold in percentages with respect to the maximum degree. We interpret a small relative differences in those as an evidence that the predicted threshold can achieve small label sizes without examining the global properties of the network other than the power-law parameter α. This best captures how close our "guess" of the right threshold was, and will, in practice, allow for a faster encoding time.

Performance Indicator ii: We compare the label sizes attained by our labeling schemes to other labeling schemes, namely state-of-the art labeling schemes for the classes of bounded-degree, sparse and general graphs using the labeling schemes suggested in [3], Theorem 1 and [8]. We interpret small label sizes for our scheme, especially in comparison with "small" classes like the class of bounded-degree graphs, as a sign that our labeling scheme efficiently utilizing the extra information about the graphs: namely that their degree distribution is reasonably well-approximated by a power-law.

Test Sets. We employ both real-world and synthetic data sets.

The six *synthetic* data sets are created by first generating a power-law degree sequence using the method of Clauset et al. [26, App. D], subsequently constructing a corresponding graph for the sequence using the Havel-Hakimi method [38]. We use the range $2 < \alpha < 3$ as suggested in [26] as this range of α occurs most commonly in modeling of real-world networks. We generate graphs of $300,000$ and $1M.$ vertices denoted $\text{s300}^{\alpha=x}$ and $\text{s1M}^{\alpha=x}$ respectively, for $x \in \{2.2, 2.4, 2.6, 2.8\}$.

The three *real-world* data sets originate from articles that found the data to be well-approximated by a power-law. The WWW data set [5] contains information on links between

[3]Recall that Δ is the graph's maximum degree.

webpages within the nd.edu domain. The ENRON data set [45] contains email communication between Enron employees (vertices are email addresses; there is a link between two addresses if a mail has been sent between them). The INTERNET data set [48] provides a snapshot the Internet structure at the level of autonomous systems, reconstructed from BGP tables. For all of these sets, we consider the underlying simple, undirected graphs. For each set, standard maximum likelihood methods were used to compute the parameter α of the best-fitting power-law curve [26]. Additional information on the data sets can be found in Table 1.

Real-Life					
Data set	$\|V\|$	$\|E\|$	α	$\Delta_{\max}$	Source
LIVEJOURNAL	3,997,962	34,681,198	2.97	14,815	[?]
WWW	325,729	1,117,563	2.16	10,721	[5]
ENRON	36,692	183,830	1.97	1,383	[45]
INTERNET	22,963	48,436	2.09	2,390	[48]
POKEC	1,632,803	30,622,564	3.5	14854	[53]
ORKUT	3,072,441	117,185,083	2.96	33313	[55]
Synthetic					
$s1M^{\alpha=2.4}$	1,000,000	1,127,797	2.4	42,683	–
$s1M^{\alpha=2.6}$	1,000,000	878,472	2.6	12,169	–
$s1M^{\alpha=2.8}$	1,000,000	751,784	2.8	1,692	–
$s300^{\alpha=2.2}$	300,000	491,926	2.2	10,906	–
$s300^{\alpha=2.4}$	300,000	327,631	2.4	3,265	–
$s300^{\alpha=2.6}$	300,000	261,949	2.6	1,410	–
$s300^{\alpha=2.8}$	300,000	227,247	2.8	1,842	–

Table 1: Data sets and their properties. All graphs are undirected and simple. $\Delta_{\max}$ is the maximum degree of any vertex in the data set.

7.2 Findings

Figure 2 shows the distribution of maximum label sizes for one synthetic and one real-world data set. The maximum label size for the predicted and empirical thresholds as well as upper bounds on the label sizes from different label schemes in the literature can be seen in Table 2 for two synthetic data sets and all three real-world data sets.

Table 2 shows the maximum label sizes achieved using different labeling schemes on our data sets. "Predicted" shows the experimental maximum label size obtained by running our scheme on the graphs, "Empirical" is the label size attained by using the empirical threshold. The remaining columns show non-experimental upper bounds for different label schemes: "Bound" is the upper bound guaranteed in Theorem 2, "c-sparse" is the labeling scheme for sparse graphs defined in Theorem 1, "BD" is the $\lceil \frac{\Delta}{2} \rceil \lceil \log n \rceil$ bounded degree graph labeling of [3], and AKTZ is the $\lceil n/2 \rceil + 6$ general graph labeling of [8]. Both "Empirical" and "Bound" using simple concatenation of labels to represent the fat bit string[4].

Our findings are as follows. For Performance Indicator (i), our labeling scheme obtains maximum label size at most 3.5% larger than what would have been obtained by using the empirical threshold for all synthetic data sets. This is expected—the synthetic data sets are graphs generated specifically to have power-law distributed degree distribution. For the real-world data sets, the labeling scheme obtains maximum label size at most 41.7% larger than by using the empirical threshold; this larger deviation is likely due to degree distributions of the data sets being close to, but not quite, power-law distributions due to natural phenomena or noise. E.g., for the ENRON data set there is sudden drop in frequency between nodes of degree < 158 and ≥ 158.

For Performance Indicator (ii), both our experimental results and theoretical upper bounds for our labeling scheme are several orders of magnitudes lower than for labeling schemes aimed at more general classes of graphs, as expected. Of the more general classes of graphs, it is most interesting to compare the upper bound of bounded degree graphs—the most restrictive class of graphs that both contains the class of power-law graphs and has an efficient labeling scheme described in the literature [3]. As seen in Table 2, the upper bound on our labeling schemes for both power-law graphs and sparse graphs have better upper bounds on label sizes, but only marginally so for data sets with low maximum degree and low values of the power-law parameter α, e.g. ENRON ($\alpha = 1.97$). The actual label sizes obtained in the experiments (the two leftmost columns of Table 2) are substantially lower than the upper bounds, that is, the labeling scheme performs much better in practice than suggested by theory (down to less than a kilobyte per vertex for all data sets). This phenomenon may be due to the degree distribution of the graphs of the data sets having only minor deviation from a power-law for small vertex degrees; our upper bounds on the label size are derived by using the very rich family $\mathcal{P}_h$ that allows very large deviation from a power-law for degrees between 1 and $\sqrt[3]{n/\log n} - 1$.

Finally, note that our labeling scheme supports adjacency for *directed* graphs by using one more bit per edge in each label to store the edge orientation. For data sets whose natural interpretation is as a directed graph (e.g., the WWW set where edges are outgoing and incoming links), the results of Table 2 thus carry over with just one more bit added to the numbers in the two leftmost columns.

8. CONCLUSION AND FUTURE WORK

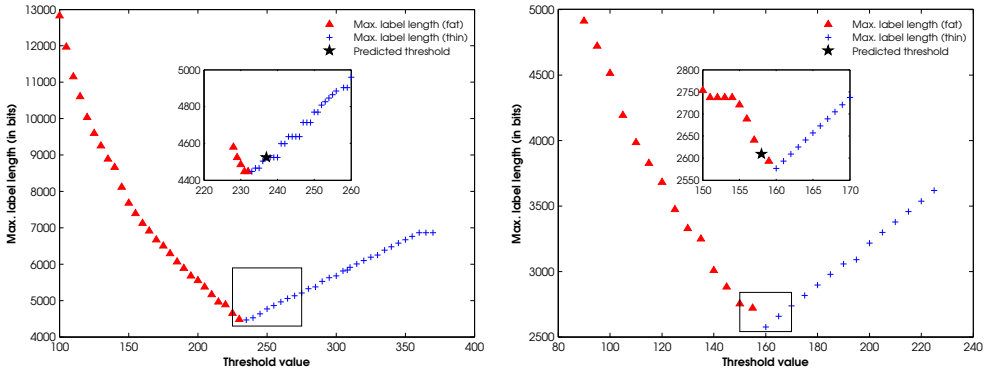
We have devised adjacency and distance labeling schemes for sparse graphs and graphs whose degree distribution approximately follows a power-law distribution. We have proven lower bounds for the class of power-law graphs showing that our strategy for adjacency labeling scheme is almost optimal. Furthermore, we have shown experimentally that the labeling scheme for power-law graphs obtain results in practice requiring little space, and that the theoretical threshold we use in our strategy is reasonably close to the optimum threshold.

8.1 Future work

We propose the following directions:

- Our labeling schemes are designed for static networks, and while it seems not difficult to extend our idea to dynamic networks, an analysis is required to account for the communication and number of re-labels such an extension will incur.

[4]Our labeling schemes introduced in this paper all make use of a succinctly represented "fat bit string"; for our experiments, we use simple concatenation of labels instead of a bit string; this incurs a $(\log n)/\alpha$ factor on the label size, but simplifies the implementation.



(a) syn300$^{\alpha=2.2}$ (b) ENRON

Figure 2: Maximum label sizes of different threshold values for the syn300$^{\alpha=2.2}$ and ENRON data sets. The triangles and crosses represent that for the tested threshold the largest label belong to fat, resp. thin node. The star indicate the position of the predicted threshold.

Data set	Predicted	Empirical	Label Diff.	Threshold Diff.	Upper-Bound	c-sparse	Bounded Degree [3]	AKTZ [8]
s1M$^{\alpha=2.4}$	4,841	4,821	0.4%	0.002%	25,012	30,079	426,820	500,006
s1M$^{\alpha=2.6}$	3,361	3,201	4.8%	0.08%	15,282	26,551	121,680	500,006
s1M$^{\alpha=2.8}$	2,101	2,061	2%	0.17%	10,081	24,566	16,920	500,006
s300$^{\alpha=2.2}$	4,523	4,447	1.7%	0.05%	24,878	18,885	103,607	150,006
s300$^{\alpha=2.4}$	2,775	2,680	3.5%	0.3%	14,404	15,420	31,008	150,006
s300$^{\alpha=2.6}$	1,958	1,920	3.1%	0.35%	9,151	13,792	13,395	150,006
s300$^{\alpha=2.8}$	1,350	1,312	2.8%	0.1%	6,244	12,849	17,499	150,006
WWW	5,245	3,060	41.7%	1%	29,225	28,445	101,840	162,870
ENRON	2,609	2,577	1.3%	0.2%	15,835	9,735	11,056	18,352
INTERNET	1,426	1,156	19.0%	0.8%	8,181	4,700	17,925	11,487
LIVEJOURNAL	63,866	28,880	221%	0.5%	12,996	183,270	??	1,998,987

Table 2: Label size in bits of labeling schemes. The two leftmost columns are experimental results with an additional difference column following; the Threshold Diff. column is the report of performance indicator iii, and the remaining are upper bounds on label sizes computed from the characteristics of the data sets.

- Labeling schemes for power law graphs can likely be devised for the realistic case where the scheme only has incomplete knowledge of the graph, for example when the expected frequency of vertices of each degree is known, but not the exact frequency of each vertex.
- One way to view labeling schemes as a hole is the extreme scenario where our data structure is disseminated across n machines. This may lead to a better understanding of overheads observed in other distributed graph storage technics.
- Closing the gap of the multiplicative logarithmic factor may be of interest to the theory community. A much more interesting gap is that of distance. As we have seen, there is a large gap between labeling schemes for short distance and adjacency for power-law (and sparse) graphs. This gap effectively deemed the distance labels uninteresting for practical applications.

9. REFERENCES

[1] I. Abraham, D. Delling, A. V. Goldberg, and R. F. Werneck. A hub-based labeling algorithm for shortest paths in road networks. In *Experimental Algorithms*, pages 230–241. Springer, 2011.

[2] D. Achlioptas, A. Clauset, D. Kempe, and C. Moore. On the bias of traceroute sampling: Or, power-law degree distributions in regular graphs. *J. ACM*, 56(4), 2009.

[3] D. Adjiashvili and N. Rotbart. Labeling schemes for bounded degree graphs. In *Automata, Languages, and Programming*, pages 375–386. Springer, 2014.

[4] A. Akella, S. Chawla, A. Kannan, and S. Seshan. Scaling properties of the internet graph. In *Proceedings of the Twenty-Second ACM Symposium on Principles of Distributed Computing, PODC 2003*, pages 337–346, 2003.

[5] R. Albert, H. Jeong, and A.-L. Barabási. Diameter of the world-wide web. *Nature*, 401(6749):130–131, 1999.

[6] S. Alstrup, S. Dahlgaard, and M. B. T. Knudsen. Optimal induced universal graphs and adjacency labeling for trees. In *Proceedings of the 58th Symposium on Foundations of Computer Science*, FOCS '15, Washington, DC, USA, 2015. IEEE Computer Society.

[7] S. Alstrup, S. Dahlgaard, M. B. T. Knudsen, and E. Porat. Sublinear distance labeling for sparse graphs. *CoRR*, abs/1507.02618, 2015.

[8] S. Alstrup, H. Kaplan, M. Thorup, and U. Zwick. Adjacency labeling schemes and induced-universal graphs. *To appear in the 47th symposium on Theory of computing (STOC)*, 2015.

[9] S. Alstrup and T. Rauhe. Small induced-universal graphs and compact implicit graph representations. In *Proceedings of the 43rd Symposium on Foundations of Computer Science*, FOCS '02, pages 53–62, Washington, DC, USA, 2002. IEEE Computer Society.

[10] S. R. Arikati, A. Maheshwari, and C. D. Zaroliagis. Efficient computation of implicit representations of sparse graphs. *Discrete Applied Mathematics*, 78(1):1–16, 1997.

[11] Y. Asano, T. Ito, H. Imai, M. Toyoda, and M. Kitsuregawa. Compact encoding of the web graph exploiting various power laws. In *Advances in Web-Age Information Management*, pages 37–46. Springer, 2003.

[12] Y. Asano, Y. Miyawaki, and T. Nishizeki. Efficient compression of web graphs. In *Computing and Combinatorics*, pages 1–11. Springer, 2008.

[13] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[14] P. Boldi, M. Rosa, M. Santini, and S. Vigna. Layered label propagation: A multiresolution coordinate-free ordering for compressing social networks. In *Proceedings of the 20th international conference on World Wide Web*, pages 587–596. ACM, 2011.

[15] P. Boldi and S. Vigna. The webgraph framework i: compression techniques. In *Proceedings of the 13th international conference on World Wide Web*, pages 595–602. ACM, 2004.

[16] B. Bollobás, O. Riordan, J. Spencer, and G. E. Tusnády. The degree sequence of a scale-free random graph process. *Random Struct. Algorithms*, 18(3):279–290, 2001.

[17] P. Brach, M. Cygan, J. Lacki, and P. Sankowski. Algorithmic complexity of power law networks. In *To appear in Proceedings of the thirty third annual ACM-SIAM symposium on Discrete algorithms*, SODA '16, 2014.

[18] A. Brady and L. J. Cowen. Compact routing on power law graphs with additive stretch. In *ALENEX*, volume 6, pages 119–128. SIAM, 2006.

[19] S. Buchegger, D. Schiöberg, L.-H. Vu, and A. Datta. Peerson: P2p social networking: early experiences and insights. In *Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*, pages 46–52. ACM, 2009.

[20] K. L. Calvert, M. B. Doar, and E. W. Zegura. Modeling internet topology. *Communications Magazine, IEEE*, 35(6):160–163, 1997.

[21] S. Caminiti, I. Finocchi, and R. Petreschi. Engineering tree labeling schemes: A case study on least common ancestors. In *Algorithms-ESA 2008*, pages 234–245. Springer, 2008.

[22] W. Chen, C. Sommer, S.-H. Teng, and Y. Wang. A compact routing scheme and approximate distance oracle for power-law graphs. *ACM Transactions on Algorithms*, 9(1):4, 2012.

[23] F. Chung and L. Lu. The average distance in a random graph with given expected degrees. *Internet Mathematics*, 1(1):91–113, 2004.

[24] F. R. Chung and L. Lu. *Complex Graphs and Networks*, volume 107. American mathematical society Providence, 2006.

[25] F. Claude and G. Navarro. Fast and compact web graph representations. *ACM Transactions on the Web (TWEB)*, 4(4):16, 2010.

[26] A. Clauset, C. R. Shalizi, and M. E. Newman. Power-law distributions in empirical data. *SIAM review*, 51(4):661–703, 2009.

[27] E. Cohen, H. Kaplan, and T. Milo. Labeling dynamic xml trees. *SIAM Journal on Computing*, 39(5):2048–2074, 2010.

[28] S. Dahlgaard, M. B. T. Knudsen, and N. Rotbart. Dynamic and multi-functional labeling schemes. In *Algorithms and Computation*, pages 141–153. Springer, 2014.

[29] Y. Eom, S. Fortunato, and M. Perc. Characterizing and modeling citation dynamics. *PLoS ONE*, 6(9):e24926, 2011.

[30] J. Fischer. Short labels for lowest common ancestors in trees. In *Algorithms-ESA 2009*, pages 752–763. Springer, 2009.

[31] C. Gavoille and A. Labourel. Shorter implicit representation for planar graphs and bounded treewidth graphs. In *Algorithms–ESA 2007*, pages 582–593. Springer, 2007.

[32] C. Gavoille, D. Peleg, S. Pérennes, and R. Raz. Distance labeling in graphs. In *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, SODA '01, pages 210–219, Philadelphia, PA, USA, 2001. Society for Industrial and Applied Mathematics.

[33] C. Gavoille, D. Peleg, S. Pérennesc, and R. Razb. Distance labeling in graphs. *Journal of Algorithms*, 53:85–112, 2004.

[34] P. Gawrychowski, A. Kosowski, and P. Uznanski. Even simpler distance labeling for (sparse) graphs. *CoRR*, abs/1507.06240, 2015.

[35] G. Goel and J. Gustedt. Bounded arboricity to determine the local structure of sparse graphs. In *Graph-Theoretic Concepts in Computer Science*, pages 159–167. Springer, 2006.

[36] J. E. Gonzalez, Y. Low, H. Gu, D. Bickson, and C. Guestrin. Powergraph: Distributed graph-parallel computation on natural graphs. In *OSDI*, volume 12, page 2, 2012.

[37] J.-L. Guillaume, M. Latapy, and L. Viennot. Efficient and simple encodings for the web graph. In *Advances in Web-Age Information Management*, pages 328–337. Springer, 2002.

[38] S. L. Hakimi. On realizability of a set of integers as degrees of the vertices of a linear graph. i. *Journal of the Society for Industrial & Applied Mathematics*, 10(3):496–506, 1962.

[39] M. Katz, N. A. Katz, A. Korman, and D. Peleg. Labeling schemes for flow and connectivity. *SIAM Journal on Computing*, 34(1):23–40, 2004.

[40] A. Korman. General compact labeling schemes for dynamic trees. *Distributed Computing*, 20(3):179–193, 2007.

[41] A. Korman and D. Peleg. Compact separator decompositions in dynamic trees and applications to labeling schemes. In *Distributed Computing*, pages 313–327. Springer, 2007.

[42] A. Korman and D. Peleg. Labeling schemes for weighted dynamic trees. *Inf. Comput.*, 205(12):1721–1740, Dec. 2007.

[43] Ł. Kowalik. Approximation scheme for lowest outdegree orientation and graph density measures. In *Algorithms and computation*, pages 557–566. Springer, 2006.

[44] D. Krioukov, K. Fall, and X. Yang. Compact routing on internet-like graphs. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 1. IEEE, 2004.

[45] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):29–123, 2009.

[46] M. Mitzenmacher. A brief history of generative models for power law and lognormal distributions. *Internet Mathematics*, 1(2):226–251, 2004.

[47] J. Moon. On minimal n-universal graphs. In *Proceedings of the Glasgow Mathematical Association*, volume 7, pages 32–33. Cambridge University Press, 1965.

[48] M. Newman. Network data. `http://www-personal.umich.edu/~mejn/netdata/`, 2013. [Online; accessed 02-Jan-2015].

[49] N. Rotbart, M. V. Salles, and I. Zotos. An evaluation of dynamic labeling schemes for tree networks. In *Experimental Algorithms*, pages 199–210. Springer, 2014.

[50] G. Siganos, M. Faloutsos, P. Faloutsos, and C. Faloutsos. Power laws and the as-level internet topology. *IEEE/ACM Trans. Netw.*, 11(4):514–524, 2003.

[51] J. P. Spinrad. *Efficient graph representations*. American mathematical society, 2003.

[52] I. Stanton and G. Kliot. Streaming graph partitioning for large distributed graphs. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1222–1230. ACM, 2012.

[53] L. Takac and M. Zabovsky. Data analysis in public social networks. In *International Scientific Conference and International Workshop Present Day Trends of Innovations*, pages 1–6, 2012.

[54] B. M. Waxman. Routing of multipoint connections. *Selected Areas in Communications, IEEE Journal on*, 6(9):1617–1622, 1988.

[55] J. Yang and J. Leskovec. Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems*, 42(1):181–213, 2015.