Characterizing Directed and Undirected Networks via Multidimensional Walks with Jumps

FABRICIO MURAI, Universidade Federal de Minas Gerais
BRUNO RIBEIRO, Purdue University
DON TOWSLEY, University of Massachusetts Amherst
PINGHUI WANG, Xi'an Jiaotong University

Estimating distributions of node characteristics (labels) such as number of connections or citizenship of users in a social network via edge and node sampling is a vital part of the study of complex networks. Due to its low cost, sampling via a random walk (RW) has been proposed as an attractive solution to this task. Most RW methods assume either that the network is undirected or that walkers can traverse edges regardless of their direction. Some RW methods have been designed for directed networks where edges coming into a node are not directly observable. In this work, we propose Directed Unbiased Frontier Sampling (DUFS), a sampling method based on a large number of coordinated walkers, each starting from a node chosen uniformly at random. It is applicable to directed networks with invisible incoming edges because it constructs, in real-time, an undirected graph consistent with the walkers trajectories, and due to the use of random jumps which prevent walkers from being trapped. DUFS generalizes previous RW methods and is suited for undirected networks and to directed networks regardless of in-edges visibility. We also propose an improved estimator of node label distributions that combines information from the initial walker locations with subsequent RW observations. We evaluate DUFS, compare it to other RW methods, investigate the impact of its parameters on estimation accuracy and provide practical guidelines for choosing them. In estimating out-degree distributions, DUFS yields significantly better estimates of the head of the distribution than other methods, while matching or exceeding estimation accuracy of the tail. Last, we show that DUFS outperforms uniform node sampling when estimating distributions of node labels of the top 10% largest degree nodes, even when sampling a node uniformly has the same cost as RW steps.

Categories and Subject Descriptors: G.3 [**Mathematics of Computing**]: Probability and Statistics

General Terms: Algorithms, Measurement

Additional Key Words and Phrases: complex networks, directed networks, graph sampling, random walks

ACM Reference Format:
Fabricio Murai, Bruno Ribeiro, Don Towsley and Pinghui Wang, 2017. Characterizing Directed and Undirected Networks via Multidimensional Walks with Jumps. *ACM* V, N, Article A (January YYYY), 34 pages.
DOI:http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

A number of studies [Boccaletti et al. 2006; Eagle et al. 2009; Leskovec and Faloutsos 2006; Leskovec et al. 2008; Mislove et al. 2007; Ribeiro et al. 2010; Rasti et al.

This work is supported by the ARO under MURI W911NF-12-1-0385, the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053 and the CNPq, National Council for Scientific and Technological Development - Brazil.
Author's addresses: F. Murai, Universidade Federal de Minas Gerais; B. Ribeiro, Purdue University; D. Towsley, College of Information and Computer Sciences, University of Massachusetts Amherst; Pinghui Wang, Xi'an Jiaotong University.

DOI:http://dx.doi.org/10.1145/0000000.0000000

2009; Volz and Heckathorn 2008; Gjoka et al. 2010; Kurant et al. 2011b; Chiericetti et al. 2016] are dedicated to the characterization of complex networks. Examples of networks of interest include the Internet, the Web, social, business, and biological networks. Characterizing a network consists of computing or estimating a set of statistics that describe the network. In this work we model a network as a directed or undirected graph with labeled vertices. A label can be, for instance, the degree of a node or, in a social network setting, someone's hometown. Label statistics (e.g., average, distribution) are often used to characterize a network.

Characterizing a network with respect to its labels requires querying vertices and/or edges; associated with each query is a resource cost (time, bandwidth, money). For example, information about web pages must be obtained by querying web servers subject to a maximum query rate. Characterizing a large network by querying the entire network is often too costly. Even if the network is stored on disk it may constitute several terabytes of data. As a result, researchers have turned their attention to the characterization of networks based on incomplete (sampled) data.

Simple strategies such as uniform node and uniform edge sampling possess desirable statistical properties: the former yields unbiased samples of the population and the bias introduced by the latter is easily removed. However, these strategies are often rendered unfeasible because they require either a directory containing the list of all node (edge) ids, or an API that allows uniform sampling from the node (edge) space. Even when the space of possible node (edge) ids is known, its occupancy is usually so low that querying randomly generated ids is expensive. An alternate, cheaper, way to sample a network is via a random walk (RW). A RW samples a network by moving a particle (walker) from a node to a neighboring node. It is applicable to any network where one can query the edges connected to a given node. Furthermore, RWs share some of the desirable properties of uniform edge sampling (i.e., easy bias removal, accurate estimation of characteristics such as the tail of the degree distribution).

On one hand, a great deal of research has focused on the design of sampling methods for *undirected networks* using RWs [Heckathorn 1997; Rasti et al. 2009]. Ribeiro and Towsley proposed Frontier Sampling (FS), a multidimensional random walk that uses n *coupled* random walkers. This method yields more accurate estimates than the standard RW and also outperforms the use of n independent walkers. In the presence of disconnected or loosely connected components, FS is even better suited than the standard RW and independent RWs to sample the tail of the degree distribution of the graph. On the other hand, few works have focused on the development of tools for characterizing *directed networks* in the wild. A network is said to be directed when edges are not necessarily reciprocated. Characterizing directed networks through crawling becomes especially challenging when only outgoing edges from a node are visible (incoming edges are hidden): unless all vertices have a directed path to all other vertices, a walker will eventually be restricted to a (strongly connected) component of the graph. Furthermore, a standard RW incurs a bias that can only be removed by conditioning on the entire graph structure. [Ribeiro et al. 2012] addressed these issues by proposing Directed Unbiased Random Walk (DURW), a sampling technique that builds a virtual undirected graph on-the-fly and performs degree-proportional jumps to obtain asymptotically unbiased estimates of the distribution of node labels on a directed graph.

In this work[1], we propose Directed Unbiased Frontier Sampling (DUFS), a method that generalizes the FS and the DURW algorithms (see Figure 1). Building on ideas in [Ribeiro et al. 2012], we extend FS to allow the characterization of networks regardless of whether they are undirected, directed with observable incoming edges, or di-

[1] Parts of this work are based on previous papers from the authors: [Ribeiro and Towsley 2010] and [Ribeiro et al. 2012].

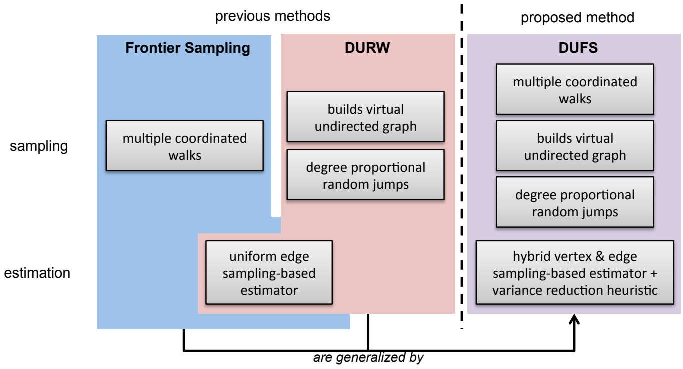
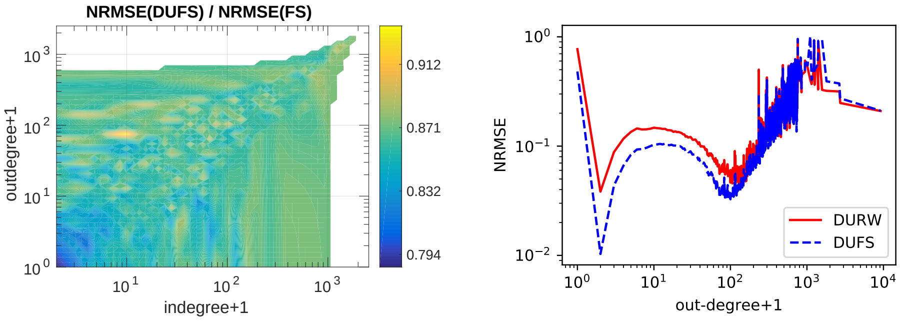

previous methods
proposed method
Frontier Sampling
DURW
DUFS
sampling
estimation
multiple coordinated walks
builds virtual undirected graph
degree proportional random jumps
multiple coordinated walks
builds virtual undirected graph
degree proportional random jumps
uniform edge sampling-based estimator
hybrid vertex & edge sampling-based estimator + variance reduction heuristic
are generalized by

Fig. 1: Proposed method (DUFS) generalizes Frontier Sampling and DURW.



(a) soc-Slashdot0902

(b) livejournal-links

Fig. 2: Comparison between proposed method (DUFS) and previous state-of-the-art respectively for visible and for invisible incoming edges scenarios; (a) NRMSE ratios between DUFS $(w = 0.1, b = 10)$ and FS $(b = 10)$ of the estimated joint in- and out-degree distribution on the soc-Slashdot0902 dataset; (b) NRMSEs associated with DUFS and DURW of the estimated out-degree distribution on the livejournal-links dataset.

rected with unobservable incoming edges. From another perspective, we adapt DURW to use multiple coordinated walkers. DUFS matches or exceeds the accuracy of FS and DURW[2], as illustrated in Figure 2. Method parameters (w and b), simulation setup, datasets and the error metric – NRMSE (normalized root mean square error) – are described in Section 5.1.

Contributions. Our main contributions are as follows:

(1) *Directed Unbiased Frontier Sampling (DUFS)*: we propose a new algorithm based on multiple coordinated random walks that extends Frontier Sampling (FS) to directed networks. DUFS extends DURW to multiple random walks.
(2) *A more accurate estimator for node label distribution*: when the number of walkers is a large fraction of the number of random walk steps (e.g., 10%), a considerable amount of information is thrown out by not accounting for the walkers initial loca-

[2] The software and all results presented in this work are available at http://bitbucket.com/after-acceptance.

tions as observations. We introduce a new estimator that combines these observations with those made during the walks to produce better estimates.

(3) *Practical recommendations*: we investigate the impact of the number of walkers and the probability of jumping to an uniformly chosen node (controlled via a parameter called random jump weight) on DUFS estimation error, given a fixed budget. By increasing the number of walkers the sequence of sampled edges approaches the uniform distribution faster, but this also increases the fraction of the budget spent to place the walkers in their initial locations. Moreover, increasing the random jump weight favors sampling node labels with large probability masses, which translates into more accurate estimates for these labels, but worse estimates for those in the tail. We study these trade-offs through simulation and propose guidelines for choosing DUFS parameters.

(4) *Comprehensive evaluation*: we compare DUFS to other random walk-based methods applied to directed networks w.r.t. estimation errors, both when incoming edges are directly observable and when they are not. In the first scenario, in addition to some graph properties evaluated in previous works, we evaluate DUFS performance on estimating joint in- and out-degree distributions, and on estimating distribution of group memberships among the 10% largest degree nodes.

(5) *Theoretical analysis:* we derive expressions for the normalized mean squared error associated with uniform node and uniform edge sampling on power law networks and show that in both cases error behaves asymptotically as a power law function of the observed degree. This helps explain our evaluation results.

Outline. Definitions are presented in Section 2. In Section 3, we review FS and DURW methods. In Section 4, we propose Directed Unbiased Frontier Sampling (DUFS) (along with some estimators), which generalizes previous methods. We investigate the impact of DUFS parameters on estimation accuracy of degree distributions and node label distributions respectively in Sections 5 and 6, providing practical guidelines on how to set them. A comparison to other random walk techniques is also provided. Section 7 discusses the performance of DUFS when the uniform node sampling mechanism is faulty. We present some related work and present our conclusions in Sections 8 and 9, respectively.

2. TERMINOLOGY SETTING

In what follows we present terminology used throughout the paper. We also present two scenarios considered in our work. Let $G_d = (V, E_d)$ be a labeled directed graph representing the network graph, where V is a set of vertices and E_d is a set of ordered pairs of vertices (u, v) representing a connection from u to v (a.k.a. edges). We refer to an edge (u, v) as an *in-edge* with respect to v and an *out-edge* with respect to u. The *in-degree* and *out-degree* of a node u in G_d are the number of distinct edges respectively into and out of u. We assume that each node in G_d has at least one edge (either an in-edge or an out-edge). Some networks can be modeled as undirected graphs. In this case, G_d is a symmetric directed graph, i.e., $(u, v) \in E_d$ iff $(v, u) \in E_d$.

Let $\mathcal{L}_v$ and $\mathcal{L}_e$ be finite (possibly empty) sets of node labels and edge labels, respectively. Each edge $(u, v) \in E_d$ is associated with a set of labels $\mathcal{L}_e(u, v) \subseteq \mathcal{L}_e$. For instance, one label $\ell \in \mathcal{L}_e(u, v)$ could be the nature of the relationship between two individuals (e.g., family, work, school) in a social network represented by nodes u and v. Similarly, we can associate a set of labels to each node, $\mathcal{L}_v(v) \subseteq \mathcal{L}_v$, $\forall v \in V$.

Input scenarios

When performing a random walk, we assume that a walker retrieves the out-edges of node where it resides by performing a query (e.g., followers list on Twitter) and that vertices are distinguishable. We define two scenarios depending on whether the walker can also retrieve in-edges.

In the *first scenario*, both out- and in-edges can be retrieved and it is possible to move the walker over any edge regardless of the edge direction (if the edge is $(u, v) \in E_d$ a walker can move from u to v and vice versa). In this case, the walker can be seen as moving over $G = (V, E)$, an undirected version of G_d, i.e., $E = \{(u, v) : (u, v) \in E_d \vee (v, u) \in E_d\}$. Define $\deg(v) = |\{(u, v) : (u, v) \in E\}|$. Let vol$(S) = \sum_{\forall v \in S} \deg(v)$, $\forall S \subseteq V$, denote the volume of the set of vertices in $S \subseteq V$.

In the *second scenario*, only out-edges are directly observable and we can build on-the-fly an undirected graph G_u based on the out-edges that have been sampled. Note that G_u is not an undirected version of G_d as some of the in-edges of a node may not have been observed. By moving the walker over G_u – possibly traversing edges in G_d in the opposite direction – we can compute its stationary behavior and thus, remove any bias by accounting for the probability that each observation appears in the sample.

While this has been mostly overlooked by other works, we emphasize that, in either scenario, it is useful to keep track of some variant of the observed graph during the sampling process. Storing information about visited nodes in memory saves resources that would be consumed to query those nodes in subsequent visits – i.e., revisiting a node has no cost. The specific variant of the observed graph to be stored will be described in the context of two random walk-based methods in the following section.

3. BACKGROUND

The method proposed in this paper generalizes two representative random-walk based methods designed for each of the respective scenarios described in Section 2. Therefore, we dedicate this section to briefly reviewing these methods. First, we describe the Frontier Sampling algorithm proposed in [Ribeiro and Towsley 2010], an n-dimensional random walk that benefits from starting its walkers at uniformly sampled vertices. This technique can be applied to undirected graphs and to directed graphs provided that edges coming into a node are observable. Then, we describe the Directed Unbiased Random Walk algorithm proposed in [Ribeiro et al. 2012], that adapts a single random walk to a directed graph when incoming edges are not directly observable. The goal of these methods is to obtain samples from a graph, which are then used to infer graph characteristics via an estimator. An *estimator* is a function that takes a sequence of observations (sampled data) as input and outputs an estimate of an unknown population parameter (graph characteristic).

3.1. Frontier Sampling: a multidimensional random walk for undirected networks

In essence, *Frontier Sampling* (FS) is a random walk-based algorithm for sampling and estimating characteristics of an undirected graph. FS performs n *coordinated* random walks on the graph. One of the advantages of using multiple walkers is that they can cover multiple connected components (when they exist), while a single walker is restricted to one component in the absence of a random jump or restart mechanism. By coordinating multiple random walkers, FS is able to sample edges uniformly at random in steady state regardless of how the walkers are initially placed.

Algorithm 1 describes FS. There are three parameters: the sampling budget B, the initial cost of placing a walker $c \geq 1$ and the average number of nodes b sampled by a walker. The initial walker locations are chosen uniformly at random over the node set (line 2). Note that the number of walkers is taken to be $n = B/(c + b)$, that the cost of a random walk step is one (except for previously sampled nodes) and that the cost of initially placing a walker, c, can be greater than one because uniform node sampling is often expensive. FS keeps a list L of n vertices representing the locations of the n walkers. At each step, a walker is chosen from L in proportion to the degree of the node

ALGORITHM 1: Frontier Sampling (FS)

```
  Input: sampling budget $B$, budget per walker $b$, cost of uniform node sampling $c$
1 $n \leftarrow B/(c+b)$ ;
2 Initialize $L = (v_1, \dots, v_N)$ with $n$ randomly chosen vertices (uniformly);
3 $i \leftarrow N \times c$ {$i$ is the used portion of the budget};
4 while $i < B$ do
5     Select $u \in L$ with probability $\deg(u)/\sum_{\forall v \in L} \deg(v)$ ;
6     Select an edge $(u, v)$, uniformly at random;
7     Replace $u$ by $v$ in $L$ and add $(u, v)$ to sequence of sampled edges;
8     $i \leftarrow i + 1$ {can be skipped if node was previously sampled} ;
9 end
```

where it is currently located (line 5). The walker then moves from u to an adjacent node v (lines 6 and 7).

Frontier sampling is equivalent to the sampling process of a single random walker over the n-th Cartesian power of G. For this reason, Frontier Sampling can be thought of as an n-dimensional random walk (see [Ribeiro and Towsley 2010, Lemma 5.1]).

Using FS samples to estimate node label distributions is simple when the input corresponds to the first scenario described in Section 2. The probability of sampling a given node is proportional to its undirected degree in G. Hence, each sample must be weighted inversely proportional to the respective node's undirected degree. Storing the undirected version of the observed graph along with labels associated with sampled nodes allows the sampler to avoid having to pay the cost of revisiting a node.

Conversely, when incoming edges are not observed, Frontier Sampling can still be adapted to remove bias. We present this method in Section 4.

3.2. Directed Unbiased Random Walk: a random walk adapted for directed networks with unobservable in-edges

The presence of hidden incoming edges but observable outgoing edges makes characterizing large directed graphs through crawling challenging. Edge (u, v) is a hidden incoming edge of node v if (u, v) can only be observed from node u. For instance, in Wikipedia we cannot observe the edge ("Columbia Records", "Thomas Edison") from Thomas Edison's wiki entry (but this edge is observable if we access the Columbia Records's wiki entry).

These hidden incoming edges make it impossible to remove any bias incurred by walking on the observed graph, unless we crawl the entire graph. Moreover, there may not even be a directed path from a given node to all other nodes. Graphs with hidden outgoing edges but observable incoming edges exhibit essentially the same problem. In [Ribeiro et al. 2012], we proposed the Directed Unbiased Random Walk (DURW) algorithm, which obtains asymptotically unbiased estimates of node label densities on a directed graph with unobservable incoming edges. Our random walk algorithm follows two main principles to achieve unbiased samples and reduce variance:

— *Backward edge traversals*: in real-time we construct an undirected graph G_u using nodes that are sampled by the walker on the directed graph G_d. The role of the undirected graph is to guarantee that, at the end of the sampling process, we can approximate the probability of sampling a node, even though in-edges are not observed. The random walk proceeds in such a way that its trajectory on G_d is consistent with that of a random walk on G_u. The walker is allowed to traverse some of the edges in G_d in a reverse direction. However, we prevent some of the observed edges to be traversed in the reverse direction by not including them in G_u. More precisely, once a node z is visited at the i-th step, no in-edges to z observed at step

ALGORITHM 2: Construction of undirected graph (common to DURW and DUFS)

```
   Input: sampling budget $B$, random jump weight $w$, cost of uniform node sampling $c$
1  Select $s \in V$ uniformly at random $\{s = s_1\}$ ;
2  Initialize $\mathcal{S} = \{s\}$ and $E = \mathcal{E}(s)$ ;
3  $i \leftarrow c$ $\{i$ is the used portion of the budget$\}$;
4  while $i < B$ do
5      $p \sim$ Uniform$(0, 1)$ ;
6      if $p \leq w/(w + deg(s))$ then
7          Select $s$ uniformly at random from $V$ {random jump} ;
8          $i \leftarrow i + c$ ;
9      else
10         Select $s$ uniformly at random from $\{v : (s, v) \in E\}$ {random walk step} ;
11         $i \leftarrow i + 1$ ;
12     end
13     if $s \notin \mathcal{S}$ then
14         $\mathcal{S} \leftarrow \mathcal{S} \cup \{s\}$ ;
15         $E \leftarrow E \cup \{(s, v) \in \mathcal{E}(s) : v \notin \mathcal{S}\}$
16     end
17 end
```

$j > i$ (by visiting nodes s such that $(s, z) \in E_d$) are added to G_u. This is an important feature to reduce the random walk transient and thus, reduce estimation errors.

— *Degree-proportional jumps*: the walker makes a limited number of random jumps to guarantee that different parts of the directed graph are explored. In DURW, the probability of randomly jumping out of a node v, $\forall v \in V$, is $w/(w + \deg(v))$, $w > 0$. The steady state probability of visiting a node v on G_u is $(w + \deg(v))/(\text{vol}(V) + w|V|)$. Similar to the cost of placing a FS walker through uniform node sampling, we assume that each random jump incurs cost $c \geq 1$.

The DURW algorithm. DURW is a random walk over a *weighted undirected connected graph* $G_u = (V, E_u)$, which is built on-the-fly. We build an undirected graph using the underlying directed graph G_d and the ability to perform random jumps. Let $G^{(i)} = (V, E^{(i)})$ denote the undirected graph constructed by DURW at step i, where V is the node set and $E^{(i)}$ is the edge set. In what follows we describe the construction of $G^{(i)}$ in Algorithm 2, since this is one of the building blocks of the proposed algorithm, DUFS.

Let $\mathcal{E}(v)$ denote the set of out-edges from a node v in G_d. Let $\mathcal{S}^{(i)} = \{s_1, \ldots, s_i\}$ be the set of nodes from V sampled by the random walk up to step i, where s_j denotes the node on which the walker resides at step j. Since V is not known, we track $G^{(i)}$ using variables $\mathcal{S} = \mathcal{S}^{(i)}$ and $E = E^{(i)}$. The walker starts at node $s_1 \in V$ (line 1). We initialize $G^{(1)} = (V, E^{(1)})$, where $E^{(1)} = \mathcal{E}(s_1)$ (line 2). The next node, s_{i+1}, is selected uniformly at random from V with probability $w/(w + \deg(s_i))$ (lines 6 to 8), where $\deg(s_i)$ is the degree of s_i in $G^{(i)}$. With probability $1 - w/(w + \deg(s_i))$, node s_{i+1} is selected by performing a random walk step from s_i, i.e. by selecting a node adjacent to s_i in $E^{(i)}$ uniformly at random (lines 9 to 12). When node s_{i+1} is visited for the first time, it is necessary to set $\mathcal{S}^{(i+1)}$ to $\mathcal{S}^{(i)} \cup \{s_{i+1}\}$ and $E^{(i+1)}$ to $E^{(i)} \cup \{(s_i, v) \in \mathcal{E}(s_i) : v \notin \mathcal{S}^{(i)}\}$ (lines 13 to 16). By restricting the set of new edges to $\{(s_i, v) \in \mathcal{E}(s_i) : v \notin \mathcal{S}^{(i)}\}$ instead of all edges visible from s_i (i.e., $\mathcal{E}(s_i)$), we comply with the requirement that once a node z, $\forall z \in V$, is visited by the RW, no edge can be added to G_u with z as an endpoint.

In order to estimate node label distributions from DURW observations, we weight samples in proportion to the inverse probability that the corresponding vertices are visited by a random walk in G_u, in steady state. Storing labels and edges associated with nodes in $\mathcal{S}^{(i)}$ saves the cost of querying repeated nodes. Such savings could be reflected in Algorithm 2 by conditioning the increase in i (lines 8 and 11) on $s \notin \mathcal{S}$.

4. GENERALIZING FS AND DURW: A NEW METHOD APPLICABLE REGARDLESS OF IN-EDGE VISIBILITY

This section is divided into two parts. In Section 4.1 we propose Directed Unbiased Frontier Sampling (DUFS), which generalizes FS to allow estimation on directed graphs with unobservable in-edges (second scenario described in Section 2). DUFS also generalizes DURW: the latter is a special case of DUFS where the number of walkers is one. Next, in Section 4.2, we describe two ways to estimate node label distributions using DUFS. The first uses only on the observations collected during the walks. The second estimator we leverages observations obtained from the initial walker locations in addition to observations obtained during the walks.

4.1. Directed Unbiased Frontier Sampling

Like FS, Directed Unbiased Frontier Sampling (DUFS) samples a network through n coordinated walks. At each step, it selects a walker in proportion to the degree of the node where it currently resides. Similar to the Directed Unbiased Random Walk, it constructs an undirected graph in real-time that allows *backward edge traversals*. Denote by $G^{(i)} = (V, E^{(i)})$ the undirected graph constructed by DUFS at step i. DUFS does not include edges in $G^{(i)}$ that would cause walkers to have a view of the graph inconsistent with the view at a previous point in time. In other words, when node u is visited for the first time at step i, u is inserted in $G^{(i)}$ along with all edges $(u, v) \in E_d$ such that v has not been sampled. Thus, the degree of u is fixed in $G^{(j)}$, for all $j \geq i$. Alternatively, letting the degree of u change at a given point would require us to discard the the entire sample up to that point, otherwise the resulting estimator would not be consistent. In fact, even that approach would not yield a consistent estimator for an infinite power law graph: node degrees would never stop changing.

It may seem that there is no need to include *degree-proportional jumps* to visit different graph components when a large number of walkers are initially spread throughout the graph (e.g., on nodes chosen uniformly). However, including degree-proportional jumps in DUFS is extremely beneficial because it prevents walkers from being trapped when initially located on vertices whose out-degree is zero or in components with no outgoing edges. More generally, it allows walkers to move from small volume to large volume components and, hence, obtain more samples among large degree nodes.

Algorithm 3 describes DUFS. In addition to FS' three parameters, it takes a random jump weight w as input. The number of walkers and their initial locations are chosen as in FS (lines 1-3). In the extreme case where $b = 0$, DUFS degenerates to uniform node sampling. When the underlying graph is symmetric and the jump weight is $w = 0$, it becomes FS. When in-edges are invisible and the number of walkers is 1, DUFS degenerates to DURW. We initialize $\mathcal{S} = L$ and $E^{(i)} = \cup_{s \in L} \mathcal{E}(s)$ (line 4). Unlike in FS, a walker is chosen from L in proportion to the sum of the *random jump weight* w and the degree of node where it is currently located *based on* $E^{(i)}$ (line 6). Similar to DURW, the next node is selected based on either a random jump or on following an edge (lines 7-14). Last, the undirected graph is updated (lines 15-18) and so is set L (line 19).

4.2. Estimation

In this section we describe two estimators of node label distributions from samples obtained by DUFS. The first estimator is based on the observations obtained from edges traversed by the random walks. The second estimator combines these observations with those obtained from the walkers initial locations. When used with a variance re-

ALGORITHM 3: Directed Unbiased Frontier Sampling (DUFS)

```
Input: sampling budget $B$, budget per walker $b$, cost of uniform sampling $c$, jump weight $w$
$n \leftarrow B/(c+b)$ {$n$ is the number of walkers};
Initialize $L = \{v_1, \dots, v_N\}$ with $n$ randomly chosen vertices (uniformly);
$i \leftarrow N \times c$ {$i$ is the used portion of the budget};
Initialize $\mathcal{S} = L$ and $E = \cup_{s \in L} \mathcal{E}(s)$ ;
while $i < B$ do
    Select $v \in L$ with probability $(w + \deg(v))/(nw + \sum_{\forall v_j \in L} \deg(v_j))$ ;
    Sample $p \sim \text{Uniform}(0, 1)$;
    if $p < w/(w + \deg(v))$ then
        Select a node $v \in V$ uniformly at random;
        $i \leftarrow i + c$;
    else
        Select an outgoing edge of $v$, $(v, v')$, uniformly at random;
        $i \leftarrow i + 1$;
    end
    if $s \notin \mathcal{S}$ then
        $\mathcal{S} \leftarrow \mathcal{S} \cup \{s\}$ ;
        $E \leftarrow E \cup \{(s, v) \in \mathcal{E}(s) : v \notin \mathcal{S}\}$
    end
    Replace $v$ by $v'$ in $L$ and add $(v, v')$ to sequence of sampled edges;
end
```

duction heuristic, the latter produces better estimates than the former. For a description of estimators of edge label distribution and other graph characteristics, please refer to [Ribeiro and Towsley 2010].

4.2.1. Node Label Distribution: random edge-based estimator. Let s_i denote the i-th node visited by DUFS, $i = 1, \dots, t$, $t \leq B - Nc$. Let θ_ℓ be the fraction of nodes in V with label $\ell \in \mathcal{L}_v$. Let $\pi(v)$ be the steady state probability of sampling node v in G_u, $\forall v \in V$. The node label distribution is estimated at step t as

$$\hat{\theta}_\ell = \frac{1}{n} \sum_{i=1}^{t} \frac{\mathbb{1}\{\ell \in \mathcal{L}_v(v)\}}{\hat{\pi}(s_i)}\,, \qquad \ell \in \mathcal{L}_v,\ t = 1, \dots, B - Nc, \tag{1}$$

where $\mathbb{1}\{P\}$ takes value one if predicate P is true and zero otherwise, and $\hat{\pi}(s_i)$ is an estimate of $\pi(s_i)$: $\hat{\pi}(s_i) = (w + \deg(s_i))S$. Here $\deg(v)$ is the degree of v in $G^{(\infty)}$ and

$$S = \frac{1}{t} \sum_{i=1}^{t} \Bigg(\frac{1}{w + \deg(s_i)}\,. \tag{2}$$

The following theorem states that $\hat{\pi}(s_i)$ is asymptotically unbiased.

THEOREM 4.1. *$\hat{\pi}(s_i)$ is an asymptotically unbiased estimator of $\pi(s_i)$.*

PROOF. To show that $\hat{\pi}(s_i)$ is asymptotically unbiased, we first note that the limit $\lim_{t\to\infty} E^{(t)} = E^{(\infty)}$ exists, since after visiting all vertices we will never add any additional edges. We then invoke Theorem 4.1 of [Ribeiro and Towsley 2010], yielding $\lim_{t\to\infty} S = |V|/(|E^{(\infty)}| + |V|w)$ almost surely. Thus, $\lim_{t\to\infty} \hat{\pi}(s_i) = \pi(s_i)$ almost surely. Taking the expectation of (1) in the limit as $t \to \infty$ yields $E[\lim_{t\to\infty} \hat{\theta}_\ell] = \theta_\ell$, which concludes our proof. □

4.2.2. Node Label Distribution: leveraging information from walkers' initial locations. The estimator presented in (1) does not make use of information associated with the initial set

Variable	Description
n_i	number of node samples with label i
$\theta_{i,j}$	fraction of nodes in $G^{(t)}$ with label i and undirected degree j
$m_{i,j}$	number of edge samples with label i and bias j
$m_i = \sum_j m_{i,j}$	total number of edge samples with label i
$N = \sum_i n_i$	total number of node samples
$M = \sum_i m_i$	total number of edge samples
$B = N + M$	total budget

Table I: Notation used in hybrid estimator.

of nodes on which the walkers are placed. When the number of walkers is large this results in the loss of a considerable amount of statistical information. However, including these observations is challenging because subsequent observations from random walk steps are not independent of the initial observations. Moreover, the normalizing constant for the random walk observations is no longer given by (2), since degree distribution estimates also depend on the information contained in the node samples.

In this section, we derive a new estimator that circumvents these problems by approximating the likelihood of RW samples by that associated with random edge sampling. We call it the *hybrid estimator* because it combines observations from initial walker locations and random walks steps. The hybrid estimator significantly improves the estimation accuracy for labels associated with large probability masses.

Let us index the node labels $\mathcal{L}_v$ from 1 to W, where $W = |\mathcal{L}_v|$. We refer to the sum $\deg(v) + w$ in DUFS as the *random walk bias* for node $v \in V$. To simplify the notation, we assume that each node has exactly one label and that random walk biases take on integer values in $[1, \ldots, Z]$, for some maximum value Z. Denote the node label distribution as $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_W)$. Let n_i denote the number of walkers starting on label i nodes and $m_{i,j}$ the number of subsequent observations of label i and bias j nodes. The notation is summarized in Table I.

We approximate random walk samples in DUFS by uniform edge samples from G_u. Experience from previous studies shows us that this approximation works very well in practice. Hence, the likelihood function given samples $\mathbf{n} = \{n_i : i = 1, \ldots, W\}$ and $\mathbf{m} = \{m_{i,j} : i = 1, \ldots, W$ and $j = 1, \ldots, Z\}$ is expressed as

$$L(\boldsymbol{\theta}|\mathbf{n},\mathbf{m}) = \frac{\prod_i \theta_i^{n_i} \prod_k (k\theta_{i,k})^{m_{i,k}}}{\left(\sum_{s,t} t\theta_{s,t}\right)^M}. \qquad (3)$$

The maximum likelihood estimator $\theta^\star$ is the value of θ that maximizes (3) subject to $0 \le \theta_i \le 1$ and $\sum_i \theta_i = 1$. This defines a constrained non-convex optimization problem. However, we can convert this optimization problem into an unconstrained problem using the reparameterization $\theta_i = e^{\beta_i} / \sum_k e^{\beta_k}$ for $i = 1, \ldots, W$. As shown in Appendix A, the partial derivatives of the resulting objective function are

$$\frac{\partial \mathcal{L}(\boldsymbol{\beta}|\mathbf{n},\mathbf{m})}{\partial \beta_i} = n_i + m_i - \frac{N e^{\beta_i}}{\sum_j e^{\beta_j}} - \frac{M e^{\beta_i} m_i/\mu_i}{\sum_s e^{\beta_s} m_s/\mu_s}, \qquad i = 1, \ldots, W, \qquad (4)$$

where $m_i = \sum_k m_{i,k}$ and $\mu_i = \sum_k m_{i,k}/k$. Setting one of the variables to a constant (say, $\beta_W = 1$) for identifiability and then using the gradient descent method to change the remaining variables according to (4) is guaranteed to converge provided that we make small enough steps. An interesting interpretation of (4) is obtained by setting the derivatives to zero and substituting back $\theta_i = e^{\beta_i} / \sum_k e^{\beta_k}$:

$$\theta_i^\star = (n_i + m_i) \left(N + M \frac{m_i/\mu_i}{\sum_s \theta_s^\star m_s/\mu_s} \right)^{-1}, \qquad i = 1, \ldots, W. \tag{5}$$

According to (5), the estimated fraction of nodes with label i is the total number of times label i was observed (i.e., $n_i + m_i$) normalized by sum of (i) the number of random node samples and (ii) the number of random edge samples weighted by the probability of sampling label i from one random edge sample. In the limit as N and M go to infinity, we can show that $\boldsymbol{\theta}^\star = \boldsymbol{\theta}$ is a solution, but we cannot prove that it is unique or that $\boldsymbol{\theta}^\star$ converges to $\boldsymbol{\theta}$. Hence, we cannot prove that $\boldsymbol{\theta}^\star$ is asymptotically unbiased.

The system of non-linear equations determined by (5) cannot be solved directly, but can be tackled by Expectation Maximization (EM). In this case, the term $\sum_s \theta_s^\star m_s/\mu_s$ in the denominator is replaced by its expected value given θ_i's from the previous iteration. Based on the same idea, if we replace $\sum_s \theta_s^\star m_s/\mu_s$ with an edge sampled-based estimator $\hat{d}$ for the average degree in G_u, we obtain the following non-recursive variant of the hybrid estimator,

$$\hat{\theta}_i = (n_i + m_i) \left(N + M \frac{m_i}{\mu_i \hat{d}} \right)^{-1}, \qquad i = 1, \ldots, W, \tag{6}$$

where $\hat{d} = M/(\sum_i \mu_i)$. Theorem 4.1 below states the conditions under which $\hat{\theta}_i$ is asymptotically unbiased (see appendix for proof). In practice, we find no significant difference between $\theta_i^\star$ and $\hat{\theta}_i$, except when the number of walkers N is very large and the jump weight w is very small. For those cases, $\theta_i^\star$ tends to be slightly more accurate than $\hat{\theta}_i$ for small values of i, which in some applications may justify the additional computational cost of executing gradient descent or EM.

THEOREM 4.1. *Let $N = \alpha B$ and $M = (1 - \alpha)B$, for some $0 < \alpha < 1$. In the limit as $B \to \infty$, the estimator $\hat{\theta}_i$ is an unbiased estimator of θ_i.*

In the special case where the label is the undirected degree itself, we have $\mu_i = m_i/i$. Hence, eq. (6) reduces to

$$\bar{\theta}_i = \frac{n_i + m_i}{N + Mi/\hat{d}}, \tag{7}$$

where $\hat{d}$ is the estimated average degree. When the average degree is known, we can show that $\bar{\theta}_i$ is unbiased and, moreover, the minimum variance unbiased estimator (MVUE) of θ_i (see appendix for proof).

When $n_i > 0$ but $m_i = 0$, the estimator in eq. (6) reduces to $\hat{\theta}_i = n_i/N$, which is essentially the MLE for uniform node sampling. It is well known that this estimator is not nearly as accurate as a random walk based estimator for large out-degree values with small probability mass. In some sense, the estimator $\hat{\theta}_i = n_i/N$ does not account for the fact that the number of random walk samples is zero. As a result, mass estimates for large out-degrees tend to have very large variance when no random walk samples are observed. Fortunately, we find that the following heuristic rule can drastically reduce the estimator variance in these cases.

Variance reduction rule. If no random edge samples are observed for out-degree i, we set the estimate $\hat{\theta}_i = 0$. This implies that we ignore any random node samples seen of nodes that have out-degree i. While this clearly results in a biased estimate, as the budget per walker b goes to infinity, the probability of invoking this rule goes to zero.

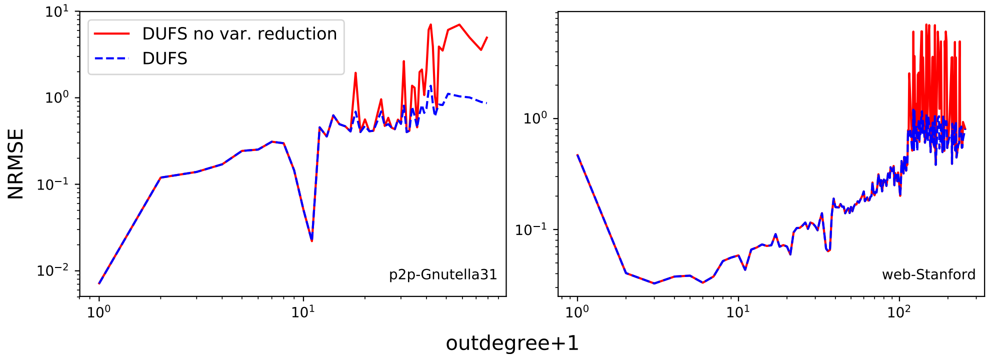


Fig. 3: (visible in-edges) Effect of variance reduction rule on NRMSE, when $B = 0.1|V|$ and $c = 1$. Using information contained in random node samples can increase variance for large out-degree estimates. However, the proposed rule effectively controls for that effect without decreasing head estimates accuracy.

Hence, it produces an asymptotically unbiased estimate. This rule can be interpreted as a combination of node-based and edge-based estimates in proportion to the reciprocals of their estimated variances. That is, when no random edge samples are observed for a given out-degree, the corresponding estimated variance is zero and hence, random node samples should be ignored. We note that the converse rule (i.e., set $\hat{\theta}_i = 0$ if no random node samples were observed) would not perform well, as the probability of sampling large out-degrees with random node sampling is very small.

We simulate DUFS on several datasets and compare the results obtained with the hybrid estimator when the rule is used and when it is not. Simulation details, datasets and the error metric (normalized root mean square error) will be described in Section 5.1. Figure 3 shows representative results of the impact of the rule when estimating out-degree distributions using DUFS in conjunction with the hybrid estimator on two network datasets (averaged over 1000 runs). The results show that the rule consistently reduces estimation error in the distribution tail without affecting estimation quality for small values of i.

In-degree distribution: impossibility result. The fact that long random walks are often approximated by random edge sampling brings up the question of whether they can be used to estimate in-degree distributions when the in-degree is not observed directly. Under random edge sampling, the number of observed edges pointing to a node is binomially distributed and a maximum likelihood estimator can be derived for estimating the in-degree distribution. This problem is related to the set size distribution estimation problem, where elements are randomly sampled from a collection of non-overlapping sets and the goal is to recover the original set size distribution from samples. In addition to in-degree distribution in large graphs, this problem is related to the uncovering of TCP/IP flow size distributions on the Internet.

In [Murai et al. 2013], we derive error bounds for the set size distribution estimation problem from an information-theoretic perspective. The recoverability of original set size distributions presents a sharp threshold with respect to the fraction of elements sampled from the sets. If this fraction lies below the threshold, typically half of the elements in power-law and heavier-than-exponential-tailed distributions, then the original set size distribution is unrecoverable (see [Murai et al. 2013, Theorem 2]).

5. RESULTS ON DEGREE DISTRIBUTION ESTIMATION

Here we focus on the estimation of degree distributions on directed networks. This section is divided into four parts. In Section 5.1, we investigate the impact of DUFS parameters on estimation accuracy. We then compare DUFS against other random walk-based methods when both outgoing and incoming edges are visible in Section 5.2. In Section 5.3, we perform a similar comparison when only out-edges are visible. Last, in Section 5.4 we provide some analysis to explain the relationship observed between the NRMSE and the out-degree (in-degree) in the results. We will refer to the edge-based estimator defined in (1) as E-DUFS.

The 15 directed network datasets in our evaluation were obtained from Stanford's SNAP [Leskovec and Krevl 2014]. These datasets describe the topology of a variety of social networks, communication networks, web graphs, one Internet peer-to-peer networks and one product co-purchasing networks. We found it informative to extract the largest strongly connected component of each directed network and to apply our methods to the resulting datasets – hereby referred to as LCC datasets – as well as to the original datasets. Figure 4 shows the out-degree probability mass function (p.m.f.) for each network, along with the out-degree p.m.f. for the corresponding LCC dataset. We opt to show the p.m.f. instead of the complementary cumulative distribution function (CCDF) because the estimation task in this work is defined in terms of the p.m.f.'s. Defining the estimation task in terms of the CCDF would give DUFS an unfair advantage, as we will explain in Section 5.2.

Simulations consist of sampling the network until a budget $B = 0.1|V|$ (i.e., 10% of the number of vertices) is depleted. Note that budget is decremented when walkers are initially placed and each time one of them moves to a node and when they perform random jumps. We construct an undirected graph in the background throughout each simulation. As a result, we assume that the cost to revisit a node is zero, even if this visit occurs due to a random jump[3].

When both outgoing and incoming edges are observable, random walks disregard edge direction, and move as if the network is undirected. In this scenario, we focus either on the estimation of the marginal out- and in-degree distributions or the joint distribution. The methods we investigate here can be used to estimate other node label distributions. For instance, if the underlying network is undirected, we can estimate the (undirected) degree distribution or even non-topological properties, such as the distribution of user nationalities in a social network. In the light of the impossibility results described in the end of Section 4.2, we focus on out-degree distribution estimation when incoming edges are not directly observable.

Let $\boldsymbol{\theta} = \{\theta_i\}_{\forall i \in \mathcal{L}}$ denote the node label distribution, where θ_ℓ is the fraction of vertices with label ℓ. Denote by $\hat{\theta}_\ell$ the estimate for θ_ℓ. We use normalized root mean square error (NRMSE) of $\hat{\theta}_\ell$ as the error metric, which is a normalized measure of the dispersion of the estimates, defined as

$$\text{NRMSE}(\ell) = \frac{\sqrt{E[(\hat{\theta}_\ell - \theta_\ell)^2]}}{\theta_\ell}. \tag{8}$$

In the case of marginal in-degree (out-degree) distribution, we refer to in-degrees (out-degrees) smaller than the average as the *head* of the distribution. We refer to the largest 1% in- (out-degree) values as the *tail* of the distribution.

[3] Note that the alternative, i.e. always taking c units off the budget per random jump, is unlikely to impact results significantly when $B = 0.1|V|$, since the vast majority of random jumps will find a non-visited node.

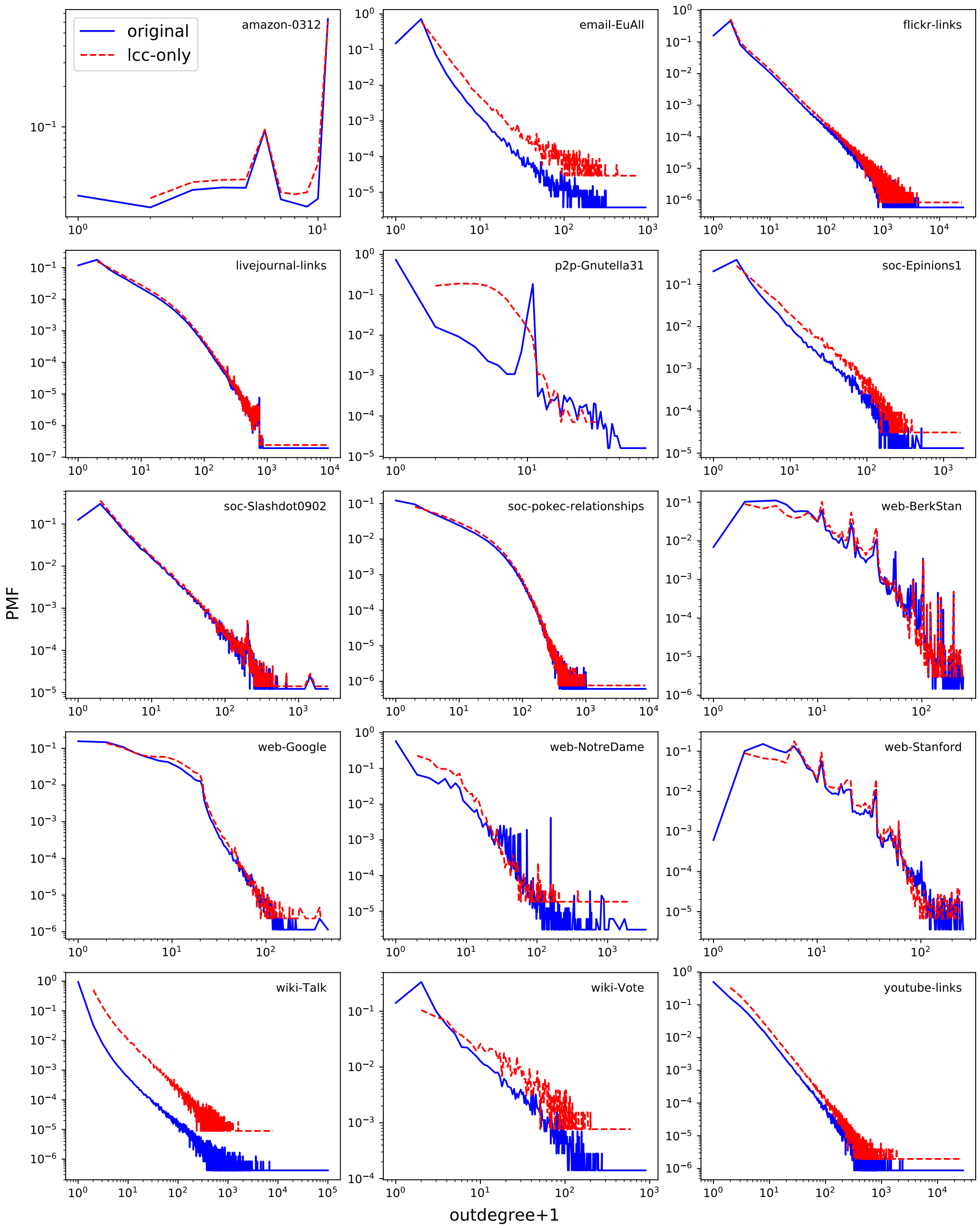


Fig. 4: Out-degree probability mass function (p.m.f.) for each network and its largest strongly connected component (LCC). A large difference between these p.m.f.s suggests it is beneficial to use multiple walkers and/or random jumps.

Table II: Practical guidelines on setting DUFS parameters to obtain accurate head or tail estimates depending on in-edge visibility and node sampling cost c.

	uniform node sampling cost			
	$c = 1$		$c = 10$	
in-edges	visible	not visible	visible	not visible
most accurate for small out-degrees	$w = 10$ $b = 1$	$w = 10$ $b = 1$	$w = 1$ $b = 10^2$	$w = 10$ $b = 1$
most accurate for large out-degrees	$w = 1$ $b = 10$	$w = 1$ $b = 10, 10^2, 10^3$	$w = 0.1$ $b = 10^3$	$w = 0.1$ $b = 10, 10^2, 10^3$

5.1. Impact of DUFS parameters and practical guidelines

To provide intuition on how random jump weight w and budget per walker b affect the accuracy of DUFS estimates, assume for now that we replace samples collected via random walks by uniform edge samples from the weighted undirected graph G_u. In this hypothetical scenario, the budget B is used to collect $N \geq 1$ uniform node samples and $B - Nc$ uniform edge samples. Clearly, when the edge-based estimator defined in (1) is used, the most accurate node label distribution estimates are obtained by setting $N = 1$, (i.e. $b = B - c$). Hence, we focus on the case where the hybrid-estimator defined in (5) is used. In particular, consider estimation of the out-degree distribution.

For a given value of b, the number of uniform node samples will be $B/(c + b)$. For each of the remaining $B - B/(c + b)$ samples, a vertex v is sampled in proportion to $\deg(v) + w$, where $\deg(v)$ is the undirected degree of v in G_u. The choice of w and b impose, individually, a trade-off between estimation accuracy of the head and of tail of the distribution. For a fixed value of w, smaller values of b translate into better estimates of the head (and worse estimates of the tail) because we collect more (less) information about that region of the distribution from uniform node samples. For a fixed value of b, larger values of w also translate into more (less) accurate estimates of the head (tail), because random jumps are more likely to move a node to low in- and out-degree nodes (as they tend to occur more frequently).

In what follows, we observe through simulations that *despite the uniform edge sampling approximation, the previous intuition holds for DUFS head estimates, but not always for tail estimates*. In many cases, as we increase the number of walkers (i.e., decrease b) or increase w, we still obtain good estimates of the tail. This occurs because varying w or b changes the transition probability matrix that governs the sampling process, and thus, the sample distribution.

We simulate DUFS on each original network dataset for combinations of random jump weight $w \in \{0.1, 1, 10\}$ and budget per walker $b \in \{1, 10, 10^2, 10^3\}$ (1000 runs each). For small values of w, DUFS behaves as FS, except for using the improved estimator. For large values of w, DUFS behaves as uniform node sampling. Last, for large values of b, DUFS behaves as DURW. We consider four scenarios that correspond to whether the incoming edges are directly observable or not and to two different costs of uniform node sampling $c = 1$ or $c = 10$. Evaluating these parameter combinations is useful to establish practical guidelines for choosing DUFS parameters, which we summarize in Table II. We observe that estimation accuracy tends to be lower for extreme values of these parameters, suggesting that combinations other than ones investigated here would not provide large accuracy gains (if any).

Visible in-edges, $c = 1$. Figure 5 (all except bottom right) show typical results when varying w and b. To avoid clutter, we show only estimates for powers of two (or the closest out-degree values) and omit results for $b = 10^3$ as they are similar to those for $b = 10^2$. Figure 5 (bottom right) shows similar results for amazon-0312, the dataset

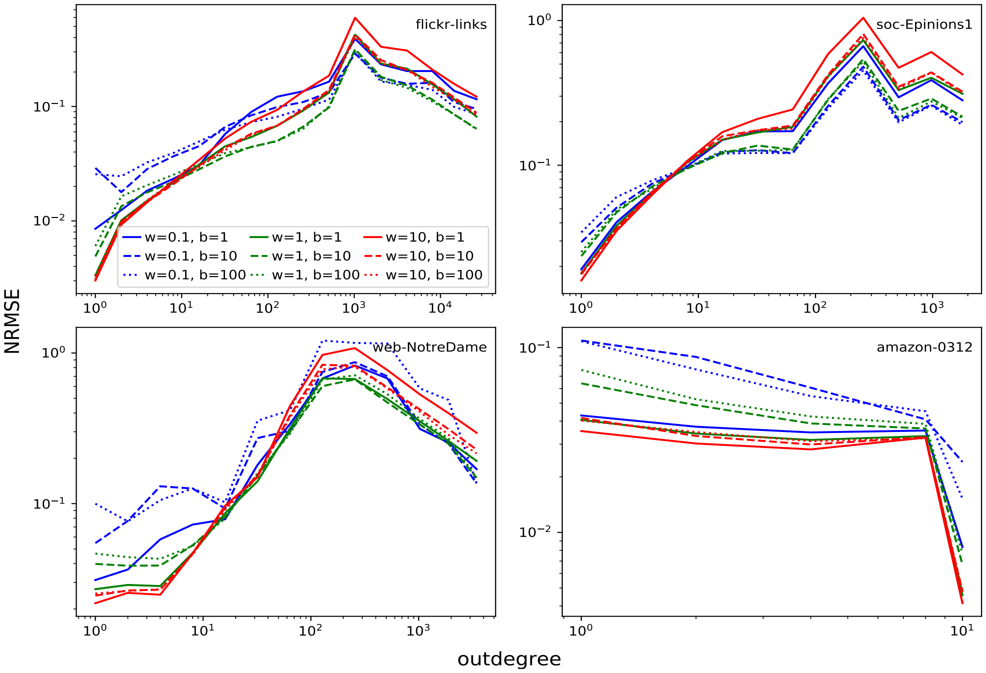


Fig. 5: (visible in-edges, $c = 1$) Effect of DUFS parameters on datasets with many connected components, when $B = 0.1|V|$ and $c = 1$. Legend shows the average budget per walker (b) and jump weight (w). Trade-off shows that configurations that result in many uniform node samples, such as $(w = 10, b = 1)$, yield accurate head estimates, whereas configurations such as $(w = 1, b = 10)$ yield accurate tail estimates.

with the smallest maximum out-degree (max. is 10). Similar to our intuition for uniform edge sampling, the NRMSE associated with the head increases with b and decreases with w, on virtually all datasets[4]. Also as expected, for a fixed values of w, $b = 1$ yields larger errors in the tail than $b \in \{10, 100\}$ (except for amazon-0312). However, contrary to the intuition for uniform edge sampling, $w = 1$ matches or outperforms $w = 0.1$ for (except for $b = 1$). This is best visualized in Figure 5 (bottom right). This happens because setting $w = 1$ allows DUFS to sample regions with large probability mass (in this case, the head) and, at the same time, allows the sampler to move walkers from low volume to high volume components more often than $w = 0.1$. We also observe that $b = 10$ outperforms $b \in \{10^2, 10^3\}$ for $w \in \{0.1, 1\}$. Dataset amazon-0312 is the only dataset where $(w = 10, b = 1)$ obtained the best results over the entire out-degree distribution. As a side note, we observe that for most datasets used here, in log-log scale, the NRMSE grows approximately linearly as a function of the out-degree up to a certain point and then starts to decrease, roughly linearly too. In Section 5.4 we explain why this is the case.

Invisible in-edges, $c = 1$. The results we obtained are similar to those obtained for the visible in-edge scenario, but NRMSEs tend to be larger. Figure 6 shows typical re-

[4] For simplicity, the observations regarding the distribution head (tail) are based on the single smallest (largest) out-degree on each dataset. Similar conclusions are obtained when combining NRMSEs associated with several of the smallest (largest) out-degrees.

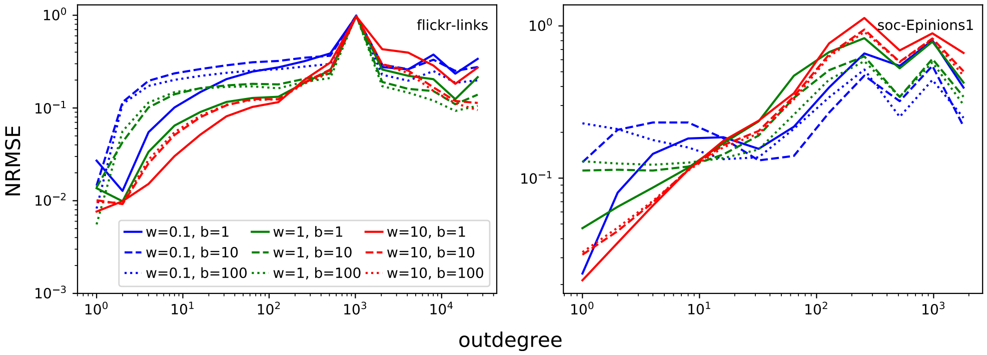


Fig. 6: (invisible in-edges, $c = 1$) Effect of DUFS parameters on datasets with many connected components, when $B = 0.1|V|$ and $c = 1$. Legend shows the average budget per walker (b) and jump weight (w). Configurations that result in many walkers which jump too often, such as $(w \geq 10, b = 1)$ yield accurate head estimates, whereas configurations such as $(w = 1, b = 10^3)$, yield accurate tail estimates.

sults for different DUFS parameters, represented by two datasets (also shown in the previous figure). Once again, the intuition for uniform edge sampling holds for the distribution head: decreasing b and increasing w yield more accurate estimates for the smallest out-degrees. While $b = 1$ results in poor estimates for the largest out-degrees, our intuition regarding w does not hold true for the tail. More precisely, in most cases $w = 1$ outperforms $w = 0.1$ (one exception being dataset soc-Epinions1). As opposed to the visible in-edge scenario, increasing b tends to provide more accurate tail estimates for $w = 1$. We investigate this effect in Section 5.3. We find that, for a fixed w, larger values of b make the random walks jump more often, moving them from small volume components to large volume components, yielding better tail estimates.

Visible in-edges, $c = 10$. Consider the case where the cost of obtaining uniform node samples is large, more precisely, 10 times larger than the cost of moving a walker. Plots for this setting can be found in our technical report [Murai et al. 2018]. It is no longer clear that using many walkers and frequent random jumps achieves the most accurate head estimates, as this could rapidly deplete the budget. In fact, we observe that setting $w = 10$ or $b = 1$ yields poor estimates for both the smallest and largest out-degrees. While increasing the jump weight w or decreasing b sometimes improves estimates in the head, it rarely does so in the tail. The best results for the smallest out-degrees are often observed when setting $w = 1$ and $b = 10$ or 10^2. On the other hand, setting $(w = 0.1, b = 10^3)$ or $(w = 1, b = 10^2)$ usually achieves relatively small NRMSEs for the largest out-degree estimates.

Invisible in-edges, $c = 10$. Plots for this setting can be found in our technical report [Murai et al. 2018]. Unlike the scenario with visible in-edges, setting $w = 10$ and $b = 1$ often produces the most accurate estimates for the smallest out-degrees. This is because many of the datasets have nodes with no out-edges; these nodes can only be reached through a neighbor or through random node sampling. Conversely, the general trends for tail estimates are similar to those observed for the visible in-edges case: large values of w and small values of b yield less accurate estimates for the largest out-degree values. For $w = 1$, however, $b = 10^2$ often outperforms $b = 10^3$. On the other hand, for $w = 0.1$ there is little difference in the estimates for different values of b.