Comparison of Random Walk Based Techniques for Estimating Network Averages

Konstantin Avrachenkov, Vivek S Borkar, Arun Kadavankandy, Jithin K Sreedharan

▶ **To cite this version:**

Konstantin Avrachenkov, Vivek S Borkar, Arun Kadavankandy, Jithin K Sreedharan. Comparison of Random Walk Based Techniques for Estimating Network Averages. Computational Social Networks, Hien T. Nguyen; Vaclav Snasel, Aug 2016, Ho Chi Minh, Vietnam. pp.27 - 38, 10.1007/978-3-319-42345-6_3 . hal-01402800

HAL Id: hal-01402800
https://inria.hal.science/hal-01402800

Submitted on 25 Nov 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Comparison of random walk based techniques for estimating network averages

Konstantin Avrachenkov[1], Vivek S. Borkar[2]
Arun Kadavankandy[1], and Jithin K. Sreedharan[1]

[1] Inria Sophia Antipolis, France
{k.avrachenkov,arun.kadavankandy,jithin.sreedharan}@inria.fr
[2] IIT Bombay, India
borkar.vs@gmail.com

Abstract. Function estimation on Online Social Networks (OSN) is an important field of study in complex network analysis. An efficient way to do function estimation on large networks is to use random walks. We can then defer to the extensive theory of Markov chains to do error analysis of these estimators. In this work we compare two existing techniques, Metropolis-Hastings MCMC and Respondent-Driven Sampling, that use random walks to do function estimation and compare them with a new reinforcement learning based technique. We provide both theoretical and empirical analyses for the estimators we consider.

Keywords: Social network analysis, estimation, random walks on graph

1 Introduction

The analysis of many Online Social Networks (OSN) is severely constrained by a limit on Application Programming Interface (API) request rate. We provide evidence that random walk based methods can explore complex networks with very low computational load. One of the basic questions in complex network analysis is the estimation of averages of network characteristics. For instance, one would like to know how young a given social network is, or how many friends an average network member has, or what proportion of a population supports a given political party. The answers to all the above questions can be mathematically formulated as the solutions to a problem of estimating an average of a function defined on the network nodes.

Specifically, we model an OSN as a connected graph $\mathcal{G}$ with node set $\mathcal{V}$ and edge set $\mathcal{E}$. Suppose we have a function $f : \mathcal{V} \to \mathcal{R}$ defined on the nodes. If the graph is not connected, we can mitigate the situation by considering a modified random walk with jumps as in [2]. Our goal is to propose good estimators for the average of $f(.)$ over $\mathcal{V}$ defined as

$$\mu(\mathcal{G}) = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} f(v). \tag{1}$$

The above formulation is rather general and can be used to address a range of questions. For example to estimate the average age of a network we can take $f(v)$ as an age of node $v \in \mathcal{V}$, and to estimate the number of friends an average network member has we can set $f(v) = d_v$, where d_v is the degree of node v.

In this work, we compare in a systematic manner several random walk based techniques for estimating network averages $\mu(\mathcal{G})$ for a deterministic function f. In addition to familiar techniques in complex network analysis such as Metropolis-Hastings MCMC [6, 8, 13, 15] and Respondent-Driven Sampling (RDS) [9, 17, 18], we also consider a new technique based on Reinforcement Learning (RL) [1, 5]. If a theoretic expression for the limiting variance of Metropolis-Hastings MCMC was already known (see e.g., [6]), the variance and convergence analysis of RDS and RL can be considered as another contribution of the present work.

Metropolis-Hastings MCMC has being applied previously for network sampling (see e.g., [8, 10] and references therein). Then, RDS method [9, 17, 18] has been proposed and it was observed that in many cases RDS practically superior over MH-MCMC. We confirm this observation here using our theoretical derivations. We demonstrate that with a good choice of cooling schedule, the performance of RL is similar to that of RDS but the trajectories of RL have less fluctuations than RDS.

There are also specific methods tailored for certain forms of function $f(v)$. For example, in [7] the authors developed an efficient estimation technique for estimating the average degree. In the extended journal version of our work we plan to perform a more extensive comparison across various methods. Among those methods are Frontier Sampling [14], Snowball Sampling [11] and Walk-Estimate [12], just to name a few.

The paper is organized as follows: in Section 3 we describe various random walk techniques and provide error analysis, then, in Section 4 we compare all the methods by means of numerical experiments on social networks. Finally, in Section 5 we present our main conclusions.

2 Background and Notation

First we introduce some notation and background material that will make the exposition more transparent. A column vector is denoted by bold lower font e.g., $\mathbf{x}$ and its components as x_i. The probability vector $\boldsymbol{\pi}$ is a column vector. A matrix is represented in bold and its components in normal font (eg: $\mathbf{A}, A_{ij}$.) In addition, $\mathbf{1}$ represents the all-one column vector in $\mathbb{R}^n$.

For a sequence of random variables (rvs), $X_n \xrightarrow{D} X$, denotes convergence in distribution to X [3].

A random walk (RW) is simply a time-homogenous first-order Markov Chain whose state space is $\mathcal{V}$, the set of vertices of the graph and the transition probabilities are given as:

$$p_{ij} = \mathbb{P}\left(X_{t+1} = j | X_t = i\right) = \frac{1}{d_i}$$

if there is a link between i and j, i.e., $(i,j) \in E, d_i$ being the degree of node i. Therefore we can think of the random walker as a process that traverses the links of the graph in a random fashion. We can define $\mathbf{P}$ the transition probability matrix (t.p.m) of the Random walk as an $|\mathcal{V}| \times |\mathcal{V}|$ matrix, such that $\mathbf{P}_{ij} = p_{ij}$. Since we consider undirected networks, our random walk is time reversible. When the graph is connected the transition probability matrix $\mathbf{P}$ is irreducible and by Frobenius Perron Theorem there always exists a unique stationary probability vector $\boldsymbol{\pi} \in \mathbb{R}^{1\times|\mathcal{V}|}$ which solves $\boldsymbol{\pi}\mathbf{P} = \boldsymbol{\pi}$, which is in fact $\pi_i = \frac{d_i}{2|E|}$. Since our state space is finite the Markov chain is also positive recurrent and the quantities such as hitting times, and cover times are finite and well-defined. An important application of Random walks is in estimating various graph functions. The random walk based techniques can be easily implemented via APIs of OSNs and can also be easily distributed.

Let us define the fundamental matrix of a Markov chain given by $\mathbf{Z} := (\mathbf{I} - \mathbf{P} + \mathbf{1}\boldsymbol{\pi}^T)^{-1}$. For two functions $f, g : \mathcal{V} \to \mathbb{R}$, we define $\sigma^2_{ff} := 2\langle \mathbf{f}, \mathbf{Zf}\rangle_\pi - \langle f, f\rangle_\pi - \langle f, \mathbf{1}\boldsymbol{\pi}^T f\rangle_\pi$, and $\sigma^2_{fg} = \langle \mathbf{f}, \mathbf{Zg}\rangle_\pi + \langle \mathbf{g}, \mathbf{Zf}\rangle_\pi - \langle f, g\rangle_\pi - \langle f, \mathbf{1}\boldsymbol{\pi}^T g\rangle_\pi$, where $\langle \mathbf{x}, \mathbf{y}\rangle_\pi = \sum_i x_i y_i \pi_i$, for any two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{|\mathcal{V}|}, \pi$ being the stationary distribution of the Markov chain. In addition N denotes the number of steps of the random walk. By the Ergodic Theorem for Markov Chains applied to graphs the following is true [6], where f is an arbitrary function defined on the vertex set $\mathcal{V}$.

Theorem 1. *[6] For a RW $\{X_0, X_1, X_2 \dots X_n, \dots\}$ on a connected undirected graph,*

$$\frac{1}{N}\sum_{t=1}^{N} f(X_t) \to \sum_{x\in\mathcal{V}} \pi(x) f(x), \quad N \to \infty,$$

almost surely.

In addition the following central limit theorems also follow for RWs on graphs from the general theory of recurrent Markov chains [13].

Theorem 2. *[13] If f is a function defined on the states of a random walk on graphs, the following CLT holds*

$$\sqrt{N}\left(\frac{1}{N}\sum_{i=1}^{N} f(X_i) - \mathbb{E}_\pi(f)\right) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \sigma^2_{ff})$$

Theorem 3. *[13] If f, g are two functions defined on the states of a random walk, define the vector sequence $\mathbf{z}_t = \begin{bmatrix} f(x_t) \\ g(x_t) \end{bmatrix}$ the following CLT holds*

$$\sqrt{N}\left(\frac{1}{N}\sum_{t=1}^{N} \mathbf{z}_t - \mathbb{E}_\pi(\mathbf{z}_t)\right) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \boldsymbol{\Sigma}),$$

where $\boldsymbol{\Sigma}$ is 2 × 2 matrix such that $\Sigma_{11} = \sigma^2_{ff}, \Sigma_{22} = \sigma^2_{gg}$ and $\Sigma_{12} = \Sigma_{21} = \sigma^2_{fg}$.

In the following section we describe some of the most commonly used RW techniques to estimate functions defined on the vertices of a graph. We also give theoretical mean squared error (MSE) for each estimator defined as $MSE = \mathbb{E}[|\hat{\mu}(\mathcal{G}) - \mu(\mathcal{G})|^2]$.

3 Description of the techniques

In light of the Ergodic theorem of RW, there are several ways to estimate $\mu(\mathcal{G})$ as we describe in the following subsections.

Basic Markov Chain Monte Carlo technique (MCMC-technique)

MCMC is an algorithm that modifies the jump probabilities of a given MC to achieve a desired stationary distribution. Notice that by the Ergodic Theorem of MC, if the stationary distribution is uniform, then an estimate formed by averaging the function values over the visited nodes converges asymptotically to $\mu(G)$ as the number of steps tend to infinity. We use the MCMC algorithm to achieve $\pi(x) = 1/|\mathcal{V}|, \forall x \in \mathcal{V}$. Let p_{ij} be the transition probabilities of the original graph. We present here the Metropolis Hastings MCMC (MH-MCMC) algorithm for our specific purpose. When the chain is in state i it chooses the next state j according to transition probability p_{ij}. It then jumps to this state with probability a_{ij} or remains in the current state i with probability $1 - a_{ij}$, where a_{ij} is given as below

$$a_{ij} = \begin{cases} \min\left(\frac{p_{ji}}{p_{ij}}, 1\right) & \text{if } p_{ij} > 0, \\ 1 & \text{if } p_{ij} = 0. \end{cases} \tag{2}$$

Therefore the effective jump probability from state i to state j is $a_{ij}p_{ij}$, when $i \neq j$. It follows that the final chain represents a Markov chain with the following transition matrix $\mathbf{P}^{MH}$

$$P_{ij}^{MH} = \begin{cases} \frac{1}{\max(d_i, d_j)} & \text{if } j \neq i \\ 1 - \sum_{k \neq i} \frac{1}{\max(d_i, d_k)} & \text{if } j = i. \end{cases}$$

This chain can be easily checked to be reversible with stationary distribution $\pi_i = 1/n \ \forall i \in \mathcal{V}$. Therefore the following estimate for $\mu(\mathcal{G})$ using MH-MCMC is asymptotically consistent.

$$\hat{\mu}_{MH}(\mathcal{G}) = \frac{1}{N} \sum_{t=1}^{N} f(X_t).$$

By using the 1D CLT for RW from Theorem 2 we can show the following central limit theorem for MH.

Proposition 1. *(Central Limit Theorem for MH-MCMC) For MCMC with uniform target distribution it holds that*

$$\sqrt{N}\left(\hat{\mu}_{MH}(\mathcal{G}) - \mu(\mathcal{G})\right) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \sigma^2_{MH}),$$

as $N \to \infty$, *where* $\sigma^2_{MH} = \sigma^2_{ff} = \frac{2}{n}\mathbf{f}^T\mathbf{Z}\mathbf{f} - \frac{1}{n}\mathbf{f}^T\mathbf{f} - \left(\frac{1}{n}\mathbf{f}^T\mathbf{1}\right)^2$

Proof: Follows from Theorem 2 above. □

Respondent Driven Sampling technique (RDS-technique)

This estimator uses the unmodified RW on graphs but applies a correction to the estimator to compensate for the non-uniform stationary distribution.

$$\hat{\mu}^{(N)}_{RDS}(\mathcal{G}) = \frac{\sum_{t=1}^N f(X_t)/d(X_t)}{\sum_{t=1}^N 1/d(X_t)} := \frac{\sum_{t=1}^N f^{'}(X_t)}{\sum_{t=1}^N g(X_t)}, \tag{3}$$

where $f^{'}(X_t) := f(X_t)/d(X_t), g(X_t) := 1/d(X_t)$. The following result shows that the RDS estimator is asymptotically consistent and also gives the asymptotic mean squared error.

Proposition 2. *(Asymptotic Distribution of RDS Estimate) The RDS estimate* $\hat{\mu}_{RDS}(\mathcal{G})$ *satisfies a central limit theorem given below*

$$\sqrt{N}(\hat{\mu}^{(N)}_{RDS}(\mathcal{G}) - \mu(\mathcal{G})) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \sigma^2_{RDS}),$$

where σ^2_{RDS} *is given by*

$$\sigma^2_{RDS} = d^2_{av}\left(\sigma^2_1 + \sigma^2_2\mu^2(\mathcal{G}) - 2\mu(\mathcal{G})\sigma^2_{12}\right),$$

where $\sigma^2_1 = \frac{1}{|E|}\mathbf{f}^T\mathbf{Z}\mathbf{f}^{'} - \frac{1}{2|E|}\sum_x \frac{f(x)^2}{d(x)} - \left(\frac{1}{2|E|}\mathbf{f}^T\mathbf{1}\right)^2$, $\sigma^2_2 = \sigma^2_{gg} = \frac{1}{|E|}\mathbf{1}^T\mathbf{Z}\mathbf{g} - \frac{1}{2|E|}\mathbf{g}^T\mathbf{1} - (\frac{1}{d_{av}})^2$ *and* $\sigma^2_{12} = \frac{1}{2|E|}\mathbf{f}^T\mathbf{Z}\mathbf{g} + \frac{1}{2|E|}\mathbf{1}^T\mathbf{Z}\mathbf{f}^{'} - \frac{1}{2|E|}\mathbf{f}^T\mathbf{g} - \frac{1}{d_{av}}\frac{1}{2|E|}\mathbf{1}^T\mathbf{f}$

Proof. Let $f^{'}(x) := \frac{f(x)}{d(x)}$ and $g(x) := \frac{1}{d(x)}$. Define the vector $\mathbf{z}_t = \begin{bmatrix} f^{'}(x_t) \\ g(x_t) \end{bmatrix}$, and let $\mathbf{z}_N = \sqrt{N}\left(\frac{1}{N}\sum_{t=1}\mathbf{z}_t - \mathbb{E}_\pi(\mathbf{z}_t)\right)$. Then by Theorem 3, $\mathbf{z}_N \xrightarrow{\mathcal{D}} \mathcal{N}(0, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}$ is defined as in the theorem. Equivalently, by Skorohod representation theorem [3] in a space $(\Omega, \mathcal{F}, \mathbb{P})$, $\Omega \subset \mathbb{R}^2$, there is an embedding of $\mathbf{z}_N$ s.t. $\mathbf{z}_N \to \mathbf{z}$ almost surely (a.s.), such that $\mathbf{z} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$. Hence the distribution of $\sqrt{N}(\hat{\mu}^{(N)}_{RDS}(\mathcal{G}) - \mu(\mathcal{G}))$ is the same as that of

$$\frac{\sum_{t=1}^N f^{'}(X_t)}{\sum_{t=1}^N g(X_t)} \overset{\mathcal{D}}{=} \frac{\frac{1}{\sqrt{N}}z^{(N)}_1 + \mu_{f^{'}}}{\frac{1}{\sqrt{N}}z^{(N)}_2 + \mu_g} = \frac{z^{(N)}_1 + \sqrt{N}\mu_{f^{'}}}{z^{(N)}_2 + \sqrt{N}\mu_g} = \frac{z^{(N)}_1 + \sqrt{N}\mu_{f^{'}}}{\sqrt{N}\mu_g(1 + \frac{z^{(N)}_2}{\sqrt{N}\mu_g})}$$

$$= \frac{1}{\sqrt{N}\mu_g}(z^{(N)}_1 - \frac{z^{(N)}(1)z^{(N)}_2}{\sqrt{N}\mu_g} + \sqrt{N}\mu_{f^{'}} - \frac{z^{(N)}_2\mu_{f^{'}}}{\mu_g} + \mathcal{O}(\frac{1}{\sqrt{N}}))$$

This gives

$$\sqrt{N}\left(\frac{\sum_{t=1}^{N} f'(X_t)}{\sum_{t=1}^{N} g(X_t)} - \frac{\mu_{f'}}{\mu_g}\right) \xrightarrow{\mathcal{D}} \frac{1}{\mu_g}\left(z_1 - z_2\frac{\mu_{f'}}{\mu_g}\right),$$

since the term $\mathcal{O}(\frac{1}{\sqrt{N}})$ tend to zero in probability, and using Slutsky's lemma [3]. The result then follows from the fact that $\mathbf{z} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$. □

Reinforcement Learning technique (RL-technique)

Consider a connected graph $\mathcal{G}$ with node set $\mathcal{V}$ and edge set $\mathcal{E}$. Let $\mathcal{V}_0 \subset \mathcal{V}$ with $|\mathcal{V}_0| << |\mathcal{V}|$. Consider a simple random walk $\{X_n\}$ on $\mathcal{G}$ with transition probabilities $p(j|i) = 1/d(i)$ if $(i,j) \in \mathcal{E}$ and zero otherwise. Define $Y_n := X_{\tau_n}$ for $\tau_n :=$ successive times to visit $\mathcal{V}_0$. Then $\{(Y_n, \tau_n)\}$ is a semi-Markov process on $\mathcal{V}_0$. In particular, $\{Y_n\}$ is a Markov chain on $\mathcal{V}_0$ with transition matrix (say) $[[p_Y(j|i)]]$. Let $\xi := \min\{n > 0 : X_n \in \mathcal{V}_0\}$ and for a prescribed $f : \mathcal{V} \mapsto \mathcal{R}$, define

$$T_i := E_i[\xi],$$
$$h(i) := E_i\left[\sum_{m=1}^{\xi} f(X_m)\right],\ i \in \mathcal{V}_0.$$

Then the Poisson equation for the semi-Markov process (Y_n, τ_n) is [16]

$$V(i) = h(i) - \beta T_i + \sum_{j \in \mathcal{V}_0} p_Y(j|i)V(j),\ i \in \mathcal{V}_0. \tag{4}$$

Here $\beta :=$ the desired stationary average of f. Let $\{z\}$ be IID uniform on $\mathcal{V}_0$. For each $n \geq 1$, generate an independent copy $\{X_m^n\}$ of $\{X_m\}$ with $X_0^n = z$ for $0 \leq m \leq \xi(n) :=$ the first return time to $\mathcal{V}_0$. A learning algorithm for (4) along the lines of [1] then is

$$V_{n+1}(i) = V_n(i)\ + a(n)\mathbb{I}\{z = i\}\times$$
$$\left[\left(\sum_{m=1}^{\xi(n)} f(X_m^n)\right) - V_n(i_0)\xi(n) + V_n(X_{\xi(n)}^n) - V_n(i)\right], \tag{5}$$

where $a(n) > 0$ are stepsizes satisfying $\sum_n a(n) = \infty$, $\sum_n a(n)^2 < \infty$. (One good choice is $a(n) = 1/\lceil \frac{n}{N} \rceil$ for $N = 50$ or 100.) Here $\mathbb{I}\{A\}$ denotes indicator function for the set A. Also, i_0 is a prescribed element of $\mathcal{V}_0$. One can use other normalizations in place of $V_n(i_0)$, such as $\frac{1}{|\mathcal{V}_0|}\sum_j V_n(j)$ or $\min_i V_n(i)$, etc. Then this normalizing term ($V_n(i_0)$ in (5)) converges to β as n increases to ∞. This normalizing term forms our estimator $\hat{\mu}_{RL}^{(n)}(\mathcal{G})$ in RL based approach.

The relative value iteration algorithm to solve (4) is

$$V_{n+1}(i) = h(i) - V_n(i_0)T_i + \sum_j p_Y(j|i)V_n(j)$$

and (5) is the stochastic approximation analog of it which replaces conditional expectation w.r.t. transition probabilities with an actual sample and then makes an incremental correction based on it, with a slowly decreasing stepwise that ensures averaging. The latter is a standard aspect of stochastic approximation theory. The smaller the stepwise the less the fluctuations but slower the speed, thus there is a trade-off between the two.

RL methods can be thought of as a cross between a pure deterministic iteration such as the relative value iteration above and pure MCMC, trading off variance against per iterate computation. The gain is significant if the number of neighbours of a node is much smaller than the number of nodes, because we are essentially replacing averaging over the latter by averaging over neighbours. The V-dependent terms can be thought of as control variates to reduce variance.

MSE of RL Estimate
For the RL Estimate the following concentration bound is true [4]:

$$\mathbb{P}\left\{|\hat{\mu}_{RL}^{(N)}(\mathcal{G}) - \mu(\mathcal{G})| \geq \epsilon\right\} \leq K \exp(-k\epsilon^2 N).$$

Thus it follows that MSE is $\mathcal{O}(\frac{1}{\sqrt{N}})$ because

$$\begin{aligned}
\mathbb{E}|\hat{\mu}_{RL}^{(N)}(\mathcal{G}) - \mu(\mathcal{G})|^2 &= \int_0^\infty \mathbb{P}\left\{|\hat{\mu}_{RL}^{(N)}(\mathcal{G}) - \mu(\mathcal{G})|^2 \geq \epsilon\right\} d\epsilon \\
&= \int_0^\infty \mathbb{P}\left\{|\hat{\mu}_{RL}^{(N)}(\mathcal{G}) - \mu(\mathcal{G})| \geq \epsilon^{1/2}\right\} d\epsilon \\
&\leq \int_0^\infty K \exp(-k\epsilon N)\, d\epsilon = \mathcal{O}\left(\frac{1}{N}\right).
\end{aligned}$$

4 Numerical comparison

The algorithms explained in Section 3 are compared in this section using simulations on two real-world networks. For the figures given below, the x-axis represents the budget B which is the number of allowed samples, and is the same for all the techniques. We use the normalized root mean squared error (NRMSE) for comparison for a given B and is defined as

$$\text{NRMSE} := \sqrt{\text{MSE}}/\mu(\mathcal{G}), \quad \text{where MSE} = E\left[(\hat{\mu}(\mathcal{G}) - \mu(\mathcal{G}))^2\right].$$

For the RL technique we choose the initial or super-node $\mathcal{V}_0$ by uniformly sampling nodes assuming the size of $\mathcal{V}_0$ is given a priori.

4.1 Les Misérables network

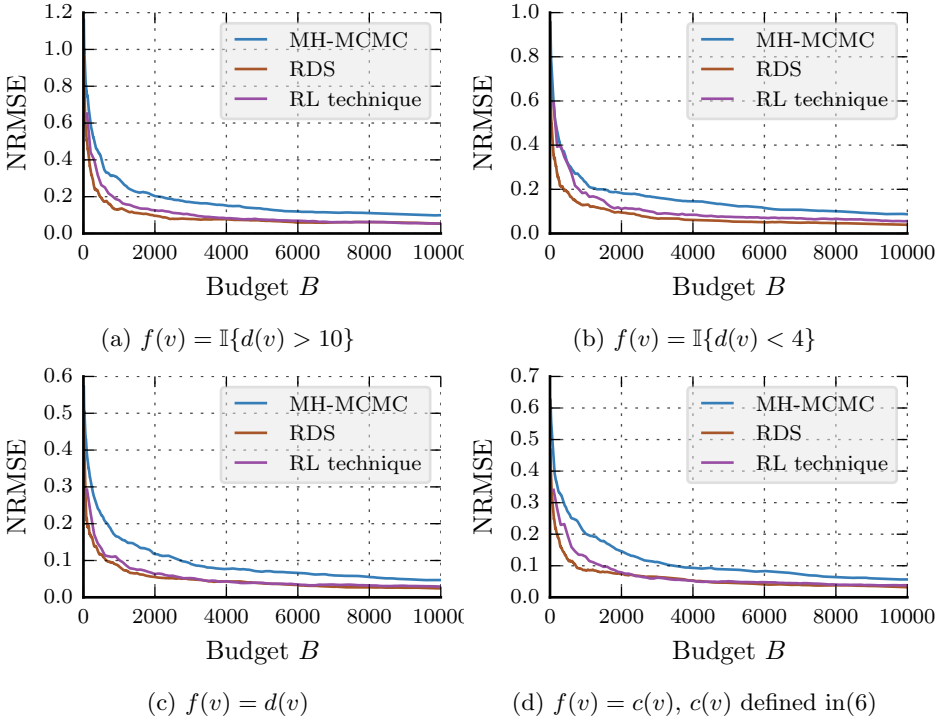
In Les Misérables network, nodes are the characters of the novel and edges are formed if two characters appear in the same chapter in the novel. The number of nodes is 77 and number of edges is 254. We have chosen this rather small

network in order to compare all the three methods in terms of theoretical limiting variance. Here we consider four demonstrative functions: a) $f(v) = \mathbb{I}\{d(v) > 10\}$ b) $f(v) = \mathbb{I}\{d(v) < 4\}$ c) $f(v) = d(v)$, where $\mathbb{I}\{A\}$ is the indicator function for set A and d) for calculating $\mu(\mathcal{G})$ as the average clustering coefficient

$$C := \frac{1}{|\mathcal{V}|}\sum_{v\in\mathcal{V}} c(v), \quad \text{where } c(v) = \begin{cases} t(v)/\binom{d_v}{2} & \text{if } d(v) \geq 2 \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

with $t(v)$ as the number of triangles that contain node v. Then $f(v)$ is taken as $c(v)$ itself.

The average in MSE is calculated from multiple runs of the simulations. The simulations on Les Misérables network is shown in Figure 1 with $a(n) = 1/\lceil \frac{n}{10} \rceil$ and the super-node size as 25.



(a) $f(v) = \mathbb{I}\{d(v) > 10\}$ (b) $f(v) = \mathbb{I}\{d(v) < 4\}$

(c) $f(v) = d(v)$ (d) $f(v) = c(v)$, $c(v)$ defined in(6)

Fig. 1: Les Misérables network: NRMSE comparisons

Study of asymptotic MSE: In order to show the asymptotic MSE expressions derived in Propositions 1 and 2, we plot the sample MSE as MSE $\times B$ in Figures 2a, 2b and 2c. These figures correspond to the three different functions we have

considered. It can be seen that asymptotic MSE expressions match well with the estimated ones.

4.2 Friendster network

We consider a larger graph here, a connected subgraph of an online social network called Friendster with 64,600 nodes and 1,246,479 edges. The nodes in Friendster are individuals and edges indicate friendship. We consider the functions a). $f(v) = \mathbb{I}\{d(v) > 50\}$ and b). $f(v) = c(v)$ (see (6)) used to estimate the average clustering coefficient. The plot in Figure 3b shows the results for Friendster graph with super-node size 1000. Here the sequence $a(n)$ is taken as $1/\lceil \frac{n}{25} \rceil$.

Now we concentrate on *single* sample path properties of the algorithms. Hence the numerator of NRMSE becomes absolute error. Figure 3c shows the effect of increasing super-node size while fixing step size $a(n)$ and Figure 3d shows the effect of changing $a(n)$ when super-node is fixed. In both the cases, the green curve of RL technique shows much stability compared to the other techniques.

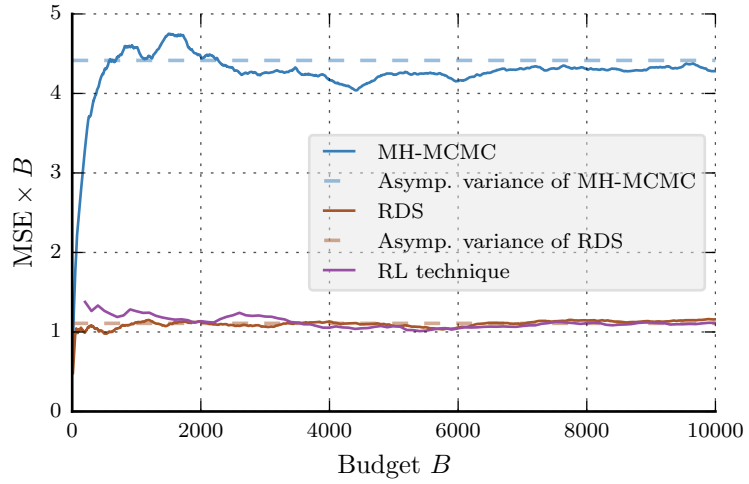
4.3 Observations

Some observations from the numerical experiments are as follows:

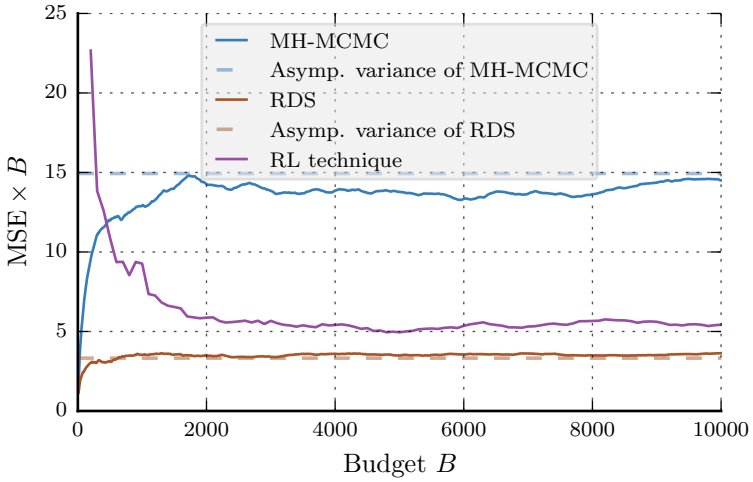
1. With respect to the limiting variance, RDS always outperforms the other two methods tested. However, with a good choice of parameters the performance of RL is not far from RDS;
2. In the RL technique, we find that the normalizing term $1/|\mathcal{V}_0| \sum_j V_n(j)$ converges much faster than the other two options, $V_t(i_0)$ and $\min_i V_t(i)$;
3. When the size of the super-node decreases, the RL technique requires smaller step size $a(n)$. For instance in case of Les Misérables network, if the super-node size is less than 10, RL technique does not converge with $a(n) = 1/(\lceil \frac{n}{50} \rceil + 1)$ and requires $a(n) = 1/(\lceil \frac{n}{5} \rceil)$;
4. If step size $a(n)$ decreases or the super node size increases, RL fluctuates less but with slower convergence. In general, RL has less fluctuations than MH-MCMC or RDS.

5 Conclusion and discussion

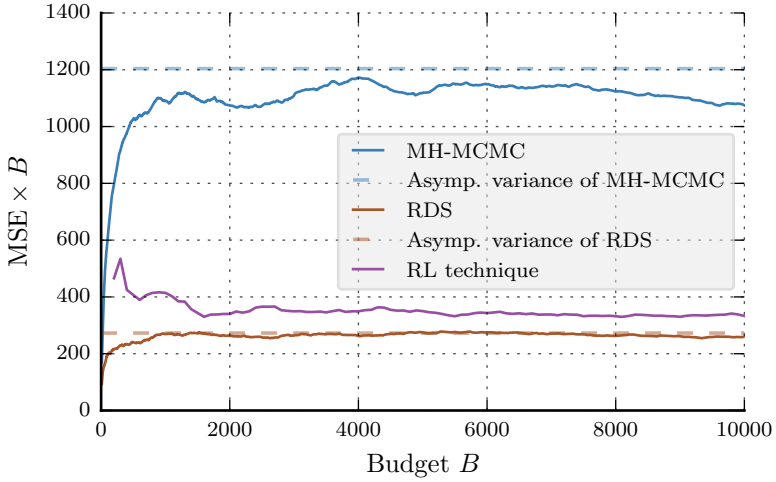
In this work we studied and compared the performances of various random walk-based techniques for function estimation on OSNs and provide both empirical and theoretical analyses of their performance. We found that in terms of asymptotic mean squared error (MSE), RDS technique outperforms the other methods considered. However, RL technique with small step size displays a more stable sample path in terms of MSE. In the extended version of the paper we plan to test the methods on larger graphs and involve more methods for comparison.



(a) $f(v) = \mathbb{I}\{d(v) > 10\}$

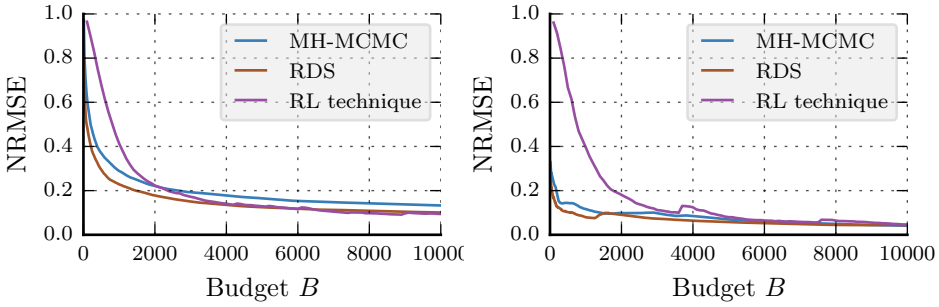


(b) $f(v) = \mathbb{I}\{d(v) < 4\}$

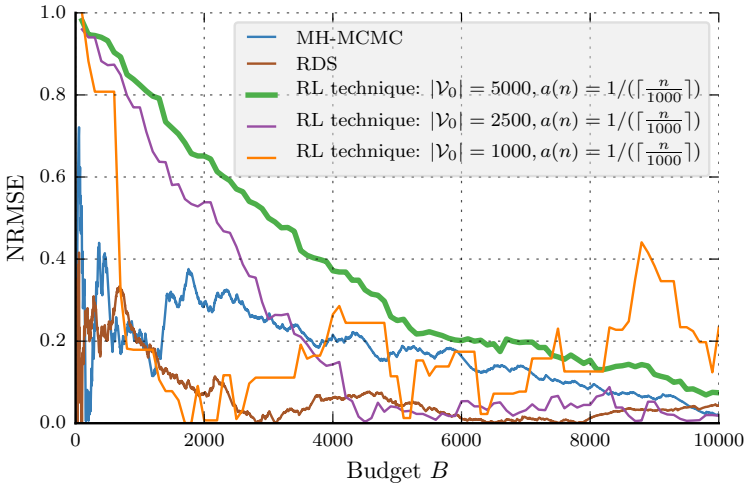


(c) $f(v) = d(v)$

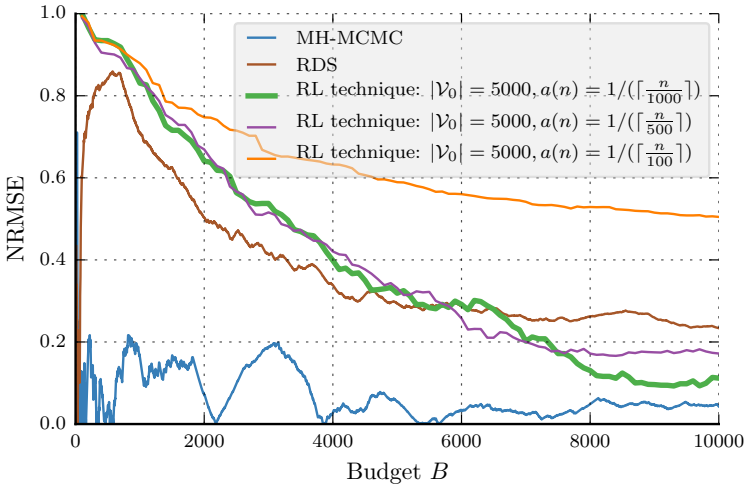
Fig. 2: Les Misérables network: asymptotic MSE comparisons



(a) $f(v) = \mathbb{I}\{d(v) > 50\}$ (b) $f(v) = c(v)$, $c(v)$ defined in(6)



(c) Single sample path: Varying super-node size



(d) Single sample path: Varying step size

Fig. 3: Friendster network: (a) & (b) NRMSE comparison, (c) & (d) Single sample path comparison with $f(v) = \mathbb{I}\{d(v) > 50\}$

.

6 Acknowledgements

This work was supported by CEFIPRA grant no.5100-IT1 "Monte Carlo and Learning Schemes for Network Analytics," Inria Nokia Bell Labs ADR "Network Science," and by the French Government (National Research Agency, ANR) through the "Investments for the Future" Program reference ANR-11-LABX-0031-01.

References

1. Abounadi, J., Bertsekas, D., Borkar, V.S.: Learning algorithms for markov decision processes with average cost. SIAM Journal on Control and Optimization **40**(3), 681–698 (2001)
2. Avrachenkov, K., Ribeiro, B., Towsley, D.: Improving random walk estimation accuracy with uniform restarts. In: Algorithms and Models for the Web-Graph, pp. 98–109. Springer (2010)
3. Billingsley, P.: Probability and measure. John Wiley & Sons (2008)
4. Borkar, V.S.: Stochastic approximation. Cambridge University Press (2008)
5. Borkar, V.S., Makhijani, R., Sundaresan, R.: Asynchronous gossip for averaging and spectral ranking. Selected Topics in Signal Processing, IEEE Journal of **8**(4), 703–716 (2014)
6. Brémaud, P.: Markov chains: Gibbs fields, Monte Carlo simulation, and queue. Springer (2013)
7. Dasgupta, A., Kumar, R., Sarlos, T.: On estimating the average degree. In: Proceedings of WWW, pp. 795–806 (2014)
8. Gjoka, M., Kurant, M., Butts, C.T., Markopoulou, A.: Walking in facebook: A case study of unbiased sampling of osns. In: Proceedings of IEEE INFOCOM, pp. 1–9 (2010)
9. Goel, S., Salganik, M.J.: Respondent-driven sampling as markov chain monte carlo. Statistics in medicine **28**(17), 2202–2229 (2009)
10. Leskovec, J., Faloutsos, C.: Sampling from large graphs. In: Proceedings of the 12th ACM SIGKDD, pp. 631–636 (2006)
11. Maiya, A.S., Berger-Wolf, T.Y.: Sampling community structure. In: Proceedings of WWW, pp. 701–710 (2010)
12. Nazi, A., Zhou, Z., Thirumuruganathan, S., Zhang, N., Das, G.: Walk, not wait: Faster sampling over online social networks. Proceedings of the VLDB Endowment **8**(6), 678–689 (2015)
13. Nummelin, E.: MC's for MCMC'ists. International Statistical Review **70**(2), 215–240 (2002)
14. Ribeiro, B., Towsley, D.: Estimating and sampling graphs with multidimensional random walks. In: Proceedings of the 10th ACM SIGCOMM, pp. 390–403 (2010)
15. Robert, C., Casella, G.: Monte Carlo statistical methods. Springer Science & Business Media (2013)
16. Ross, S.M.: Applied probability models with optimization applications. Courier Corporation (2013)
17. Salganik, M.J., Heckathorn, D.D.: Sampling and estimation in hidden populations using respondent-driven sampling. Sociological methodology **34**(1), 193–240 (2004)
18. Volz, E., Heckathorn, D.D.: Probability based estimation theory for respondent driven sampling. Journal of official statistics **24**(1), 79 (2008)