クエリ数に着目したグラフサンプリング手法の比較

岩崎　謙汰†　　首藤　一幸†

† 東京工業大学 情報理工学院 数理・計算科学系　〒 152-8552 東京都目黒区大岡山２丁目１２－１
E-mail: †iwasaki.k.ah@m.titech.ac.jp, ††shudo@is.titech.ac.jp

あらまし　ノード ID が未知の大規模なソーシャルネットワークの特徴量を推定するためには，ランダムウォークによるグラフサンプリング手法が有用である．実際のソーシャルネットワーク上のサンプリングでは隣接ノードを取得する API によるクエリを繰り返し使用することによって行われ，クエリによる取得はサンプリングの過程の中でボトルネックになりうる．なぜなら，多くのソーシャルネットワークサービスでは単位時間あたりに使用できるクエリ数が制限されている場合がほとんどだからである．しかし、既存のグラフサンプリング研究ではクエリ数に着目せずサンプルサイズに基づく手法比較がほとんどであり，現実のソーシャルネットワークの特徴量推定に対して適切にグラフサンプリング手法を推薦しているとは言えない状況となっている．本研究では，クエリ数に着目したグラフサンプリング手法の考え方を提示する．代表的なランダムウォークベースのグラフサンプリング手法である，Simple Random Walk with re-weighting (SRW-rw)，Non-Backtracking Random Walk with re-weighting (NBRW-rw)，Metropolis-Hastings Random Walk (MHRW) に対して，アルゴリズム中にクエリが必要となるタイミングを述べる．実験として，実際のソーシャルネットワークグラフに対してサンプルサイズ基準とクエリ数基準によるグラフサンプリングの精度評価を行い，どのように変化するか考察した．

キーワード　グラフサンプリング，ソーシャルグラフ，ランダムウォーク

1. はじめに

大規模ネットワークのサンプリングは，Online Social Networks (OSNs) や world wide web といったソーシャルネットワーク解析において基本的かつ重要な問題である．ネットワークが巨大でありネットワーク全体を計算処理するのが不可能である状況，もしくはノード ID を始めとしたネットワークの全体情報を取得できない状況下では，サンプリングはネットワークの特徴量を推定するための現実的な手段であり，多くの研究で使用されてきた [1] [2] [3]．このように，ネットワークをノードとエッジ集合であるグラフ構造と捉えサブグラフを抽出することをグラフサンプリング [4] と呼ぶ．

一般的に，ノード ID を始めとしたネットワークの全体情報は公開されていないため，ソーシャルネットワークを特徴量を推定することは簡単ではない．例えば，ノード ID をランダムにサンプルするような一様独立サンプリングは，ノード ID が未知であるため実現不可能である．したがって，隣接関係を辿っていくグラフサンプリング手法が現実的である．これをクローリング的手法と呼ぶ [5]．クローリング的手法では，特にランダムウォークベース手法が有用とされている．なぜなら，アルゴリズムがシンプルであり実装を適用しやすく，またマルコフ連鎖による解析により不偏グラフサンプリングが実現できるからである．ランダムウォークベース以外のカテゴリでは幅優先サンプリング (BFS) [6] などの走査的手法が提案されているが，サンプリングの偏りが未知のため特徴量推定には不向きである．

クローリング的手法のグラフサンプリング手法を実際のソーシャルネットワークに適用する場合，OSN の API やスクレイピングなどのクエリを繰り返し使用することでサンプリングを行う．ここでのクエリとは，インターネットにアクセスすることであるノードの隣接ノードリストを取得することを指すこととする．図 1 の例では，研究者がクエリによって隣接ノードリストを取得している．過去の研究の一例として [2]，Facebook 上のユーザの友達リストを繰り返し取得することで Facebook 上のサンプリングを行っている．サンプリングの過程においては，クエリによる隣接ノードリスト取得がボトルネックになりうる．なぜなら多くのソーシャルネットワークサービスでは単位時間あたりに使用できるクエリ数が制限されているからである．また，制限されていなかったとしても，計算機内メモリやディスクにアクセスする速度より通信によるアクセスの方が時間がかかることからも，クエリによる隣接ノードリスト取得がボトルネックになりうることが言える．

このことから，クエリ数に基づくグラフサンプリング手法の性能比較が必要である．しかし，既存研究によるグラフサンプリング手法の推定精度の比較 [7], [8] はサンプルサイズ (サンプル列の長さ) を基準とした実験が多く，現実のソーシャルネットワークのサンプリングにおいて適切な手法を推薦しているとは言えないという問題がある．

本研究では，クエリ数に着目したグラフサンプリング手法の比較を提案する．ランダムウォークベース手法の代表例である Simple Random Walk with re-weighting (SRW-rw) [5], [9], Metropolis-Hastings Random Walk (MHRW) [5], [9], [10], Non-backtracking Random Walk with re-weighting (NBRW-rw) [7] に対して，アルゴリズム中にクエリが必要となるタイミングを述べる．また実際のソーシャルネットワークグラフに対

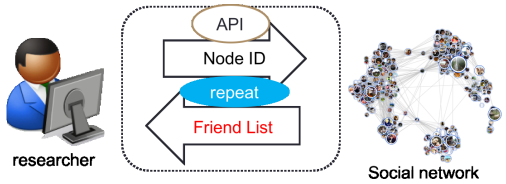


図 1: API による隣接ノードリストの取得の例

し，クエリ数に基づく特徴量推定を行い，グラフサンプリング手法の性能比較を行った．

本論文の構成は以下の通りである．本研究の背景として，第 2 章で用語の表記や定義の説明を行い，第 3 章でランダムウォークによるグラフサンプリング手法を述べる．第 4 章では計算機実験を行う．第 5 章で本研究の関連研究を述べ，第 6 章で本研究のまとめについて述べる．

2. 準　　備

本章ではグラフの表記方法や定義を述べた後に，本研究で用いるグラフの特徴量について述べる．さらに，既存のグラフサンプリング手法やクラスタ係数推定の既存手法について述べる．

2.1 表　　記

本論文では，ソーシャルネットワークを無向グラフ $G(V,E)$ で表す．$V=\{v_1,v_2,...,v_n\}$ はノード (頂点) の集合であり，グラフ全体のノード数を $n=|V|$ とする．Eはエッジの集合である．ノード $v\in V$ の隣接ノード集合を $N(v)=w\in V:(v,w)\in E$ とする．ノード v の次数を $d(v)=|N(v)|$ とする．

2.2 グラフの特徴量

本節では，複雑ネットワークを特徴づける代表的な特徴量を述べる．複雑ネットワークは実世界にける巨大で複雑な構造をしているネットワークと定義され，ソーシャルネットワークのほとんどが複雑ネットワークに属している．本論文では，グラフサンプリング手法の比較のため，次数分布とクラスタ係数の推定誤差を使用する．

2.2.1 次 数 分 布

全ノードに対して次数 k のノードの割合を $p(k)$ と表すと次数分布が定義できる．ソーシャルネットワークにおいては次数分布がべき乗則に従う．つまり $p(k)\propto k^{-\gamma}$ となる．これをスケールフリー性と呼ぶ．この性質が示すことは，ほとんどのノードは次数が小さいものであるが次数が大きいノードが一部存在するということである．この次数が大きいノードはしばしばハブと呼ばれる [11].

2.2.2 クラスタ係数

クラスタ係数はネットワークの重要な特徴量の一つであり，ネットワーク分析に盛んに使用されている [2]．複雑ネットワークの用語では三角形のことをクラスタと呼ぶ [11]．クラスタという用語は一般的には群れ，集団などを意味し，研究関係ではクラスタ分析，クラスタ同期など様々な意味に用いられる．本論文では三角形の意味のみで用いる．人間関係のネットワークに限らず，多くの現実のネットワークにはクラスタがたくさん存在する [11]．ネットワークのクラスタ係数を定義するためには，まずノード v を含む三角形の数から v のクラスタ係数 $c(v)$ を定義する．$d(v)$ 個ある v の隣接ノードから 2 つのノードを選び出す方法は $d(v)(d(v)-1)/2$ 通り存在する．これによって選ばれた 2 つのノード間にエッジが存在すれば，この 2 つのノードとノード v によって三角形が一つできる．したがって，v を含む三角形は最大 $d(v)(d(v)-1)/2$ 個ある．ここで v を含む三角形の数を $\triangle_i$ とするとクラスタ係数 $c(v)$ を次のように定義する．

$$c(v)=\begin{cases}0 & d(v)=0 \text{ または } d(v)=1\\ \dfrac{2\triangle_i}{d(v)(d(v)-1)} & \text{otherwise}\end{cases} \tag{1}$$

クラスタ係数の定義から $c(v)\in[0,1]$ である．

ネットワーク全体のクラスタ係数 C をノードごとのクラスタ係数の平均値

$$C=\frac{1}{n}\sum_{v\in V}c(v) \tag{2}$$

で定義する．$c(v)$ がノードに対しての値で，C はグラフ G に対する値である．

どのネットワークに対しても $C\in[0,1]$ であり，完全グラフに対しては $C=1$ となる．ほとんどの現実のネットワークにおいて，C は大きい．これは複雑ネットワークの特徴の一つである [11]．本研究では，各ランダムウォークに対し C の値をナイーブな手法と Counting Triangles 法によって推定する．

3. ランダムウォークによるグラフサンプリング

本章では，ランダムウォークによるグラフサンプリング手法のアルゴリズムと不偏性を説明する．まず最初にベースとなる不偏グラフサンプリングについて述べた後に各手法のアルゴリズムと推定方法を説明する．本研究では，代表的な 3 手法 Simple Random Walk with Re-weighting (SRW-rw)，Non-backtracking Random Walk with Re-weighting (NBRW-rw)，Metropolis-Hastings Random Walk (MHRW) を取り扱う．

3.1 不偏グラフサンプリング

ソーシャルネットワークのノードもしくはトポロジーに着目した特徴量を推定する時には，クローリングによる不偏グラフサンプリングが必要である．すなわち，ランダムウォークによって一様ノードサンプルを得ることを考える．本節では，特定の特徴を持つノードの割合を不偏推定することを目標とする．つまり，不偏グラフサンプリングとは，任意の関数 $f:V\to\mathbb{R}$ の一様分布に関する期待値を得るためランダムウォークによる推定の方法を構築することである．すなわち，一様分布を $\mathbf{u}\stackrel{\text{def}}{=}[u(1),u(2),\ldots,u(n)]=[1/n,1/n,\ldots,1/n]$ と表した時に

$$\mathbb{E}_{\mathbf{u}}(f)\stackrel{\text{def}}{=}\sum_{v\in V}f(v)\frac{1}{n} \tag{3}$$

が推定値の期待値となるようなサンプリング手法である．関数を適切に選択することによって求めたいノードの特徴量を特定することができる．例えば，グラフ G の次数分布 $(\mathbb{P}\{D_G=d\},d=1,2,\ldots,n-1)$ を推定したい場合は，

$v \in V$ に対して，$f(v) = \mathbb{1}_{\{d(v)=d\}}$，すなわち，もし $d(v) = d$ ならば $f(v) = 1$，そうでなければ $f(v) = 0$，となるような関数 f を選ぶ．

次に，グラフ G 上のランダムウォークによる不偏サンプリングのための数学的基礎になるマルコフ連鎖理論について述べる．グラフ G 上の一般的なランダムウォーク，もしくは可逆性のある既約な有限マルコフ連鎖 $\{X_t \in V, t = 0, 1...\}$ が次のような遷移確率行列 $\mathbf{P} \stackrel{\text{def}}{=} \{P(v,w)\}_{v,w\in V}$ を持つように定義する．

$$P(v,w) = \mathbb{P}\{X_{t+1} = w \mid X_t = v\}, v, w \in V, \tag{4}$$

$\forall v \in V$ に対して $\sum_{w\in V} P(v,w) = 1$ である．各エッジ $(v,w) \in E$ には遷移確率 $P(v,w) \geqq 0$ が割り当てられ，ランダムウォークはノード v からノード w への遷移が可能になる．また，グラフ G にセルフループがなくても，自己ノードへの遷移を設定してもよい．すなわち，$P(v,v) > 0$ となる $v \in V$ が存在しても良い．しかし，エッジが存在しないノード間は遷移できない．すなわち，$P(v,w) = 0, \forall (v,w) \notin E\,(v \neq w)$

定常分布 $\boldsymbol{\pi} = [\pi(v), v \in V]$ とする．任意の関数 $f : V \to \mathbb{R}$ に対して次のような推定量を定義する．

$$\hat{\mu}_t(f) \stackrel{\text{def}}{=} \frac{1}{t}\sum_{s=1}^{t} f(X_s) \tag{5}$$

定常分布 $\boldsymbol{\pi}$ に関する関数 f の期待値は次のように与えられる．

$$\mathbb{E}_{\boldsymbol{\pi}}(f) \stackrel{\text{def}}{=} \sum_{i\in V} \pi(i) f(i). \tag{6}$$

[12] より $\{X_t\}$ が定常分布 $\boldsymbol{\pi}$ の有限で既約なマルコフ連鎖であるとき，任意の初期分布 $\mathbb{P}\{X_0 = v\}, v \in V, (t \to \infty)$ に対して

$$\hat{\mu}_t(f) \to \mathbb{E}_{\boldsymbol{\pi}}(f)\ almost\ surely\ (a.s.) \tag{7}$$

が成立つ．ただし $\mathbb{E}_{\boldsymbol{\pi}}(|f|) < \infty$ とする．

3.2 Simple Random Walk with Re-weighting

まず最初に SRW-rw について述べる．SRW-rw がサンプルノードの収集から推定値の算出の仕方まで含めたサンプリングアルゴリズムなのに対し，SRW は単なるランダムウォークの遷移アルゴリズムの概要を表す．この手法は，SRW により得られたサンプル列と，不偏サンプリングを達成するための適切な再度重み付けプロセスに基づいて行われる．これは本質的にはマルコフ連鎖によって生成されたランダムサンプルに適用された重点サンプリングの特殊なケースである．この手法の基本的な考え方は，SRW の定常分布によって生じるサンプリングの偏りを再度重み付けによって正していくことである．

グラフ G 上の SRW のサンプル列の取得方法について述べる．SRW によって訪れたノードのサンプル列を表現するマルコフ連鎖を $\{X_t\}$ とする．この遷移確率行列を $\mathbf{P}^{SRW} = P^{SRW}(v,w)_{v,w\in V}$ とすると，$P^{SRW}(v,w)$ は

$$P^{SRW}(v,w) = \begin{cases} \frac{1}{d(v)} & (v,w) \in E \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

と表現できる．遷移確率の具体例を図 2 に示した．遷移確率行列

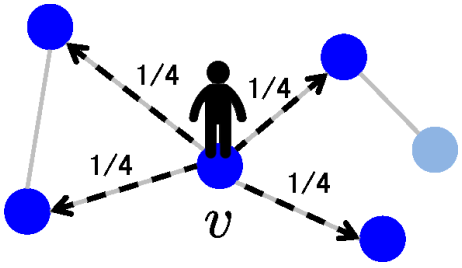


図 2: SRW の遷移確率の例

$\mathbf{P}^{SRW}$ は既約であり，定常分布 $\pi^{SRW}(v) = d(v)/(2|E|), v \in V$ に関して可逆であることが知られている．

SRW からの t 個のサンプル $\{X_s\}_{s=1}^{t}$ があると仮定する．この時，任意の関数 $f : V \to \mathbb{R}$ に対して，重み付け関数 $w : V \to \mathbb{R}$ は次のように決まる．

$$w(v) = \frac{u(v)}{\pi(v)} = \frac{1}{n} \cdot \frac{2|E|}{d(v)}, v \in V.$$

既約な有限マルコフ連鎖なので，$t \to \infty$ のとき

$$\hat{\mu}_t(wf) = \frac{1}{t}\sum_{s=1}^{t} w(X_s) f(X_s) \to \mathbb{E}_{\boldsymbol{\pi}}(wf) = \mathbb{E}_{\boldsymbol{u}}(f)\ a.s. \tag{9}$$

は強一致推定量である．しかし，これらは実用的ではない．なぜなら n や $|E|$ は通常事前に知ることはできないからである．したがって次のような推定量が代わりに使われる．$t \to \infty$ のとき

$$\frac{\hat{\mu}_t(wf)}{\hat{\mu}_t(w)} = \frac{\sum_{s=1}^{t} w(X_s) f(X_s)}{\sum_{s=1}^{t} w(X_s)} \to \mathbb{E}_{\boldsymbol{u}}(f)\ a.s. \tag{10}$$

この時，$w(v) = 1/d(v)$ と設定することで，不偏推定を行うことができる．本研究では $\hat{\mu}_t(wf)/\hat{\mu}_t(w), w(v) = 1/d(v)(v \in V)$ を SRW-rw における不偏推定をして扱う．

一例として，SRW-rw による次数分布の推定について述べる．対象グラフ G について，次数分布 $\mathbb{P}\{D_G = d\}$ を推定するために，$v \in V$ に対して関数 $f(v) = \mathbb{1}_{\{d(v)=d\}}$ を選択する．この時，任意の次数 d に対して

$$\frac{\hat{\mu}_t(wf)}{\hat{\mu}_t(w)} = \frac{\sum_{s=1}^{t} \mathbb{1}_{\{d(X_s)=d\}}/d(X_s)}{\sum_{s=1}^{t} 1/d(X_s)} \to \sum_{v\in V} \mathbb{1}_{\{d(v)=d\}} \frac{1}{n}\ a.s.,$$

とする．これは推定量 $\hat{\mu}_t(wf)/\hat{\mu}_t(w)$ が次数分布 $\mathbb{P}\{D_G = d\}$ の正当な不偏推定をもたらしていること示している．

SRW-rw のアルゴリズム内でクエリが必要となるのは，隣接ノードリストから遷移先ノードを決める時と，次数情報を重み付けの時に使用するときである．次数情報は訪れたノードからわかるため，クエリ数は訪問したノードの固有ノード数と等しくなる．

3.3 Non-backtracking Random Walk with Re-weighting

この節では Non-backtracking Random Walk with Re-weighting (NBRW-rw) [7] について述べる．

NBRW-rw は NBRW により得られたサンプル列と，不偏サンプリングを達成するための適切な再度重み付けプロセスに

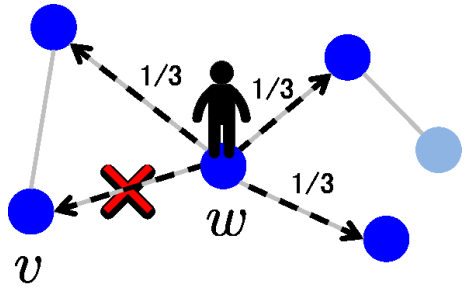


図 3: NBRW の遷移確率の例

基づいて行われる．後半の再度重み付けプロセスについてはSRW-rw と同様なプロセスを適用することができることが証明されている．また NBRW-rw による推定量は SRW-rw による推定量より低い分散値になることがわかっている [7]．この節では NBRW の遷移方法と，重み付けプロセスの概要について述べる．

NBRW の遷移方法は 1 つ前のノードに遷移することを避けながら，隣接ノードから一様ランダム選択し遷移するランダムウォークである．例外として，初期ノードと次数 1 のノードに存在する場合はこの限りでない．遷移確率の具体例を図 3 に示した．

NBRW-rw の再度重み付けプロセスについて述べる．NBRW によって訪れたノードの中で t ステップ目を $X'_t \in V$ とする．X'_t から次のノード X'_{t+1} を決定する時に X'_t だけでなく X'_{t-1} にも依存する．なぜなら，1 つ前のノードを避けるアルゴリズムだからである．したがって，$\{X'_t\}_{t\geqq 0}$ 自体は V ノード状態空間上でマルコフ連鎖ではない．しかし, [7] により次の式がなる立つことがわかっている．

$$\frac{1}{t}\sum_{s=1}^{t} f(X'_s) \to \mathbb{E}_{\boldsymbol{\pi}}(f) a.s. \tag{11}$$

$\boldsymbol{\pi}$ は SRW の定常分布であるため，SRW と同様の重み付けプロセスで不偏推定ができる．この証明は [7] に示されている．NBRW-rw のアルゴリズム内でクエリが必要となるのは，隣接ノードリストから遷移先ノードを決める時と，次数情報を重み付けの時に使用するときである．SRW-rw 同様に次数情報は訪れたノードからわかるため，クエリ数は訪問したノードの固有ノード数と等しくなる．

3.4 Metropolis-Hastings Random Walk

SRW や NBRW が次数の高いノードにサンプルが偏りやすいのに対し，MHRW は定常分布が一様分布になるように遷移確率を適切に変形することができる．Metropolis-Hastings (MH) アルゴリズム [13] は直接サンプルすることが難しい確率分布 μ からサンプリングするための，一般的な MCMC 手法である．今回のように一様分布 $\mu_v = \frac{1}{n}$ からサンプルを行いたい場合，次のような遷移確率を定義すると達成できることがわかっている．

$$P^{MH}(v,w) = \begin{cases} \min(\frac{1}{d(v)}, \frac{1}{d(w)}) & (v,w) \in E \\ 1 - \sum_{y \neq v} P^{MH}(v,y) & w = v \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

また，遷移確率の具体例を図 4 に示した．

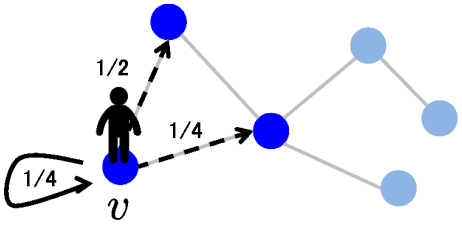


図 4: MHRW の遷移確率の例

この時，定常分布は $\pi^{MH}(v) = \frac{1}{n}$ となり，これは一様分布である．SRW と違い，MHRW は自己のノードに遷移することがある．その場合は，新たにサンプル列に追加する．この MH アルゴリズムは Algorithm1 のように表現できる．この時 $X_t \in V$ は MHRW の t 番目のノードであり，X_0 は恣意的に選ばれるとする．注目すべきことは，Algorithm1 は自己遷移確

Algorithm 1 MHRW における MH アルゴリズム (at time t)

```
隣接ノードリスト N(X_t) から一様ランダムにノード w を選択する
p ~ U(0,1) を生成する
if p ≦ d(X_t)/d(w) then
    X_{t+1} ← v
else
    X_{t+1} ← X_t
end if
```

率 $P^{MH}(v,v)$ を求める必要がないことと，t ステップ目のノード X_t の隣接ノードリストのノードの次数を全て知る必要があるわけではないことである．その代わり，ランダムに選ばれたノード w の次数情報だけあれば，w に遷移するかしないか決定することができる．

MHRW による不偏推定は SRW-rw と違って再度重み付け計算を必要としない．なぜなら，MHRW の定常分布が一様分布だからである．MHRW からの t 個のサンプル $\{X_t\}_{s=1}^{t}$ があると仮定すると，任意の関数 $f : V \to \mathbb{R}$ に対して既約な有限マルコフ連鎖なので, $t \to \infty$ のとき

$$\frac{1}{t}\sum_{s=1}^{t} f(X_t) \to \mathbb{E}_{\boldsymbol{u}}(f)\, a.s., \tag{13}$$

となる．

ここで注意すべき点は，Algorithm1 にも表現されている通り，自己遷移した場合も 1 サンプルとして追加する点である．

MHRW のアルゴリズム内でクエリが必要となるのは，ノード v に滞在している時，ノード v の隣接ノードリストを取得する時と，ノード v の遷移先候補ノードの次数情報を得る時に使用する．したがって，クエリ数は訪問したノードの固有ノード数が基本だが，自己ループを起こした場合追加クエリが必要となる．

3.5 クエリ数に着目したグラフサンプリング

特徴量推定のためのグラフサンプリングにおいて，隣接ノードリストを取得するクエリが発生しうるタイミングは次の 2 種類である．

- クローリングアルゴリズムで次の遷移ノードを決定する時
- 特徴量推定のための関数 $f(v)$ 計算時，$v \in V$ (式 5 より)

クローリングアルゴリズムに関しては，次の遷移先ノードを決定するために隣接ノードリストの取得クエリが必要である．つまり，一度訪れたノードに関しては必ずそのノードの隣接ノードリストを取得するクエリを 1 度使用する．SRW と NBRW については，訪れた固有ノード数がクローリングアルゴリズム中のクエリ数と等しくなる．MHRW の場合は，セルフループを起こした時，クエリが無駄になる可能性があるため，訪れた固有ノード数 $\leq$ クローリングアルゴリズム中のクエリ数となる．一度訪れたノードの隣接ノードリストは，再度訪れた時や，特徴量推定の時に再利用できる．

特徴量推定のための関数 $f(v)$ 計算時のクエリ数については，推定したい特徴量によって場合が異なる．大きく分けて，クローリングアルゴリズム中に使用したクエリによって取得した隣接ノードリストを再利用することで特徴量推定が可能な場合と，さらにクエリが必要となる特徴量が存在する．前者は $f(v)$ の計算が ノード v の隣接ノードリストの情報で計算できる場合である．具体例としては，平均次数，次数分布，Counting Triangles 法によるクラスタ係数推定などがある．後者は，ナイーブなクラスタ係数推定などが当てはまる．

図 5 は，ナイーブなクラスタ係数推定の時，つまり $f(v) = c(v)$ を計算するためにクエリを使用するノードの範囲の例である．青色のノードがクローリングアルゴリズム中に使用するクエリの範囲なのに対し，クラスタ係数を計算するために黄色の範囲までクエリで隣接ノードを取得する必要がある．このように，グラフサンプリングで求めたい特徴量によって必要なクエリの範囲が異なる．ちなみに，自分の隣接ノードとさらにその隣接ノードの隣接ノードまでの範囲のことをエゴネットワークと呼ぶ．

一方で，同じ特徴量を推定する場合でも，$f(v)$ の設定の仕方を変えると必要なクエリの範囲が変わる場合もある．クラスタ係数を例にとると，Counting Triangles 法では，$f(v) = \phi_k \cdot w(v)$ とする [8]．この時，ϕ_k は ランダムウォークの k ステップ目にいるとき，$k+1$ ステップ目のノードと $k-1$ ステップ目のノードの間にエッジが存在すれば 1，そうでなければ 0 の値をとる．$w(v)$ の定義はランダムウォークによって変化するが，次数情報で決まる関数である．したがって，クローリングアルゴリズム中に必要なクエリのみでクラスタ係数を推定することができる．

4. 実　　験

本章では，クエリ数基準とサンプルサイズ基準での SRW-rw，NBRW-rw，MHRW の性能を比較する．

4.1 データセット

Stanford Network Analysis Project (SNAP) [14] のデータセットで公開されているソーシャルネットワーク・引用ネットワークを用いて実験を行った．表 1 に各データセットの概要を示す．実用的にグラフサンプリングが行われるケースは，対象となるソーシャルネットワークは未知であるが，実験では全体像を知っているグラフデータに対して，シミュレーションを行う．

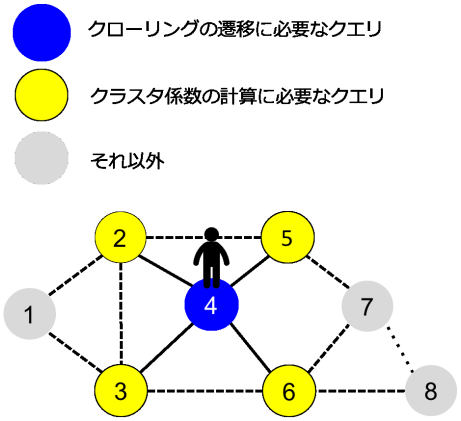
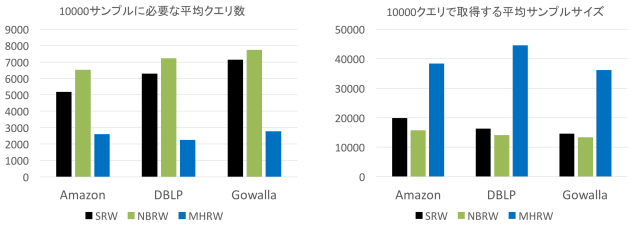


図 5: 隣接ノードリストを取得するクエリの範囲の例



(a) 10000 サンプルサイズに必要な平均クエリ数　**(b)** 10000 クエリあたりの平均サンプルサイズ

図 6: 各ネットワークに対するランダムウォーク別のクエリ数とサンプルサイズの関係

表 1: ネットワーク統計量

ネットワーク	全ノード数 n	平均次数	平均クラスタ係数
Amazon	334,863	5.530	0.3967
DBLP	317,080	6.622	0.6324
Gowalla	196,591	9.668	0.2367

4.2 各ランダムウォークのクエリ数

図 6 は各ランダムウォーク SRW，NBRW，MHRW を 100 回シミュレーションした時の平均クエリ数とサンプルサイズである．図 6a は SRW，NBRW，MHRW によるサンプリングを行った時，サンプルサイズ (サンプルノード列の長さ) が 10000 に達するまでに必要な平均クエリ数を表している．図 6b は，各ランダムウォークの 10000 クエリで取得できる平均サンプルサイズを表している．値が小さい順から NBRW，SRW，MHRW となっている．

4.3 クラスタ係数推定

本節では，図 7，8 で行った 2 種類のクラスタ係数推定の実験について述べる．どちらの実験でも，横軸をサンプルサイズ（左側）とクエリ数（右側）とした時の正規化平均二乗誤差 (NRMSE) [15] をプロットした．NMRSE の値が低いほど推定精度が良いと言える．NMRSE の計算方法はクラスタ係数の推定値を $\hat{C}$ とした時に $\frac{1}{C}\sqrt{\mathbb{E}[(\hat{C}-C)^2]}$ と計算できる．図 7

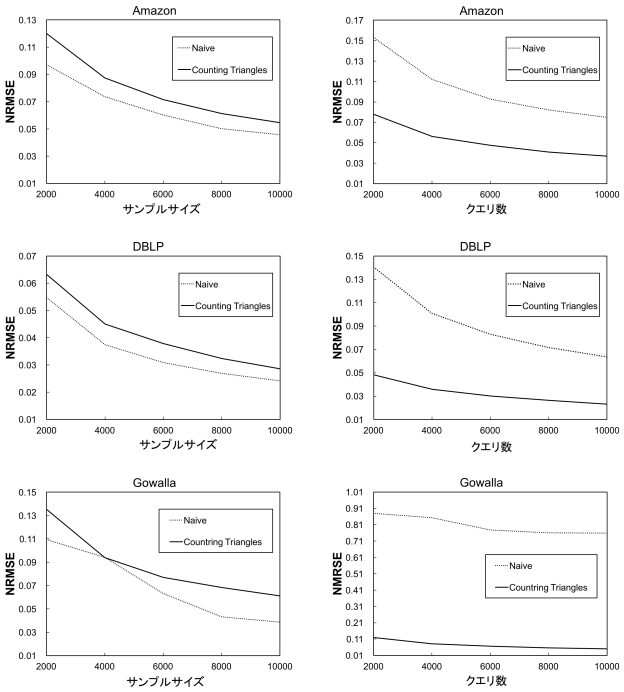


図 7: ナイーブな手法と Counting Triangles 法によるクラスタ係数推定の NRMSE

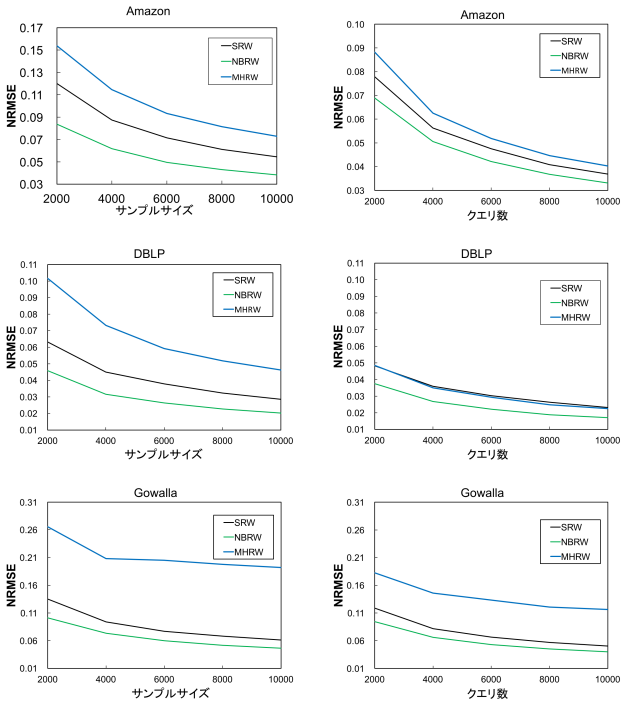


図 8: Counting Triangles 法によるランダムウォーク別のクラスタ係数推定の NRMSE

では，サンプリング方法を SRW-rw に固定し，ナイーブな推定手法と Counting Triangles 法による近似手法での NRMSE を比較した．Amazon，DBLP では始点を 100 個選び，独立にシミュレーションを行った．Gowalla では始点を 10 個選び独立にシミュレーションを行った．ナイーブな手法とは，式 5 の関数 $f : V \to \mathbb{R}$ に対して $f(v) = c(v), v \in V$ と定義する．ここでの $c(v)$ は式 1 で定義した関数である．つまり，ランダムウォークで訪れたノード毎にクラスタ係数を定義通り計算し，SRW-rw により不偏推定を行う．あるノードのクラスタ係数を定義通り計算するためには，そのノードの隣接ノードリストだけでなく，隣接ノードの隣接ノードリストまで取得する必要があるため，ランダムウォークの遷移以外のクエリが発生する．一方で，Counting Triangles 法はランダムウォークの遷移に必要なクエリに対して追加クエリなしにクラスタ係数を推定できる手法である．詳細は付録 1. に述べた．図 8 では，クラスタ係数推定方法を Counting Triangles 法に固定し，ランダムウォークによる遷移方法を変えた時の NEMSE を比較した．全てのネットワークでそれぞれ始点を 100 個選び，独立にシミュレーションを行った．それぞれ SRW-rw，NBRW-rw，MHRW と Counting Triangles 法を組み合わせた [8], [16]．MHRW と Counting Triangles 法の組み合わせについては過去に提案されていなかったため，本研究で新たにアルゴリズムを考案した．そのアルゴリズムは付録 1. に述べた．

4.4 次数分布の推定誤差

ランダムウォーク別に次数分布の推定を行う．それぞれのネットワークで相補累積分布関数 $\mathbb{P}\{D_g > d\}$ (CCDF) を評価し SRW-rw，NBRW-rw，MHRW とで比較する．$\mathbb{P}\{D_g > d\}$ を推定するためには，$f(v) = \mathbb{1}_{\{d(v)>d\}}, v \in V$ と定義し，それぞれ推定を行う．クラスタ係数同様，NRMSE を計算することで推定精度を比較する．ここでの NRMSE の計算方法は，$\frac{1}{x}\sqrt{\mathbb{E}[(\hat{x}(t) - x)^2]}$ となる．ここで $\hat{x}(t)$ は t サンプル取ったときの推定値であり，x は真値である．この時不偏推定であれば，$x = \lim_{t\to\infty} \hat{x}(t)$ となる．

図 9 は各ネットワークに対して 100 個の始点から独立にシミュレーションを行い，各手法の NRMSE をプロットした．左側がサンプルサイズ基準の精度であり，右側がクエリ数基準の精度である．値が小さいほど精度が良い．MHRW の推定精度も同じように計算したが，SRW-rw と NBRW-rw からかけ離れて悪い結果が出たため，今回 SRW-rw と NBRW-rw の比較のみグラフに表示した．

4.5 考　　察

第 3.5 節で述べた通り，クエリ数に着目してグラフサンプリング手法を比較する．

図 6 からは，クローリングアルゴリズム中に必要なクエリ数をランダムウォーク別に比較することができる．図 7 は，クローリングアルゴリズムは固定して $f(v)$ の設定を変えた時の例である．図 8,9 は $f(v)$ を固定してクローリングアルゴリズムを変えた時の例である．

図 7 を見ると，左側のサンプルサイズ基準と右側のクエリ数基準では，精度が逆転していることがわかる．クエリ数基準ではナイーブ手法より Counting Triangles 法の方が NRMSE が小さいため精度が良いのに対し，サンプルサイズ基準ではナイーブ手法の方が精度が良い．これは，Counting Triangles 法が $f(v)$ を確率的に計算しているのに対し，ナイーブ手法では定義通りクラスタ係数計算しているためである．しかし，ナイーブ手法は $f(v)$ の計算に追加のクエリが必要なため，クエ

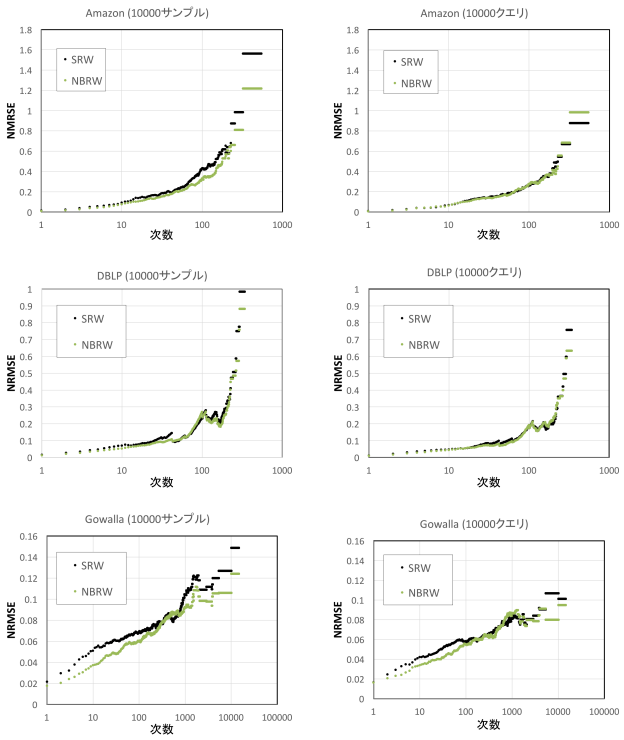


図 9: $\mathbb{P}\{D_g > d\}$ を推定した時の次数 d 当たりの NRMSE

リ数基準で比較した場合は Counting Triangles 法の方が良い結果になる．この場合，現実のソーシャルネットワークでのサンプリングを考えると，クエリ数基準の実験結果を採用すべきである．

図 8,9 の左側のグラフはサンプルサイズ基準でのランダムウォーク比較であるが，どの結果も NBRW，SRW，MHRW の順番に精度が良い．サンプルサイズ基準の NBRW vs. SRW の結果は [7], [8] ですでに述べられている．図 9 のサンプルサイズ基準の次数分布の NRMSE は概ね SRW より NBRW の方が低い値を取っており，NBRW の方が精度が高いという結論が得られるのに対し，クエリ数基準の結果を見ると NBRW，SRW の結果が拮抗しているように見える．これは，図 6 からわかるように，NBRW より SRW の方が 1 クエリあたりのサンプルサイズが大きいためである．つまり，クエリ数基準で考えた時，SRW は NBRW に対してサンプルサイズ基準のときよりも精度が縮まる，もしくは逆転することが予想される．同様に図 8 のクエリ数基準の NBRW と SRW の精度はサンプルサイズ基準のときよりも縮まっている．また，MHRW と SRW の精度の差も同様である．図 8 のサンプルサイズ基準では，MHRW より SRW の方が明らかに精度が良いが，クエリ数基準の結果では精度の差が縮まり，DBLP 上での実験では逆転が生じている．

5. 関 連 研 究

本章では，グラフサンプリングの関連研究について述べる．ネットワークから不偏サンプリングを達成する研究は多くなされている．Gjoka ら [2] は，SRW による定常分布から再度重み付けをすることで不偏サンプリングを達成する SRW-rw と MH アルゴリズムによる MHRW との比較を行い，SRW-rw が精度面で優位であることを示した．Lee ら [7] は，SRW-rw の派生である NBRW-rw を提案し，NBRW-rw の不偏性と SRW-rw よりも優位であることを理論面と実験面で示した．

グラフの特徴量を推定する目的のサンプリングについての研究もいくつかなされている．クラスタ係数の推定に特化した手法については，Hardiman と Katzir [16] が近似手法である Counting Triangles 法のアイデアを初めて提案し，SRW-rw と組み合わせた手法を提案した．岩崎ら [8] が Counting Triangles 法と NBRW-rw を組み合わせたクラスタ係数推定手法を提案し，SRW-rw よるも優位であることを示した．MHRW と Counting Triangles 法の組み合わせは過去に提案されておらず，本研究で初めて提案し，アルゴリズムの概要を付録 1. に示した．

Chiericetti ら [17] は，一様サンプリングにおいてのクエリ複雑性について述べた．論文中の対象手法は MHRW とグラフの事前情報を必要とする Rejection sampling と Maximum-degree sampling であるため，本研究の対象手法とは異なる．

6. 結　　論

現実のソーシャルネットワークにおけるグラフサンプリングではクエリ数基準の性能比較が重要であり，過去の研究においてクエリ数基準での実験がなされていない場合，その研究で推薦された手法が現実のソーシャルネットワークにとって有用な手法とは異なる可能性がある．また，今後新規のグラフサンプリング手法を提案する場合はクエリ数基準の性能比較の結果を実験に入れるべきである．

本研究では，クエリ数基準のグラフサンプリングを考える時の着目点を示した後に，従来のサンプルサイズ基準とクエリ数基準での特徴量推定の精度の差異を実験により示した．また，MHRW と Counting Triangles 法の組み合わせによる新規のクラスタ係数推定手法に対して，クエリ数基準の実験を行う例を示した．従来のグラフサンプリング手法である SRW-rw，NBRW-rw，MHRW に関しては，サンプルサイズ基準では NBRW-rw，SRW-rw，MHRW の順に精度が良いという評価だったが，クエリ数基準に変えることで各手法間の差が縮まり，対象グラフと推定する特徴量によっては逆転が起きる可能性を示した．また，特徴量推定の関数 $f(v)$ を手法別に変えた場合もサンプルサイズ基準とクエリ数基準で結果が逆転する例を示した．

今後の課題は，クエリ数とサンプルサイズ数の関係を理論的に表すことである．サンプルサイズと特徴量推定の精度の関係は過去の研究で出されているので，その結果を利用しクエリ数と特徴量推定の精度の関係を求めることが予想される．

謝辞

本研究の一部は，国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の委託業務として行われました．本研究は JSPS 科研費 25700008 および 16K12406 の助成を受けたものです．

文　献

[1] Y.Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong. Analysis of topological characteristics of huge online social networking services. In *Proceedings of the 16th international conference on World Wide Web*, pp. 835–844. ACM, 2007.

[2] M. Gjoka, M. Kurant, C.T. Butts, and A. Markopoulou. Walking in Facebook: A case study of unbiased sampling of OSNs. In *Proceedings IEEE Infocom*, pp. 1–9. IEEE, 2010.

[3] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pp. 29–42. ACM, 2007.

[4] J. Leskovec and C. Faloutsos. Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 631–636. ACM, 2006.

[5] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou. Practical recommendations on crawling online social networks. *Selected Areas in Communications, IEEE Journal on*, Vol. 29, No. 9, pp. 1872–1892, 2011.

[6] M. Kurant, A. Markopoulou, and P.dd Thiran. Towards unbiased bfs sampling. *Selected Areas in Communications, IEEE Journal on*, Vol. 29, No. 9, pp. 1799–1809, 2011.

[7] C. H. Lee, X. Xu, and D. Y. Eun. Beyond random walk and metropolis-hastings samplers: why you should not backtrack for unbiased graph sampling. In *ACM SIGMETRICS Performance Evaluation Review*, Vol. 40, pp. 319–330, 2012.

[8] K. Iwasaki, K. Shudo. Estimating the clustering coefficient of a social network by a non-backtracking random walk. In *IEEE BigComp 2018*, pp. 114–118. IEEE, 2018.

[9] A. H. Rasti, M. Torkjazi, R. Rejaie, N. Duffield, W. Willinger, and D. Stutzbach. Respondent-driven sampling for characterizing unstructured overlays. In *INFOCOM 2009, IEEE*, pp. 2701–2705. IEEE, 2009.

[10] M. Al Hasan and M. J. Zaki. Output space sampling for graph patterns. *Proceedings of the VLDB Endowment*, Vol. 2, No. 1, pp. 730–741, 2009.

[11] 増田直紀, 今野紀雄. 複雑ネットワーク. 近代科学社, 2010.

[12] G. L. Jones, et al. On the markov chain central limit theorem. *Probability surveys*, Vol. 1, pp. 299–320, 2004.

[13] W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, Vol. 57, No. 1, pp. 97–109, 1970.

[14] Stanford large network dataset collection. https://snap.stanford.edu/data/.

[15] K. Avrachenkov, B. Ribeiro, and D. Towsley. Improving random walk estimation accuracy with uniform restarts. In *International Workshop on Algorithms and Models for the Web-Graph*, pp. 98–109. Springer, 2010.

[16] S. J. Hardiman and L. Katzir. Estimating clustering coefficients and size of social networks via random walk. In *Proceedings of the 22nd international conference on World Wide Web*, pp. 539–550. International World Wide Web Conferences Steering Committee, 2013.

[17] F. Chiericetti, A. Dasgupta, R. Kumar, S. Lattanzi, and T. Sarlós. On sampling nodes in a network. In *Proceedings of the 25th International Conference on World Wide Web*, pp. 471–481. International World Wide Web Conferences Steering Committee, 2016.

付　録

1. MHRW による Counting Triangles 法

本付録では，本研究の実験に用いたクラスタ係数推定の近似アルゴリズムである Counting Triangles 法の基本的な考え方と今まで提案されていなかった MHRW による Counting Triangles 法のアルゴリズムを提案する．

Counting Triangles 法は，ランダムウォーク中に発生する三角形構造を調べることでクラスタ係数を推定する技法である．これまで SRW と NBRW に対して Counting Triangles 法を適用する手法が提案されてきた [8], [16]．Counting Triangles 法の良い点は，ランダムウォークの遷移に必要なクエリに対して追加のクエリが必要ない点である．クラスタ係数を定義通り計算する手法では，クラスタ係数を計算するのに追加のクエリが必要であった [2]．したがって，追加のクエリが必要かどうかは重要なポイントの 1 つである．

Counting Triangles 法の基本的な考え方は，ランダムウォーク中に訪問した次数 2 以上のノード v に対して，v の隣接ノードリストから一様ランダムに 2 つのノード v_1, v_2 を選択する．v_1, v_2 間にエッジが存在すれば，三角形構造としてカウントする．ここで三角形構造が存在する確率の期待値がノード v のクラスタ係数に等しくなるように重み付け係数を定義する．SRW の場合，重み付け係数が $d(v)/(d(v)-1)$ であり，NBRW の場合重み付け係数が 1 である．次数が 1 以下のノードはクラスタ係数が 0 なので，三角形構造の存在を確認する必要がない．実装上では，$v_1 \in N(v_2)$ または $v_2 \in N(v_1)$ が確認できればよい．つまり，v_1または v_2 の隣接ノードリストを取得する必要がある．SRW と NBRW による Counting Triangles 法では，ノード v に訪問した時に $v_1 = \{v$ の 1 ステップ前に訪問したノード $\}$, $v_2 = \{v$ の 1 ステップ後に訪問したノード $\}$ と定義することで，隣接ノードリストから一様ランダムに 2 ノード選択し，一方のノードの隣接ノードリストを知っている状態を満たしている．また，追加クエリも必要としない．

続いて MHRW による Counting Triangles 法について述べる．MHRW の遷移先ノードは隣接ノードリストから一様ランダムに選ばれるノードではないため，SRW や NBRW のように $v_1 = \{v$ の 1 ステップ前に訪問したノード $\}$, $v_2 = \{v$ の 1 ステップ後に訪問したノード $\}$ と定義することができない．したがって，v_1, v_2 の選び方を工夫する必要がある．

本研究では，MHRW で訪れた次数 2 以上のノード v に対して，v_1 を {MHRW の遷移アルゴリズム Algorithm1 中の遷移先候補ノード w} と定義し，v_2を $\{v$ の隣接ノードリストから v_1を除いたリスト $N(v)/\{v_1\}$ から一様ランダムに選択したノード $\}$ と定義する．この時，$v_2 \in N(v_1)$ である確率の期待値は，ノード v のクラスタ係数に等しい．また，MHRW では遷移アルゴリズム内で遷移先候補ノードへの受理確率を求めるために，遷移先候補ノードの次数を知る必要があるため，v_1 の隣接ノードリストを遷移アルゴリズム内で取得する．したがって，Counting Triangles 法を適用するにあたって追加クエリは必要ない．