精度とそのばらつきに着目したグラフ生成モデルの比較

福田　萌斐†　　中嶋　一貴†　　首藤　一幸†

† 東京工業大学　〒 152–8550 東京都目黒区大岡山 2–12–1
E-mail: †{fukuda.m.ai, nakajima.k.an}@m.titech.ac.jp, shudo@is.titech.ac.jp

あらまし　グラフ構造の特徴を比較して分析したり，現実のネットワークを模倣するために，グラフの生成モデルは使用されている．現実のネットワークを模倣するために生成モデルを用いる場合は，目標とするグラフ統計量の生成ごとの誤差とばらつきが小さいことが望ましい．しかしながら，既存の多くの生成モデルは，ランダムにエッジを張ることでグラフを生成するため，生成グラフの統計量が生成ごとにどの程度の誤差やばらつきをもつか明らかではない．本研究では，ネットワークのトポロジを，ノードの次数を基準として分析するために提唱された，dK-series という枠組みに基づくランダムグラフ生成モデルによる生成グラフの統計量の生成ごとの誤差とばらつきを考察する．さらに，新たに次数分布とクラスタ係数を入力としたグラフ生成モデルである 1K+ を提案し，そのモデルが他の生成モデルよりも誤差を小さくできることを示す．

キーワード　グラフサンプリング，ソーシャルネットワーク，グラフ生成モデル

1 はじめに

ソーシャルネットワーキングサービス（SNS）の利用者は年々増加しており，2020 年現在で Facebook[1] は 24 億人，Twitter[2] は 3 億人が利用している．このような巨大なソーシャルネットワークを，ユーザをノード，ユーザ間の関係をエッジとしたグラフ構造として解析する研究が盛んに行われている [1,2].

そうした巨大なグラフ構造は大規模かつ全体のトポロジが不明な場合が多く，サンプリングにより全体の統計量を推定する試みがなされている [3,4]．サンプリングの手法には主にランダムウォークが用いられており [5–7]，それに基づいた推定アルゴリズムの研究が盛んにされてきたが，推定できる主な統計量はノード数，次数分布，クラスタ係数など局所的な情報から計算できる統計量に限られるという問題点がある．

そうした中，推定が容易である局所的な統計量を入力とした生成モデルにより生成されたグラフを用いて，他の推定が困難な統計量までをも推定する 2.5K グラフの生成モデルが Gjoka らによって提案された [8]．これは dK-series と呼ばれるランダムグラフ生成モデルの枠組み [9] に基づいており，dK-series では次数の確率分布を入力として与えグラフを生成する．ノードの組の次数分布と次数に依存するクラスタ係数を入力として生成された 2.5K グラフが，現実のネットワークをよく模倣できることが示されている [8].

2.5K グラフの生成モデルで生成されたグラフは，大域的に定義された統計量も高精度に推定することが可能であるが，ランダムにグラフを生成する特徴上，同じ入力でも，入力以外の統計量が生成ごとに異なるグラフが生成される．生成されるグラフは，入力する統計量以外について，生成されるごとに大きくばらつくのか，または常に一定の範囲に収まり信頼できるのかということが明らかではない．

[1] http://www.facebook.com　[2] https://twitter.com

表 1　記　　法

$G(V,E)$	重みなしの無向グラフ
V,E	ノード集合，エッジ集合
V_k	次数が k のノードの集合
n,m	ノード数，エッジ数
v_i	G のノード
d_i	ノード v_i の次数
$\bar{d}$	G の平均次数 $\frac{m}{n}$
c_i	ノード v_i のクラスタ係数 [10]
$\bar{c}$	G の平均クラスタ係数 [10]
$\bar{c}(k)$	次数が k のノードにおける平均クラスタ係数 [11]

本研究では，実際のネットワークのデータセットに対して，dK-series に基づく生成モデルを用いて多数のグラフを生成し，生成されたグラフの各統計量の精度や信頼性，実行時間について比較・考察した．さらに，新たに次数分布とクラスタ係数を入力してグラフを生成するモデル（1K+）を提案し，それについても同様に比較・考察した．

2 準　　備

本論文では，ノードの集合 V とエッジの集合 E によって定義されるグラフ構造 $G(V,E)$ を取り扱う．本稿で扱うグラフ構造は重みなし，無向であり，多重辺と自己ループがないものとする．

表 1 に記法をまとめて示す．

2.1 関 連 研 究

Erdős-Rényi モデル

Erdős-Rényi モデル（ER モデル）は 1950 年代終わりに導入されたランダムグラフモデルである [12].

ER モデルには，$G(n,m)$ モデル，$G(n,p)$ モデルと呼ばれる 2 種類のモデルがある．$G(n,m)$ モデルは，n 個のノードと m

本のエッジをもつグラフの集合の中から一様ランダムに選択するモデルである．対して，$G(n,p)$ モデルは，独立に確率 p で各頂点のペアに対してエッジを構築するモデルである．すなわち，$G(n,p)$ モデルで期待されるエッジの本数は $\binom{n}{2}p$ となる．

ER モデルによりランダムグラフを生成する主な動機は，同じノード数 n，エッジ数 m を持つグラフが一般的にどのような性質を満たすかを検証することであり，現実のネットワークを模倣するモデルとして十分ではない．$G(n,m)$ モデルはそれぞれのエッジを張る確率が互いに独立ではないため，一般的には $G(n,p)$ モデルがよく使われており，$G(n,p)$ モデルでは通常 p を n の関数とみて $n \to \infty$ における構造を考える．

コンフィグレーションモデル

次数分布を固定したランダムグラフの生成モデルをコンフィグレーションモデルと呼ぶ．コンフィグレーションモデルは 1970 年代から研究されており [13–21] ，ここでは，Newman, Strogatz and Watts によるアプローチを用いてモデルを適用する [22,23].

$d_i \cdots d_n$ は独立で $P(d_i = k) = \frac{|V_k|}{n}$ を満たすとする．ノード数 n と次数分布 $P(d_i = k)$ を入力とし，ノード v_i から d_i 本の半辺が出ているとし，ランダムに 2 つの半辺を選んで繋ぎエッジを構築する．

この構成方法では多重辺や自己ループが生成される可能性がある．多重辺や自己ループを回避するためのアルゴリズムも提案されている [24].

*d*K-series

Priya らによって提唱された dK-series [9] は，あるグラフについて，そのうち d 個のノードからなる連結な部分グラフにおける全ての次数相関を特定する確率分布 $P(k_1, k_2, \cdots k_d)$ を用いて，グラフを系統的に特徴付ける枠組みである．

0K グラフでは，グラフ G の次数平均 $\bar{k}$ のみを固定する．すなわち，前述の Erdős-Rényi モデルにおける $G(n,m)$ モデルが相当する．

1K グラフは，ノードの次数分布 $P(k)$ が与えられたグラフであり，前述のコンフィグレーションモデルによって生成できる．

2K グラフは，グラフ G の各エッジがどの次数をもつノード同士を繋いだものかを，Joint Degree Distribution（JDD）$P(k_1, k_2)$ として特定し固定して生成される．これはコンフィグレーションモデルの拡張によって実現できる [9]．多重辺と自己ループを回避した 2K グラフの生成モデルは Isabelle らによって提案されている [25,26]．Algorithm 3 に生成アルゴリズムを示した．

3K グラフは，図 1 に示されるような部分グラフの分布 $P_\wedge(k_1, k_2, k_3)$，$P_\triangle(k_1, k_2, k_3)$ を固定する．

d が増えると部分グラフの種類が増える．図 2 に示すように，ノード数 n を用いて nK グラフを定義すると，これは模倣したいネットワークのグラフ G と一致する．

Priya らは，dK-randomizing rewiring と呼ばれる，エッジを各 dK で特徴付けられる範囲でランダムに交換する方法を用いて，もとのグラフの統計量と比較している．$d = 2$ でほとんどの統計量を，$d = 3$ ではほぼ忠実に全ての統計量を再現できることが示されており，一方 d が増えるにしたがって dK グラフ生成にかかる計算の複雑さが急速に増大することも分かっている [9].

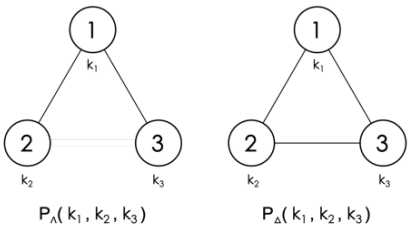


図 1　3 ノードの連結な部分グラフ

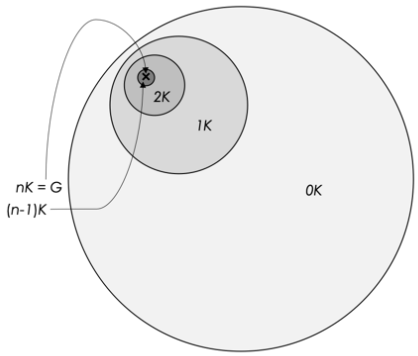


図 2　dK-series の階層

また，確率分布を入力としてそれを満たすグラフを生成することは，3K 以上では現実的に困難である [9].

2.5K グラフ

Gjoka らは，大規模で全体のトポロジが不明なソーシャルネットワークのようなグラフ構造において，サンプリングによって高精度に推定できる統計量を入力としてグラフを生成し，推定が難しい統計量までも再現する試みを提案した．

Gjoka らは dK-series に従い，2K グラフにおける入力である JDD に加え，平均クラスタ係数 $\bar{c}$ も固定した 2.25K グラフ，また，次数に依存するクラスタ係数 $\bar{c}(k)$ もほとんど固定した 2.5K グラフの生成モデルを示し，精度に関して 2K グラフと比較した．また，高精度に JDD とクラスタ係数を推定する方法も提案した [8,27].

2.5K グラフは，1) 2K グラフの生成アルゴリズムにおいて全てのノードに位置を示す数を割り当て，全てのノードペアをその近さによってソートすることで三角形の多いグラフを生成し，2) 同じ次数を持つ 2 つのノードが持つ，異なる 2 本のエッジを JDD を満たしたまま張り替えることでクラスタ係数を減らす，という 2 段階によって生成される．

ここでは 2.5K グラフが，生成の際に固定した統計量だけでなく，極大クリーク分布以外の幅広い主要な統計量について目標とするグラフと似ることが示されている．しかし，ランダム

グラフとしての性質上，生成ごとに異なるグラフが生成されるが，それらがどの程度のばらつきを持ち，一度生成されたものがどの程度信頼できるものなのかについては言及されていない．

クラスタ係数固定のランダムグラフ

JDD と次数に依存するクラスタ係数 $\bar{c}(k)$ を固定する 2.5K グラフに対して，クラスタ係数を固定するグラフ生成モデルについての研究も多くなされている．

Serrano ら [28] は，大きくないクラスタ係数について，次数に依存するクラスタ係数 $\bar{c}(k)$ と次数分布 $P(d_i = k)$ を満たすランダムグラフの生成モデルを提案した．

また，Wang [29] によれば，与えられたクラスタ係数を満たすグラフ生成モデルは Bansal ら [30], Newman [31], Gleeson [32] などによっても提案されている．

Bansal らは平均クラスタ係数 $\bar{c}$ を，Newman は一つのエッジが複数の三角形の生成に使われない限りノードそれぞれのクラスタ係数 c_i を，それぞれ満たすグラフを生成するモデルを提案し，Gleeson は Newman の生成モデルの制約に着目し改善したモデルを提案した．Bansal らの生成モデル以外は，次数分布を同時に満たすことができない．

3 生成グラフの誤差とそのばらつき

実際のネットワークのデータセットから計算した統計量を入力として，dK-series に基づく生成モデルを用いてランダムグラフを複数生成し，生成されたグラフの様々な統計量についてデータセットと比較する．

3.1 実験準備

本実験では，トポロジの異なる表 2 の 3 つのデータセットを使用する．全てのデータセットを無向グラフとして扱う．

0K グラフは，Python のネットワーク・グラフ解析のためのライブラリ NetworkX における関数 `gnm_random_graph`() [33] を使用して生成する．1K グラフは，Algorithm 2 の通り実装したアルゴリズムを用いて生成する．2K グラフは，Gjoka らによるソフトウェア [34] における `construct_simple_2K`() を用いて生成する．2.5K グラフは，Gjoka らによるソフトウェア [34] における `construct_triangles_2K`(), `mcmc_improved_2_5_K`() を用いて生成する．

生成されたグラフと元のグラフの統計量の比較指標として，2 つの離散的な分布の誤差を比較する，NMAE（Normalized Mean Absolute Error）を用いる．NMAE の定義は以下の通りである．

$$\mathrm{NMAE}(\widehat{\vec{x}}, \vec{x}) = \frac{\sum(|\widehat{x_i} - x_i|)}{\sum x_i}$$

ここで，$\widehat{\vec{x}}$ は入力に用いたデータセットの統計量の離散分布，$\vec{x}$ は生成されたグラフの統計量の離散分布に対応する．

また，本実験は表 3 に表す通りの環境で実行した．

3.2 実験手順

各データセットに対し，以下の実験を行う．

（1） ノード数，エッジ数，次数分布，JDD，次数に依存するクラスタ係数の分布を計算する．

（2） それぞれのランダムグラフ生成モデルを用いて 100 ずつグラフを生成する．

- ノード数とエッジ数を入力として 0K グラフを 100 生成する．
- 次数分布を入力として 1K グラフを 100 生成する．
- JDD を入力として 2K グラフを 100 生成する．
- JDD と次数に依存するクラスタ係数の分布を入力として 2.5K グラフを 100 生成する．

（3） 生成されたグラフに対して，以下の統計量について元のデータセットとの誤差 NMAE を計算する．

- 最短距離分布
- 極大クリーク分布
- サイクル分布
- ラプラシアン行列の固有値上位 20 個の分布（スペクトル）
- 近接中心性

（4） 誤差の分布について考察する．

表 2 データセット

Dataset	ノード数 n	エッジ数 m	平均次数 $\bar{d}$	平均クラスタ係数 $\bar{c}$
Caltech [35]	769	16 656	21.65	0.409
Rice [35]	4 087	184 828	45.22	0.294
wiki-Vote [36]	7 115	100 762	14.16	0.141

表 3 実験環境

OS	macOS Mojave 10.14.5
プロセッサ	2.7 GHz Intel® Core™ i7
メモリ	16 GB 2133 MHz LPDDR3
言語	Python 2.7.10（2K，2.5K グラフの生成） Python 3.6.3（その他のグラフの生成）

3.3 実験結果

統計量の分布，誤差の分布，およびグラフ生成にかかる実行時間について評価する．

3.3.1 統計量の分布

Caltech のデータセットとそこから生成されたグラフについて，上記 5 つの統計量の分布をプロットしたものを図 3 に示す．

それぞれの図中の赤線が目標とするグラフの統計量の分布をプロットしたものであり，他が生成された 100 グラフの統計量の分布をプロットしたものである．統計量によっては，dK の d が増えるほど，赤線の分布に近づく傾向があることが分かる．

3.3.2 誤差の分布

Caltech，Rice，wiki-Vote の 3 つのデータセットから生成されたグラフそれぞれについて，上記 5 つの統計量の誤差の分布をプロットしたものを図 4，5，6 に示す．

どの統計量でも，誤差のばらつきは生成モデルによらず小さく，特に 0K はどの統計量についても非常にばらつきが小さいと言える．

最短距離分布，スペクトル，近接中心性は特に各 dK でばらつきが安定して小さく，d が大きくなるにつれて誤差も小さくなる傾向があることが分かる．

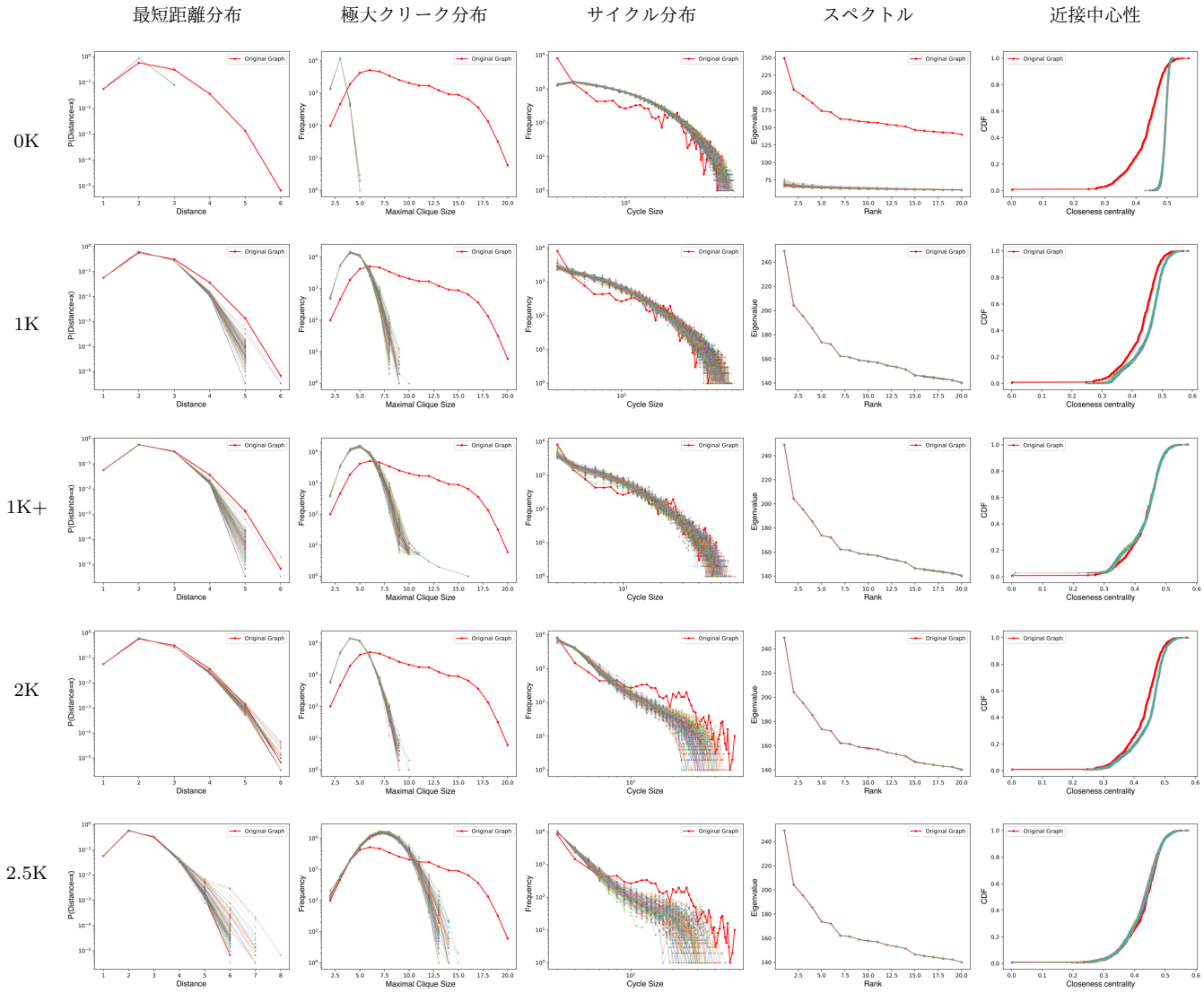


図 3　Caltech の統計量分布

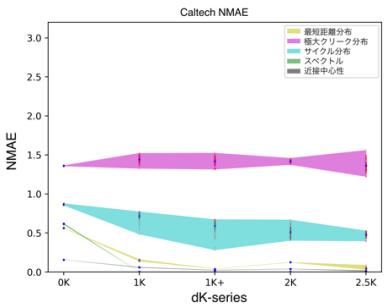


図 4　Caltech の誤差の分布

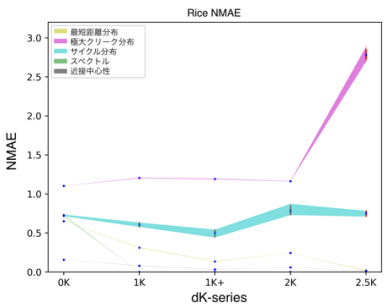


図 5　Rice の誤差の分布

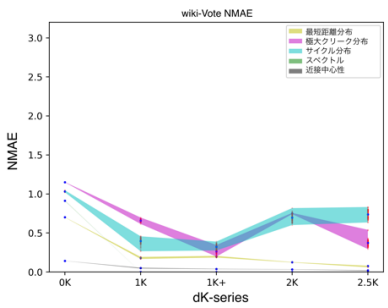


図 6　wiki-Vote の誤差の分布

極大クリーク分布やサイクル分布の誤差とそのばらつきは，他の統計量に比べると大きく，分布もデータセットのトポロジによって異なる．

3.3.3　実 行 時 間

それぞれの生成モデルで 1 グラフを生成するのにかかった実行時間は表 4 の通りである．それぞれ 10 個のグラフを生成し，平均値をとった．0K，1K は Python 3.6.3，2K と 2.5K は Python 2.7.10 で実行した．

表 4　実行時間 [秒]

Dataset	0K	1K	1K+	2K	2.5K
Caltech [35]	0.0658	1.17	1.13	0.532	11.3
Rice [35]	0.629	64.0	53.1	5.09	84.7
wiki-Vote [36]	0.382	47.2	42.5	3.29	62.7

0K グラフは安定して速く生成することができる．2.5K グラフの生成はこの中でもっとも時間がかかる．

4 グラフ生成モデル 1K+ の提案

本研究では，次数分布を固定し，次数に依存するクラスタ係数を 1K よりも入力に近づける生成モデルを提案する．この生成モデルでは，図 3 から分かる通り，従来のランダムグラフ生成モデルによって生成されるグラフが現実のグラフと比べて大きなサイズのクリークを持たない傾向にある事実から，クリークに着目し，1) 次数 k に対する三角形の数に応じてクリークを生成し，2) それぞれのノードの次数を通常のコンフィグレーションモデルを用いて一致させる，という 2 段階によってグラフを生成する．本稿では，この生成モデルによって生成されるグラフを 1K+ グラフと呼ぶことにする．

l 個のノードを使ってサイズ l のクリークを生成したとき，各ノードが $\frac{(l-1)(l-2)}{2}$ 個の三角形に参加するため，全体で数え上げられる三角形の数は，$l \times \frac{(l-1)(l-2)}{2} = \frac{l(l-1)(l-2)}{2}$ 個となる．サイズ 5 のクリークでは，各ノードが $\frac{(5-1)(5-2)}{2} = 6$ 個の三角形に参加しているため，全体で $6 \times 5 = 30$ 個の三角形が数え上げられることになる．

提案する生成モデルでは，各次数に対し，目標とする三角形の数にもっとも近い数の三角形を構成できるクリークを，同じ次数をもつノードどうしで生成するというアプローチをとる．

Algorithm 1 に実装のアルゴリズムを示す．入力は，次数分布 $P(d_i = k)$ から割り当てられた次数列 $d_1 \cdots d_n$ と，次数に依存するクラスタ係数 $\bar{c}(k)$ から計算された，次数 k のノードが参加する三角形の数の合計 $ntri[k]$ とする．

この 1K+ アルゴリズムによって生成されたグラフは，クラスタ係数を入力としない 0K，1K，2K のグラフ生成モデルによって生成されたグラフと比べ，目標とするグラフの次数に依存するクラスタ係数に近づく．図 7 に，生成されたグラフの次数に依存するクラスタ係数の分布を示す．図中の赤い点が目標とするグラフの分布をプロットしたものであり，他が生成された 100 グラフの分布をプロットしたものである．1K+ アルゴリズムによって生成されたグラフの次数に依存するクラスタ係数の分布が，0K，1K，2K のグラフ生成モデルによって生成されたグラフと比べ赤い点によく重なっていることがわかる．

Algorithm 1 1K+ グラフ生成のアルゴリズム

Require: ノードの次数列 $d_1 \cdots d_n$，次数 k のノードが参加する三角形の数の合計 $ntri[k]$
Ensure: 1K+ グラフ
  $V' \leftarrow \{1, \cdots n\}$
  $E' \leftarrow$ 空集合
  $V'_k \leftarrow d_i = k$ であるノードの ID i
  for all 次数 k **do**
    if $|V'_k| < k + 1$ **then**
      if $ntri[k] <$ サイズ $|V'_k|$ のクリークで生成できる三角形の数 **then**
        $ntri[k]$ にもっとも近い数の三角形を生成できるクリークを V'_k 中のノードで生成
        E' を更新
      else
        サイズ $|V'_k|$ のクリークを V'_k 中のノードを全て用いて生成
        E' を更新
      end if
    else
      if $|V'_k| \geqq$ 次数 k のクリークで生成できる三角形の数 **then**
        サイズ $k+1$ のクリークを V'_k 中のノードで生成
        E' を更新
      else
        ntri[k] にもっとも近い数の三角形を生成できるクリークを V'_k 中のノードで生成
        E' を更新
      end if
    end if
  end for
  コンフィグレーションモデル（Algorithm 2）で次数を合わせる
  $G' \leftarrow (V', E')$
  return G'

4.1 実　　験

第 3 章での実験と同様のことを 1K+生成モデルに対しても行う．次数分布と次数に依存するクラスタ係数の分布を入力として，Algorithm 1 の通り Python 3.6.3 で実装した 1K+-グラフ生成モデルでグラフを 100 生成する．

4.2 実 験 結 果

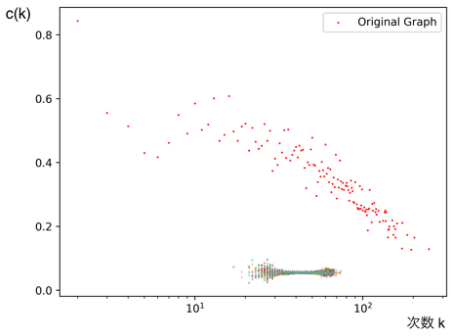
Caltech のデータセットの統計量の分布は，図 3 に示す．図 4，5，6 から分かる通り，提案した 1K+ のアルゴリズムによって生成されるグラフは，2K よりも誤差が小さくなることがあり，例えば最短距離分布に関して，Caltech のグラフではもっとも誤差が小さく，Rice でも 2.5K の次に誤差が小さい．また，1K+ グラフの極大クリーク分布とサイクル分布の誤差は，Wiki-Vote のグラフでは平均的にもっとも小さくなっている．Caltech や Rice では 1K から多少改善されている．さらに，実行時間について，表 4 から分かる通り，1K+ グラフは，1K グラフより生成に必要な入力が増えているが，より短い時間で生成できる．

5 まとめと今後の課題

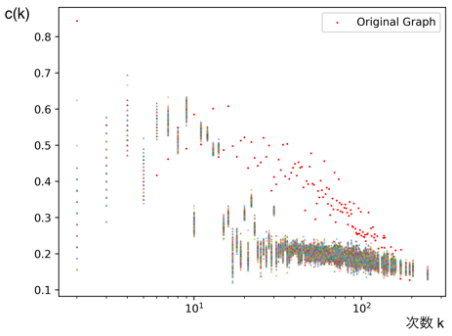
本研究では，dK-series に基づくグラフ生成モデルによって生成されるグラフの各統計量がどの程度ばらつくかについて検証した．また，それぞれの生成モデルによって生成されたグラフの統計量の誤差の分布を考察した．これにより，dK-series に基づくグラフ生成モデルで入力として与えた統計量以外の統計量について，誤差のばらつきが小さいことを示した．

さらに，次数に依存するクラスタ係数を入力として，クリークに着目し三角形を構築する 1K+ グラフの生成モデルを提案し，これが 2.5K グラフの生成より実行時間を短縮でき，また，トポロジによっては 2K グラフよりも統計量の誤差を小さくできることを示した．

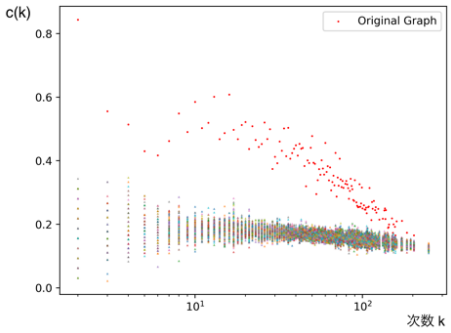
今後の課題は，1K+ グラフの生成モデルを，入力として与えていた次数に依存するクラスタ係数により近づける改善を行



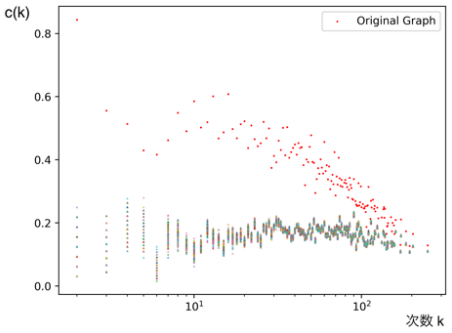
(a) 0K



(c) 1K+



(b) 1K



(d) 2K

図 7　次数に依存するクラスタ係数の分布（Caltech）

い，極大クリークの分布の誤差をはじめとする統計量の誤差も安定してより小さくすることである．

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．

文　献

[1] Kwak, H., Lee, C., Park, H. and Moon, S. B.: What is Twitter, a social network or a news media?, *Proceedings of the 19th international conference on World Wide Web*, pp. 591–600 (2010).

[2] Ferrara, E.: *Measurement and Analysis of Online Social Networks Systems*, pp. 891–893 (2014).

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

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

[5] Hu, P. and Lau, W. C.: A Survey and Taxonomy of Graph Sampling, *arXiv preprint arXiv:1308.5865* (2013).

[6] Gjoka, M., Kurant, M., Butts, C. T. and Markopoulou, A.: Walking in Facebook: A Case Study of Unbiased Sampling of OSNs, *2010 Proceedings IEEE INFOCOM*, pp. 1–9 (2010).

[7] 中嶋一貴, 首藤一幸: プライベートなノードを含むソーシャルネットワークの統計量推定, DEIM2019 第 11 回データ工学と情報マネジメントに関するフォーラム (2019).

[8] Gjoka, M., Kurant, M. and Markopoulou, A.: 2.5 k-graphs: from sampling to generation, *2013 Proceedings IEEE INFOCOM*, pp. 1968–1976 (2013).

[9] Mahadevan, P., Krioukov, D., Fall, K. and Vahdat, A.: Systematic topology analysis and generation using degree correlations, *ACM SIGCOMM Computer Communication Review*, Vol. 36, No. 4, pp. 135–146 (2006).

[10] Watts, D. J. and Strogatz, S. H.: Collective dynamics of 'small-world' networks, *Nature*, Vol. 393, No. 6684, pp. 440–442 (1998).

[11] Pusch, A., Weber, S. and Porto, M.: Generating random networks with given degree-degree correlations and degree-dependent clustering, *Physical review. E*, Vol. 77, No. 1, p. 017101 (2008).

[12] Erdős, P. and Rényi, A.: On Random Graphs I, *Publicationes Mathematicae Debrecen*, Vol. 6, p. 290 (1959).

[13] Ma, J., van den Driessche, P. and Willeboordse, F. H.: Effective degree household network disease model, *Journal of Mathematical Biology*, Vol. 66, No. 1, pp. 75–94 (2013).

[14] Bender, E. A. and Canfield, E. R.: The Asymptotic Number of Labeled Graphs with Given Degree Sequences, *Journal of Combinatorial Theory, Series A*, Vol. 24, No. 3, pp. 296–307 (1978).

[15] Bollobás, B.: A Probabilistic Proof of an Asymptotic Formula for the Number of Labelled Regular Graphs, *European Journal of Combinatorics*, Vol. 1, No. 4, pp. 311–316 (1980).

[16] Chung, F. and Lu, L.: The Average Distance in a Random Graph with Given Expected Degree, *Internet Mathematics*, Vol. 1, No. 1, pp. 91–113 (2004).

[17] Chung, F. and Lu, L.: Connected Components in Random Graphs with Given Expected Degree Sequences, *Annals of Combinatorics*, Vol. 6, No. 2, pp. 125–145 (2002).

[18] Luczak, T.: Sparse random graphs with a given degree sequence, *Proceedings of the Symposium on Random Graphs, Poznan*, pp. 165–182 (1989).

[19] Molloy, M. and Reed, B. A.: A Critical Point for Random

Graphs with a Given Degree Sequence, *Random structures & algorithms*, Vol. 6, No. 2-3, pp. 161–180 (1995).
[20] Molloy, M. and Reed, B. A.: The Size of the Giant Component of a Random Graph with a Given Degree Sequence, *Combinatorics, Probability & Computing*, Vol. 7, No. 3, pp. 295–305 (1998).
[21] C. Wormald, N.: The asymptotic connectivity of labelled regular graphs, *Journal of Combinatorial Theory, Series B*, Vol. 31, No. 2, pp. 156–167 (1981).
[22] E.J. Newman, M., H. Strogatz, S. and Watts, D. J.: Random Graphs with Arbitrary Degree Distributions and their Applications, *Physical review. E*, Vol. 64, No. 2, p. 026118 (2001).
[23] Durrett, R.: *Random Graph Dynamics*, Cambridge Series in Statistical and Probabilistic Mathematics (2006).
[24] I Del Genio, C., Kim, H., Toroczkai, Z. and Bassler, K.: Efficient and Exact Sampling of Simple Graphs with Given Arbitrary Degree Sequence, *PloS one*, Vol. 5, No. 4, p. e10012 (2010).
[25] Stanton, I. and Pinar, A.: Constructing and Sampling Graphs with a Prescribed Joint Degree Distribution, *Journal of Experimental Algorithmics*, Vol. 17, pp. 3–1 (2012).
[26] Stanton, I. and Pinar, A.: Sampling Graphs with a Prescribed Joint Degree Distribution Using Markov Chains, *Proceedings of the 13th Workshop on Algorithm Engineering and Experiments*, pp. 151–163 (2011).
[27] Tillman, B., Markopoulou, A., Gjoka, M. and Buttsc, C. T.: 2K+ Graph Construction Framework: Targeting Joint Degree Matrix and Beyond, *IEEE/ACM Transactions on Networking*, Vol. 27, No. 2, pp. 591–606 (2019).
[28] Serrano, M. A. and Boguná, M.: Tuning clustering in random networks with arbitrary degree distributions, *Physical review. E*, Vol. 72, p. 036133 (2005).
[29] Wang, C., Lizardo, O. and Hachen, D.: Algorithms for generating large-scale clustered random graphs, *Network Science*, Vol. 2, No. 3, pp. 403–415 (2014).
[30] Bansal, S., Khandelwal, S. and Meyers, L. A.: Exploring biological network structure with clustered random networks, *BMC Bioinformatics*, Vol. 10, No. 1, p. 405 (2009).
[31] Newman, M. E.: Random Graphs with Clustering, *Physical review letters*, Vol. 103, p. 058701 (2009).
[32] Gleeson, J. P.: Bond percolation on a class of clustered random networks, *Physical review. E*, Vol. 80, No. 3, p. 036107 (2009).
[33] gnm_random_graph – NetworkX 1.10 documentation., `https://networkx.github.io/documentation/networkx-1.10/reference/generated/networkx.generators.random_graphs.gnm_random_graph.html` (Last accessed on 12 February 2020).
[34] 2.5K Generator Source Code., `http://www.minasgjoka.com/2.5K/instructions/` (Last accessed on 12 February 2020).
[35] Index of /2.5K/graphs. Minas Gjoka's personal website., `http://www.minasgjoka.com/2.5K/graphs/` (Last accessed on 12 February 2020).
[36] Stanford Large Network Dataset Collection., `https://snap.stanford.edu/data/` (Last accessed on 12 February 2020).

付　　録

コンフィグレーションモデル（1K グラフ生成）のアルゴリズム

Algorithm 2 に示す．入力の分布によってはそれをちょうど満たすグラフを生成できない可能性があるが，簡単のために，ここではグラフを生成可能な入力が与えられるものとする．

Algorithm 2 1K グラフ生成のアルゴリズム

Require: ノードの次数列 $d_1 \cdots d_n$
Ensure: 1K グラフ
  $k_{rem} \leftarrow$ 残り次数を入れる長さ n の配列 $\{d_1 \cdots d_n\}$
  $k_{sum} \leftarrow$ 次数和 $\sum_{i=1}^{n} d_i$
  $V' \leftarrow \{1, \cdots n\}$
  $E' \leftarrow$ 空集合
  $t \leftarrow$ 反復回数，初期値 0
  $t_{max} \leftarrow$ 最大反復回数（任意）
  while $k_{sum} > 0$ かつ $t < t_{max}$ **do**
    $k_{max} \leftarrow$ 残り次数の最大値
    $v_s \leftarrow d_s = k_{max}$ となるノード
    $v_e \leftarrow$ 残り次数 $k_{rem}[v_e]$ に比例する確率でランダムに選ばれたノード
    if v_s と v_e の間にエッジがない **then**
      $E' \leftarrow E' \cup (v_s, v_e)$
      $k_{rem}[v_s] \leftarrow k_{rem}[v_s] - 1$
      $k_{rem}[v_e] \leftarrow k_{rem}[v_e] - 1$
      $k_{sum} \leftarrow k_{sum} - 2$
    end if
    $t \leftarrow t + 1$
  end while
  $G' \leftarrow (V', E')$
  return G'

Algorithm 3 2K グラフ生成のアルゴリズム

Require: 次数 k のノードと次数 l のノードの間にあるエッジの本数 $JDD(k, l)$，ノードの次数列 $d_1 \cdots d_n$
Ensure: 2K グラフ
  $V' \leftarrow \{1, \cdots n\}$
  $E' \leftarrow$ 空集合
  $E'_{possible} \leftarrow$ 全てのノードペア $\{(v_i, v_j) | 1 \leqq i, j \leqq n, i \neq j\}$
  $JDD'(d_i, d_j) \leftarrow 0$
  $d'_1 \cdots d'_n \leftarrow 0$
  for all $(v_i, v_j) \in E'_{possible}$ **do**
    if $JDD'(d_i, d_j) < JDD(d_i, d_j)$ かつ $d'_i < d_i$ かつ $d'_j < d_j$ **then**
      $E' \leftarrow E' \cup (v_i, v_j)$
      $d'_i \leftarrow d'_i + 1$
      $d'_j \leftarrow d'_j + 1$
      $JDD'(d_i, d_j) \leftarrow JDD'(d_i, d_j) + 2$
    end if
  end for
  $G' \leftarrow (V', E')$
  return G'

2K グラフ生成のアルゴリズム

Algorithm 3 に示す．これも同様に，入力の分布によってはそれをちょうど満たすグラフを生成できない可能性があるが，簡単のために，ここではグラフを生成可能な入力が与えられるものとする．