

分散スケッチの重複測定の回避法とスケッチ配置法

キム ウヒョン[†] 上山 憲昭[†]

[†] 立命館大学 情報理工学部 〒567-8570 大阪府茨木市岩倉町 2-150

E-mail: [†]is0592vi@ed.ritsumei.ac.jp, ^{††}kamiaki@fc.ritsumei.ac.jp

あらまし コンテンツの多様化に伴いネットワークの規模とトラフィックが増大する中、限られたメモリ資源で効率的かつ高精度なトラフィック測定が求められている。スケッチアルゴリズムは、フロー情報を小規模なデータ構造に要約し、低コストで近似的な測定を可能とする手法であり、近年注目されている。一方、分散型スケッチでは重複測定やスケッチノード配置の最適化が課題となる。本稿では、媒介中心性に基づくスケッチノード配置方式と、重複測定を回避する方式を提案し、計算機シミュレーションにより有効性を評価する。

キーワード スケッチ, 重複測定, フラグ, 媒介中心性

Methods for Duplicate Measurement Avoidance and Sketch Placement in Distributed Sketching

Woohyeon KIM[†] and Noriaki KAMIYAMA[†]

[†] College of Information Science and Engineering, Ritsumeikan University

2-150, Iwakura-cho, Ibaraki, Osaka 567-8570, JAPAN

E-mail: [†]is0592vi@ed.ritsumei.ac.jp, ^{††}kamiaki@fc.ritsumei.ac.jp

Abstract As the development of content continues to expand network scale and traffic volume, there is an increasing demand for efficient and highly accurate traffic measurement under limited memory resources. Sketch algorithms have recently attracted significant attention as a low-cost approach that summarizes flow information into compact data structures, enabling approximate traffic measurement. However, in distributed environments, challenges such as redundant measurements and the optimization of sketch node placement arise. In this paper, we propose a sketch node placement scheme based on betweenness centrality, along with a method to avoid redundant measurements, and evaluate their effectiveness through simulations.

Key words Sketch, Redundant Measurement, Flag, Betweenness Centrality

1. はじめに

近年、ネットワークの規模と複雑性の増大に伴い、効率的かつ正確なトラフィック測定の重要性が高まっている。従来のトラフィック観測手法では、測定精度を高めると大規模な計算コストやメモリコストが発生し、一方でサンプリング等によりコストを削減すると測定精度が低下するという課題が存在する。このような課題を解決する手段として、スケッチアルゴリズムが注目されている。スケッチは、観測対象のデータをリアルタイムで集計しつつ、メモリ使用量を抑え、必要に応じて統計量を推定することができる。大規模ネットワークにおけるトラフィック測定では、スケッチを用いることで効率的かつ正確な測定が可能となる。

このようなスケッチは、中央集散的にスケッチノードを設置する方式と、分散型でスケッチノードを配置する方式が存在する。中央集散的に設置した場合、実装が簡単で、精度も高くなるが、全てのフローが限られたノードに集中するため、ノード

の負荷が大きくなる。一方、分散型は負荷の分散が可能であるが、いくつかの課題が存在する。各ノードが中央の制御を受けないため、同一フローが複数のスケッチノードで重複して測定される問題である。重複測定は推定精度の低下につながる。さらに、スケッチノードをどの位置に設置するかによって、測定効率に大きな差が生じる問題も存在する。

これらの課題を解決するために、本稿では、(1) パケットヘッダにフラグを設定することで重複測定を回避する方式と、(2) 媒介中心性に基づく段階的なスケッチ配置方式、の2つの方式を提案する。本方式により、ネットワーク全体で効率的かつ正確なトラフィック測定を実現し、完全自律分散による効率的なスケッチ測定環境の実現を目指す。また計算機シミュレーションより、提案方式の有効性を確認する。

2. スケッチアルゴリズム

2.1 スケッチの概要

スケッチアルゴリズムとは、大規模なデータストリームを対

象として、すべてのデータを保存することなく、限られたメモリ資源の下で統計量を近似的に推定するための確率的データ構造である。スケッチでは、観測対象となる各データを逐次的に処理し、ハッシュ関数などを用いて内部の要約情報を更新することで、データの到着順序や総数に依存せずに処理を行うことができる。

スケッチアルゴリズムには、頻度推定、ヘビーヒッタ検出、分位点推定など、目的に応じてさまざまな手法が提案されており、求める統計量に応じて適切なアルゴリズムを選択する必要がある。

2.2 Count-Min Sketch

Count-Min Sketch (CMS) は、データストリーム中に出現する要素の頻度を、限られたメモリ資源のもとで近似的に推定するための代表的なスケッチアルゴリズムである。CMS では、各要素 x の実際の出現頻度を $f(x)$ 、推定された頻度を $\hat{f}(x)$ 、ストリーム全体に含まれる要素数の総和を N とすると、式 (1) で誤差保証が理論的に与えられる。

$$f(x) \leq \hat{f}(x) \leq f(x) + \varepsilon N, 1 - \delta \text{ の確率で保証される} \quad (1)$$

ここで ε は許容誤差率、 δ は失敗確率を表す。式 (1) が示すように、CMS による推定値は常に実際の頻度以上となり、確率 $1 - \delta$ で最大誤差が εN に抑えられることが保証される。

CMS では、あらかじめ許容誤差率 ε と失敗確率 δ を設定することで、必要なメモリサイズが理論式に基づいて決定される。すなわち幅 w 、深さ d を持つ二次元配列によって構成され、それぞれ式 (2) と式 (3) で与えられる。

$$w = \left\lceil \frac{e}{\varepsilon} \right\rceil \quad (2)$$

$$d = \left\lceil \log \frac{1}{\delta} \right\rceil \quad (3)$$

ここで、 w は各行に配置されるカウンタ数（頻度を記録する配列要素）、 d は使用する独立なハッシュ関数の個数に対応する。誤差率 ε を小さく設定するほど幅 w が増大し、失敗確率 δ を小さく設定するほど深さ d が増大するため、推定精度とメモリ使用量との間にはトレードオフの関係が存在する。

CMS では、 d 個の独立なハッシュ関数 h_1, h_2, \dots, h_d を用いてデータを処理する。データストリーム中で要素 x が観測された場合、各ハッシュ関数に対応するカウンタを式 (4) のように更新する。

$$\forall i \in \{1, \dots, d\}, C[i][h_i(x)] \leftarrow C[i][h_i(x)] + 1 \quad (4)$$

この処理により、CMS はデータを保存することなく、到着した要素を逐次的にカウントすることが可能となる [1]。

3. 関連研究

3.1 スケッチの配置問題

ネットワーク測定において、スケッチをどのノードに配置するかは測定精度や効率に大きく影響する重要な課題である。Chen らは、スケッチ配置問題に対して主に 2 種類のアプローチが存在することを示している [3]。一つは MILP (mixed-integer linear programming) による完全最適化、もう一つはヒューリスティックによる近似最適化である。

MILP を用いる手法では、全てのフロー状況やネットワークポロジの組み合わせを計算することで理論上は最適配置を得ることができる。しかし、実際のネットワークではフローや負荷が動的に変化するため、計算コストが非常に高く、トポロジの変化による拡張が困難である。一方、ヒューリスティック手

法は計算コストを抑えはでき、拡張も簡単であるが、最適な配置ではない。[3] では、計算コストと負荷分散の両方を考慮した最適に近い配置手法を提案しているが、MILP ほどの最適化は行えず、ヒューリスティックよりは最適な配置の探索に時間がかかるという特徴を持つ。

3.2 重複測定問題

分散型ネットワークにおいて、複数のノードが同一フローを測定することによる重複測定も重要な課題である。Li らは、中央集約的なスケッチ配置ではリソース使用量は少ないものの、特定ノードに負荷が集中する問題があることを指摘している [2]。これに対し、協力型分散スケッチ配置を提案しており、負荷分散と共に重複測定を削減している。しかし、この手法はフロー傾向を事前に解析し、セグメント単位でスケッチを配置する設計であるため、リアルタイムで変化するネットワーク環境での適用は難しい。

3.3 既存研究の限界

既存研究から、以下の課題が明らかである。

(1) スケッチ配置の最適化と計算コストのトレードオフ：MILP は理論上の最適化度が高いが計算コストが極めて高く、拡張するには困難である。一方、ヒューリスティックは計算コストは低く拡張にも容易であるが、最適化は MILP に劣る。

(2) 分散型における重複測定：事前にフローを解釈することで重複測定の削減はできたものの、リアルタイム変動に弱く、重複測定を完全に除去できたわけではない。

これらの課題を解決するため、本稿では新たな配置方法として媒介中心性に基づくスケッチノード配置方式と、フラグを用いた重複測定の完全排除方式の 2 つの方式を提案し、分散環境でも高精度かつ効率的な測定を目指す。

4. 提案方式

4.1 重複測定の回避（フラグ設定）

分散型スケッチ配置環境では、同一フローが複数のスケッチノードを通過する可能性があり、重複測定が発生する恐れがある。重複測定を回避するために、パケットのヘッダに 1 ビットのフラグを付加することで、重複測定を回避する方式を提案する。このフラグは、当該パケットが既にスケッチによって測定されたか否かを示すものである。初期状態では、全てのパケットのフラグ値は 0 に設定されている。フラグ値が 0 のパケットがスケッチノードを通過した場合、当該パケットは測定対象となり、スケッチ処理が行われると同時に、フラグ値は 1 に更新される。一方、フラグ値が 1 のパケットが他のスケッチノードを通過した場合、当該パケットは既に測定済みであると判断され、スケッチ処理は行われず、そのまま転送される。

この方式により、同一フローに属するパケットが複数のスケッチノードで重複して測定されることを防ぐことができる。具体的には、提案方式におけるフラグ情報を IPv4 ヘッダの Options フィールドに格納する。IPv4 の Options フィールドは Type-Length-Value (TLV) 形式で構成されており、Option Type、Length フィールドは 1 バイト、Value は 2 バイトである。Length フィールドを 4 と設定することで、Option 全体の長さを 4 バイトとし、IPv4 ヘッダの 32 ビット境界に整合させている。Value 部は 2 バイトで構成され、先頭バイトの 1 ビットを「当該パケットが既に測定済みであること」を示すフラグとして利用する。残りのビットおよび後続の 1 バイトはパディングとして 0 に設定する。フラグ情報を格納した IPv4 ヘッダの形式を図 1 に示す。

	4	8	16	19	24	31
Version	Header Length	Type of Service	Total Packet Length			
Identifier			Flags	Fragment Offset		
Time to Live		Protocol ID	Header Checksum			
Source IP Address						
Destination IP Address						
IP Header Options (Sketch Flag)						

図 1: 提案方式におけるフラグ情報を含む IPv4 ヘッダ形式

また、IPv6 においても同様にフラグ情報をヘッダ内に格納することが可能である。IPv6 では IPv4 の Options フィールドに相当する機構として、Extension Header が定義されており、本稿では Hop-by-Hop Options Header を利用する。Hop-by-Hop Options Header も TLV 形式で構成されており、Option Type, Opt Data Len および Option Data からなる。Option Data 部の 1 ビットをフラグとして利用し、残りのビットはパディングとして 0 で埋める。フラグ情報を格納した IPv6 ヘッダの形式を図 2 に示す。

0	4	12	16	19	24	31
Version	Traffic Class	Flow Label				
Payload Length			Next Header	Hop limit		
Source Address						
Destination IP Address						
Extension headers (Hop-by-Hop Options Header)						

図 2: 提案方式におけるフラグ情報を含む IPv6 ヘッダ形式

4.2 効率的な配置（媒介中心性を用いた段階的配置方式）

スケッチノードの配置は測定性能に大きな影響を与える。本稿では、実用的な計算時間で配置順序を決定するため、ヒューリスティックな配置方式を用いる。媒介中心性（betweenness centrality）は、ネットワーク内のノードが他のノード間の最短経路上にどの程度位置しているかを表す指標であり、式 (5) で定義される。

$$BC(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \quad (5)$$

ここで、 σ_{st} はノード s からノード t への最短経路の総数を表し、 $\sigma_{st}(v)$ はそれらのうちノード v を通過する最短経路の数を表す。

単純に媒介中心性の高い順にスケッチノードを配置する方法では、既に測定可能なフローが集中する可能性があり、カバレッジの観点で非効率となる場合がある。そこで本稿では、以下の手順により媒介中心性を段階的に更新する反復的な配置方式を提案する。

- (1) 全フローを対象として媒介中心性を計算
- (2) 最も媒介中心性の高いノードを選択し、最初のスケッチ配置ノードに決定
- (3) 選択されたノードを通過するフローを測定済みとみなし、フロー集合から除外
- (4) 残存するフローのみを対象として、再度媒介中心性を

計算

- (5) 上記の手順を繰り返し、スケッチノードの配置順序を決定

5. 性能評価

5.1 評価条件

本稿では、ネットワークのノード、リンク、人口などの情報として、米国の商用 ISP のバックボーンネットワークである、“allegiance telecom”、“at home network”、“att”、“cais internet”、“verio”の 5 種類のトポロジを使用する。ノード間の距離は、トポロジに含まれる各ノードの経度および緯度情報を用いて計算し、距離に基づいてダイクストラ法を適用することで最短経路を求める。ノード間距離の計算式は式 6 の通りである。

$$d(\mathbf{x}_1, \mathbf{x}_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (6)$$

シミュレーションにおけるトラフィックの発生は、各トポロジに付与された人口データを用い、重力モデルに基づいて各フロー毎に 10 万回行った。本稿ではスケッチを用いたネットワーク測定を対象としているため、使用するスケッチアルゴリズムとして代表的なアルゴリズムである Count-Min Sketch を採用した。CMS の誤差率を 0.001 (0.1%)、失敗確率を 0.01 (1%) に設定し、一つのスケッチノードに必要なメモリ容量は、理論式に基づく計算の結果、約 53KB である。

効率性の評価では、スケッチノード数を一つずつ増加させることでスケッチの比率を段階的に変化させ、ネットワーク全体で発生するフローのうち、どの程度をスケッチによって測定できるかを評価するため、式 (7) で定義されるカバレッジを指標として用いる。ここで C はカバレッジ、 P_{sketch} は測定済みのパケットの数、 P_{total} はネットワークで発生したパケットの数を意味する。

$$C = \frac{P_{\text{sketch}}}{P_{\text{total}}} \quad (7)$$

スケッチの比率は、全ノード数に対するスケッチ設置ノード数の割合として定義し、式 (8) で表される。ここで、 R はスケッチの比率、 N_{sketch} はスケッチ処理ノードの数、 N_{node} はネットワーク全体でのノードの数を意味する。

$$R = \frac{N_{\text{sketch}}}{N_{\text{node}}} \quad (8)$$

効率性の比較対象として、(1) スケッチをランダムに配置した場合（スケッチノードをランダムに配置し、10 回シミュレーションして、平均を求める）、(2) 媒介中心性（betweenness centrality, bc）の高い順に配置した場合、(3) 本稿で提案する方式に基づいて配置した場合の 3 通りの配置方式を用いる。精度の評価には、既存研究 [4] で用いられている評価指標である式 (9) に基づいて算出する。

$$\frac{1}{N} \sum_{i=1}^N |f(x_i) - \hat{f}(x_i)| \quad (9)$$

ここで、 N は評価対象となるフローの総数を表し、 $f(x_i)$ は実際のフロー量、 $\hat{f}(x_i)$ は CMS による推定値である。精度評価においても、スケッチノード数を 1 つずつ増加させた場合の変化を観察し、ランダム配置、BC 順配置、提案方式の 3 通りについて比較を行う。

5.2 効率性の評価

本節では、5 種類の実ネットワークトポロジ（allegiance telecom, at home network, att, cais internet, verio）を用いて、提案手法の測定性能を評価する。評価指標として、スケッチノードの比率（横軸）に対するフローカバレッジ（縦軸）を用い、提案

手法、媒介中心性に基づく配置方式、およびランダム配置方式を比較する。図3～7に、5つの各トポロジにおける各方式のカバレッジを、スケッチ比率に対して各々示す。

図3に示す verio のトポロジの結果では、提案手法はスケッチ比率 0.3 付近においてカバレッジ 1 を達成しており、他の手法を上回る性能を示している。ただし、スケッチの比率 0.2 までの範囲では、媒介中心性に基づく配置方式が提案手法と同等のカバレッジを示している。図4に示す allegiance Telecom のトポロジの結果では、提案手法はスケッチ比率 0.2 付近においてカバレッジ 1 を達成しており、他の手法と比較して高い測定性能を示している。一方で、スケッチの比率 0.1 までの初期段階では、媒介中心性に基づく配置方式が提案手法とほぼ同等のカバレッジを示している。

図5に示す att のトポロジの結果では、提案手法はスケッチ比率 0.3 付近においてカバレッジ 1 を達成しており、ランダム配置方式を大きく上回る結果を示している。ただし、媒介中心性に基づく配置方式が提案手法とほぼ同じカバレッジを表している。図6に示す at home network のトポロジの結果では、提案手法はスケッチ比率 0.3 付近においてカバレッジ 1 を達成しており、他の手法と比較して高い測定性能を示している。注目すべき点として、図3～図5の結果とは異なり、媒介中心性に基づく配置方式がランダム配置方式よりも低い性能を示している。図7に示す cais internet のトポロジの結果では、提案手法はスケッチ比率 0.5 付近においてカバレッジ 1 を達成しており、他の配置方式を上回る結果を示している。注目すべき点として、図6と同様に媒介中心性に基づく配置方式がランダム配置方式よりも低い性能を示している。

以上の結果を総合すると、提案手法はいずれのトポロジにおいても、他の手法と比較して最も早い段階で 100% のカバレッジへ収束しており、特定のネットワーク構造に依存しない安定した性能を有することが確認された。一方、媒介中心性に基づく配置方式は、スケッチ比率の増加に伴い、初期段階におけるカバレッジの立ち上がりは提案手法と同じく急激に増加するものの、一定の数値に達した後からはカバレッジの増加率が低くなる傾向がみられる。これは、媒介中心性がネットワーク構造上の重要性を反映する指標である一方で、フローの通過関係や重複測定の影響を直接考慮していないためであると考えられる。

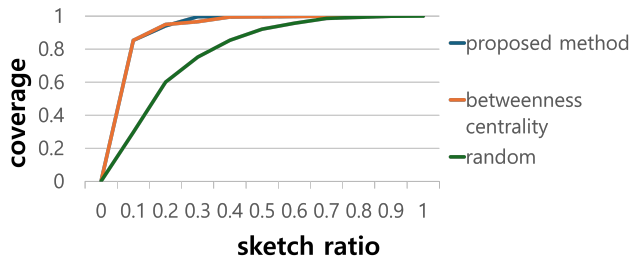


図 3: Verio のカバレッジ

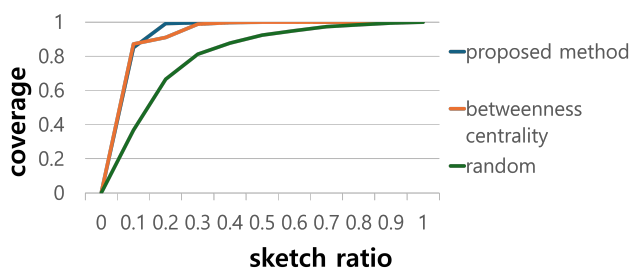


図 4: Allegiance Telecom のカバレッジ

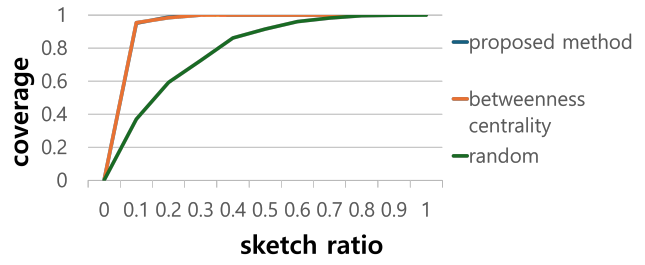


図 5: ATT のカバレッジ

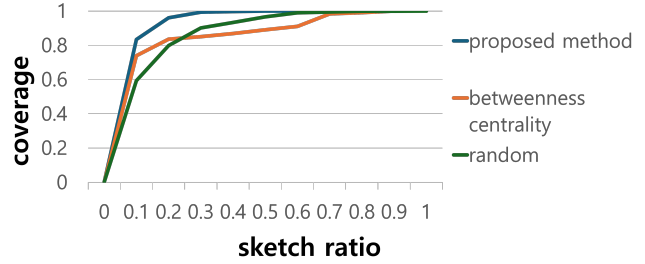


図 6: At Home Network のカバレッジ

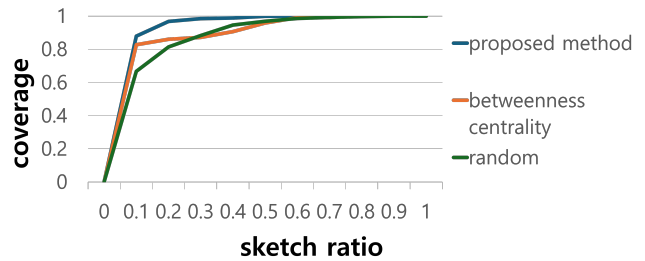


図 7: CAIS Internet のカバレッジ

5.3 精度の評価

本節では、5種類の実ネットワークトポロジを用いて、提案手法の測定精度を評価する。評価指標として、スケッチノード数を 1 からトポロジ内の全ノード数まで段階的に増加させた際の、実際のフローの頻度と CMS による推定値との差の平均（平均誤差）を用いた。図8～12に、5つの各トポロジにおける各方式の平均誤差を、スケッチノード数に対して各々示す。

図8に示す verio トポロジの結果において、提案手法は少数のスケッチノード数の段階から平均誤差がほかの手法と比較して同等または小さく、高い推定精度を達成している。特に、ランダム配置方式と比較して初期段階における誤差の差は最大 280 に達しており、媒介中心性に基づく配置方式よりも少ないスケッチノード数で誤差がほぼ 0 に収束していることが確認できる。図9に示す allegiance telecom トポロジの結果においても、提案手法はスケッチノード数が少ない場合でも他の方式と比較して平均誤差を抑えている。また、媒介中心性に基づく配置方式と比較すると、初期段階において提案手法の方が誤差の収束が早い傾向が見られる。

図10に示す att トポロジの結果では、提案手法と媒介中心性に基づく配置方式がほぼ同程度の平均誤差を示している。しかしながら、提案手法はランダム配置方式と比較して初期段階から誤差が小さく、少数のスケッチノードにおける推定精度の優位性が確認できる。図11に示す at home network トポロジの結果では、提案手法は他の方式と比較して、スケッチノード数の増加に伴う平均誤差の減少が最も早いことが示されている。注目すべき点として、媒介中心性に基づく配置方式がランダム配

置方式より平均誤差が高いことが挙げられる。図 12 に示す cais internet トポロジの結果においても、提案手法は他の方式と比較して、スケッチノード数の増加に伴う平均誤差の減少が最も早いことが示されている。他方式と比較して、媒介中心性に基づく配置方式がランダム配置方式より平均誤差が高いことが特徴的である。

図 8 から図 12 に示す各トポロジにおける結果を見ると、提案手法はトポロジに寄らず、少数のスケッチノードであっても平均誤差が非常に小さく、正確な推定推定が可能であることが確認できる。また、提案手法は att 以外のトポロジでスケッチノード数が増加するにつれて提案手法の平均誤差はほぼ 0 に収束し、全てのトポロジにおいて他方式よりも、ネットワーク全体のフローをより少ないスケッチノードで高い精度を保証し、把握できることが確認された。以上の結果から、提案手法は分散型環境においても CMS によるフロー測定の精度を高く維持できることが確認できる。

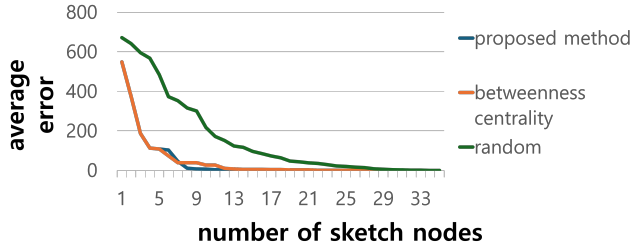


図 8: Verio の平均誤差

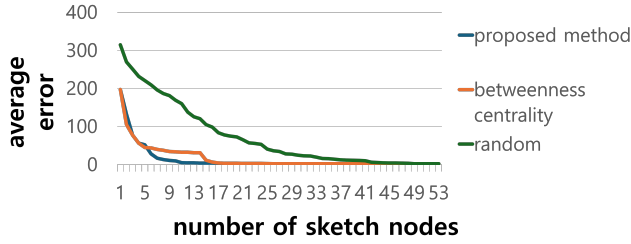


図 9: Allegiance Telecom の平均誤差

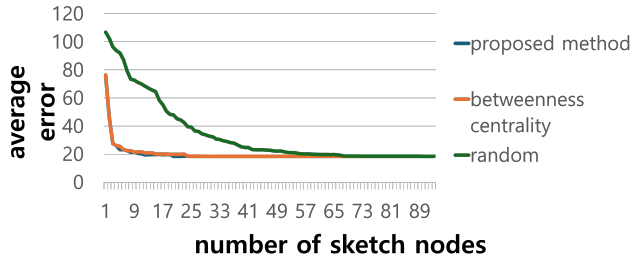


図 10: ATT の平均誤差

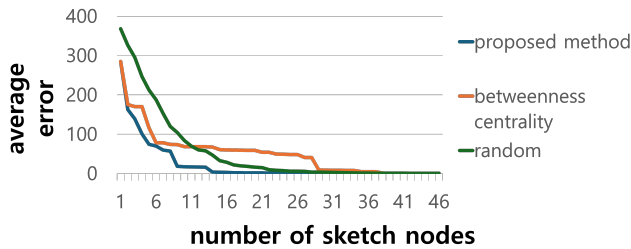


図 11: At Home Network の平均誤差

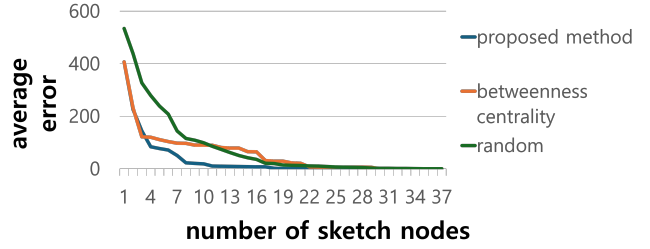


図 12: CAIS Internet の平均誤差

6. ま と め

本稿では、分散型ネットワーク環境における CMS によるフロー測定の効率と精度の向上を目的として、媒介中心性に基づくスケッチノード配置方式とフラグ設定による重複測定の回避の 2 種類を提案した。5 種類の実ネットワークトポロジに対する評価では、提案手法はトポロジに関係なく、少数のスケッチノードでも多くのフローをカバーでき、平均誤差が非常に小さく、高精度な推定が可能であることを確認した。特に、提案方式は少数のスケッチを用いた場合でも高いカバレッジを維持し、他の配置方式よりも少数のスケッチノードで効率的にネットワーク全体のフローを把握できることが確認された。

謝辞 本稿は JSPS 科研費 (25K03113, 23K28078) の助成を受けたものである。ここに記して謝意を表す。

文 献

- [1] G. Cormode and S. Muthukrishnan, "An Improved Data Stream Summary: The Count-Min Sketch and Its Applications," J. Algorithms, vol. 55, no. 1, pp. 58–75, Apr. 2005.
- [2] F. Li, K. Guo, J. Shen, and X. Wang, "Effective Network-Wide Traffic Measurement: A Lightweight Distributed Sketch Deployment," Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM), pp. 181–190, 2024.
- [3] X. Chen, Q. Xiao, H. Liu, Q. Huang, D. Zhang, X. Liu, L. Hu, H. Zhou, C. Wu, and K. Ren, "Eagle: Toward Scalable and Near-Optimal Network-Wide Sketch Deployment in Network Measurement," Proc. ACM SIGCOMM Conf. Data Commun., pp. 291–310, 2024.
- [4] R. Friedman, "An Evaluation of Software Frequency Sketches," Proc. ACM Int. Conf. Distributed and Event-Based Systems (DEBS), pp. 18–29, 2024.