【VLDB2013勉強会】

Session 14: Temporal, Stream and Event processing

担当：渡辺陽介（東京工業大学）

担当論文

1. Streaming Algorithms for k-core Decomposition
   - Erdem Sarıyüce (OSU), Buğra Gedik (Bilkent University), Gabriela Jacques-Silva (IBM T.J. Watson Research Center), Kun-Lung Wu (IBM T.J. Watson Research Center), Ümit Çatalyürek (OSU)

4. Sketch-based Geometric Monitoring of Distributed Stream Queries
   - Minos Garofalakis (Technical University of Crete (Greece), Daniel Keren (Haifa University), Vasilis Samoladas (Technical University of Crete)

5. Efficient Recovery of Missing Events
   - Jianmin Wang (Tsinghua University), Shaoxu Song (Tsinghua University), Xiaochen Zhu (Tsinghua University), Xuemin Lin (University of New South Wales)

Streaming Algorithms for k-core Decomposition

- グラフをk-coreと呼ばれる部分グラフに分解する問題
  - ある部分グラフHにおけるすべての頂点が，必ずk個以上の頂点とつながっている場合，Hを*seed k-core*と呼ぶ
  - Seed k-coreであるHのうち，他のseed k-coreの部分グラフになっていないものをk-coreと呼ぶ

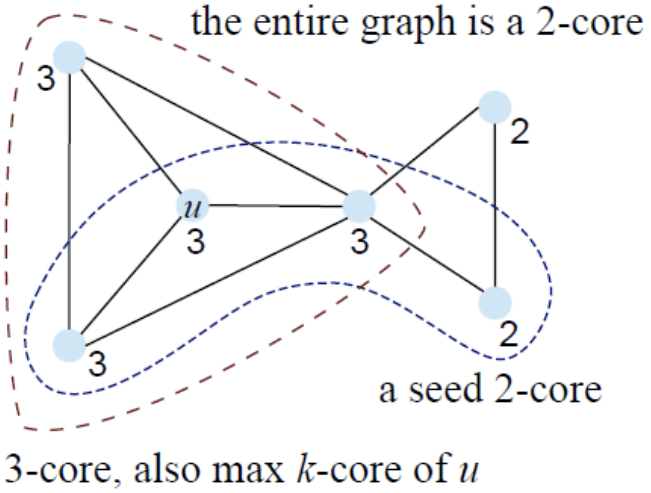


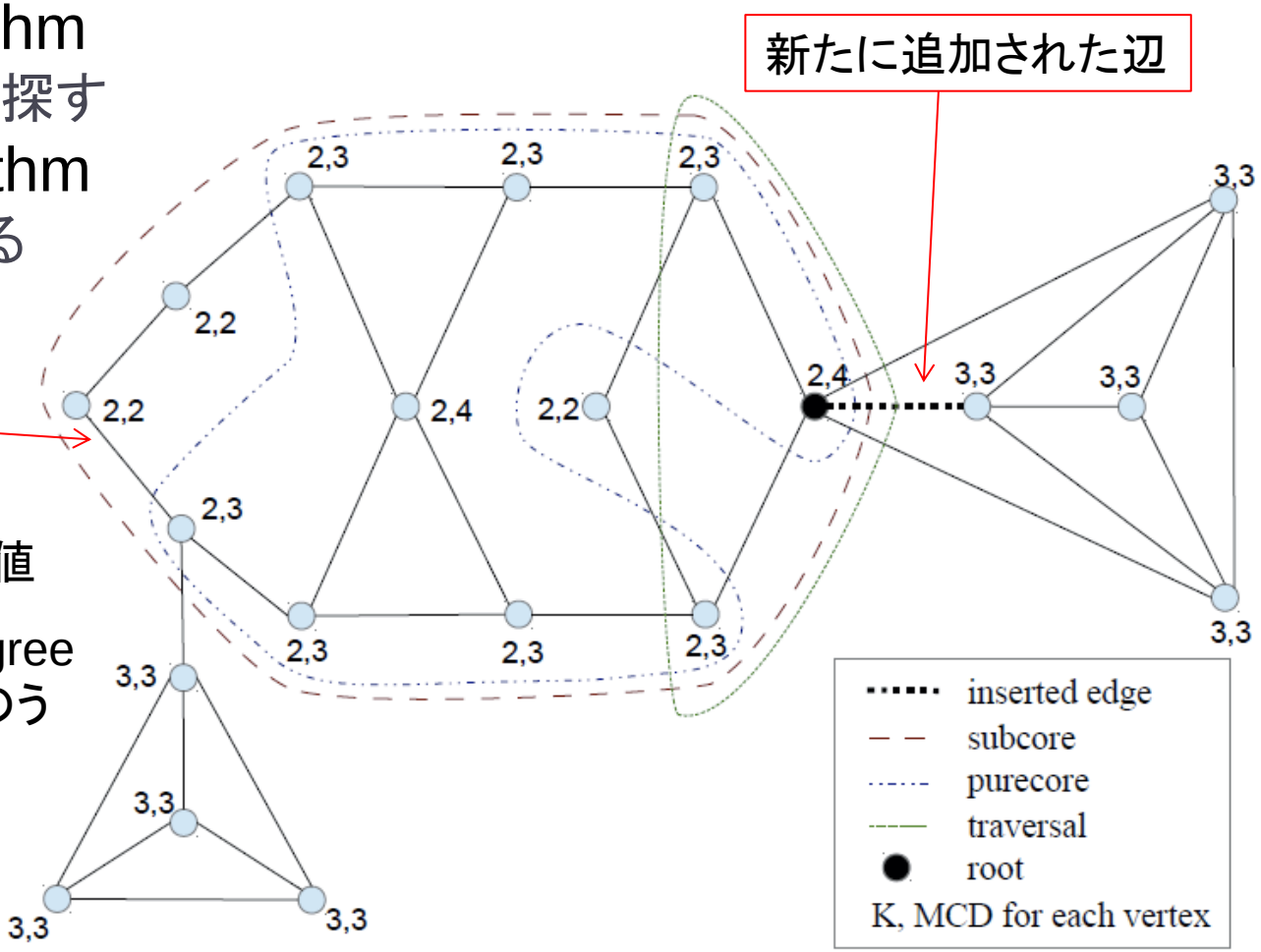
Figure 1より引用

- 辺の追加・削除に対してk-core分解を効率よく計算するためのインクリメンタルアルゴリズムの提案

Incremental Algorithms (3種類)

- Subcore algorithm
  - K(u)だけで探す
- Purecore algorithm
  - K(u)とMCD(u)で探す
- Traversal algorithm
  - 本命．色々頑張る

Figure 2より引用



K(u): 頂点uを含むk-coreのk値

MCD(u): maximum-core degree
ある頂点uに隣接する頂点wのうち，K(u)≤K(w)な頂点の数

評価実験

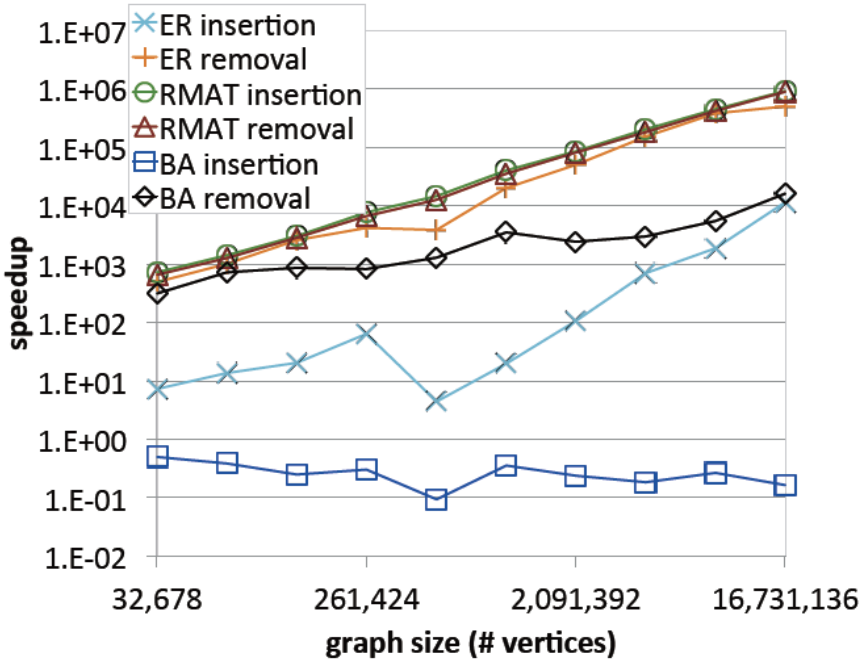
- 実験データ
  - 人工データ
    - SNAPライブラリ使用
  - 実データ（表1）

Table 1より引用

Graph file	Number of vertices	Number of edges	Maximum degree	Average degree	Max k
caidaRouterLevel	192,244	609,066	1,071	6.336	32
eu-2005	862,664	16,138,468	68,963	37.415	388
citationCiteseer	268,495	1,156,647	1,318	8.616	15
coAuthorsCiteseer	227,320	814,134	1,372	7.163	86
coAuthorsDBLP	299,067	977,676	336	6.538	114
coPapersCiteseer	434,102	16,036,720	1,188	73.885	844
cond-mat	16,726	47,594	107	5.691	17
power	4,941	6,594	19	2.669	5
protein-interaction-1	9,673	37,081	270	7.667	14

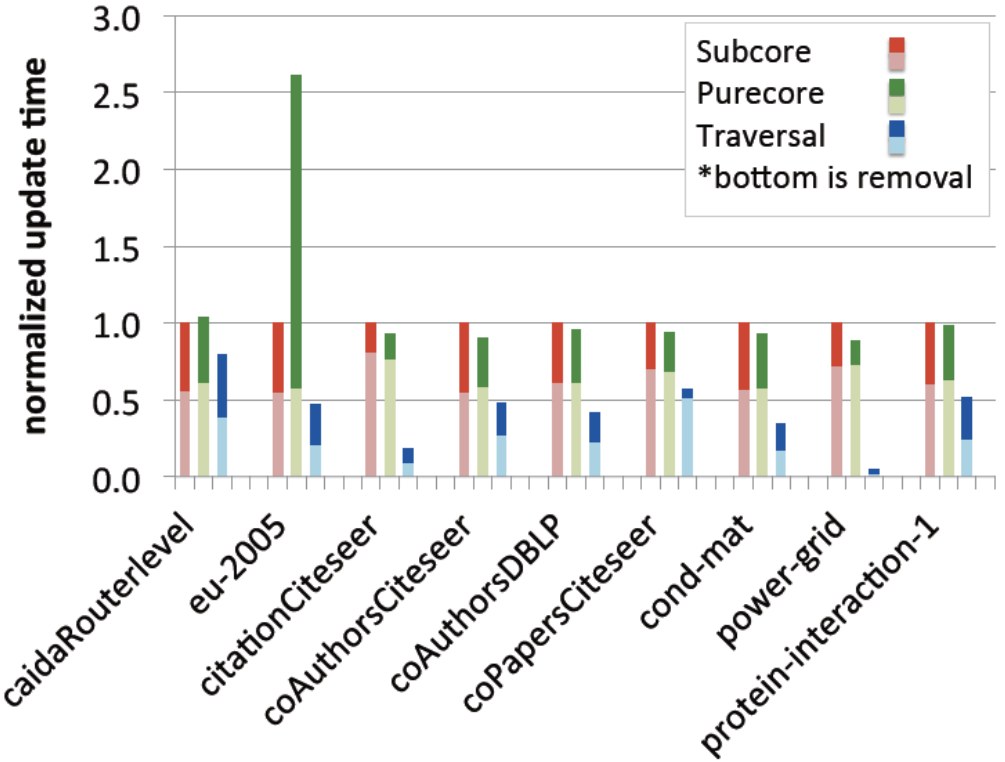
非インクリメンタル処理との比較

Figure 7より引用



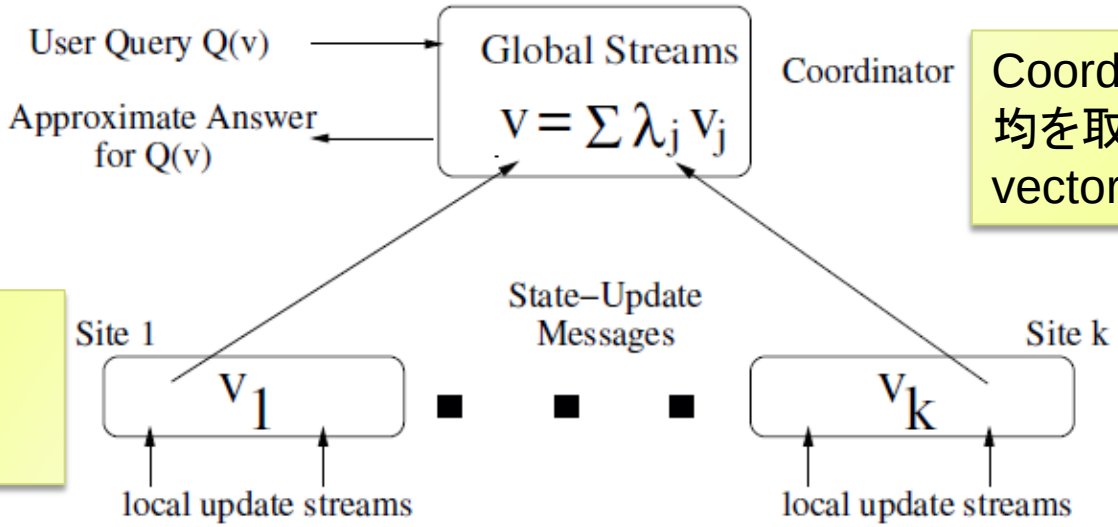
3種類の手法の比較

Figure 10より引用



Sketch-based Geometric Monitoring of Distributed Stream Queries

- 分散ストリーム環境においては，全データを中央に集めることは現実的ではない
  - データ量，ネットワーク帯域，電力消費（センサーネット）などの問題
- 各サイトで集約値であるsketchを計算し，中央ではsketchを集めてそこから近似的な解を得る



Coordinatorで重み付き平均を取ってglobal statistics vector vを計算

各サイトでlocal statistics vector v_kを計算

Figure 1より引用

論文の貢献

- AMS sketch [1, 2, 15] を用いて各サイトの情報を集め，Geometric method [31] で近似解を計算

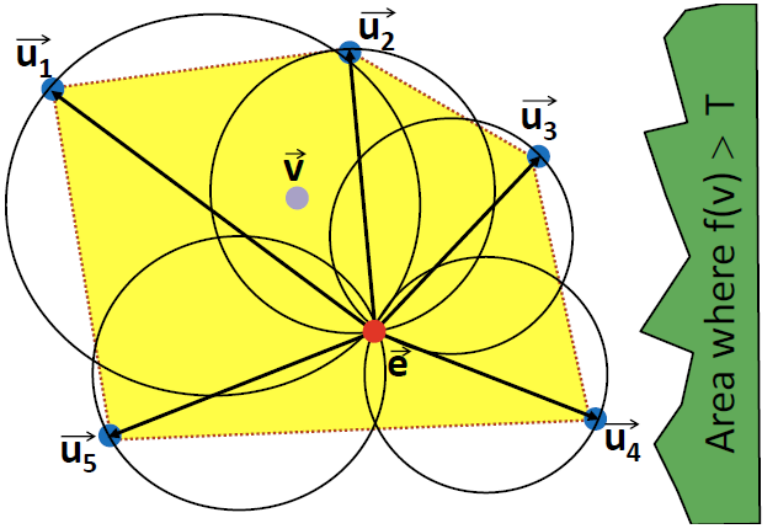
- 多様な近似計算を対象
  - Inner product
  - Range aggregate
    - e.g. quantiles, histograms, wavelets, heavy-hitters

Geometric method [31]

- Global vector vに対する関数f(v)が，f(v)>Tとならないかを分散環境で監視する手法

- 以前にGlobal vector v を推定した時の値 e と各サイトjのlocal vectorのずれΔv_jを求める
- 本当の v は e からΔv_jずれた範囲に入る(右図の黄色の多角形領域)
- 黄色の領域の代わりにbounding ballを使う
- 各サイトは推定値eと自分のlocal vectorから自分のbounding ballの範囲内のみをチェックする

Figure 2より引用



提案手法では，各u_jの値自体もストリームのSketchであり集約値

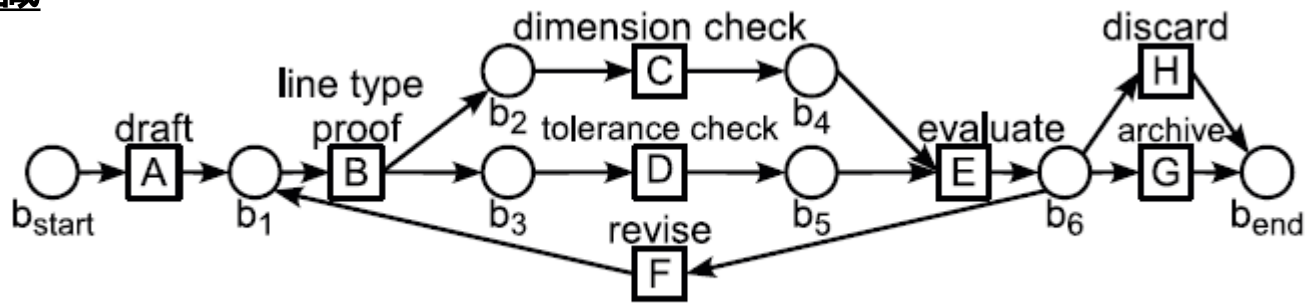
Efficient Recovery of Missing Events

- 実データのイベントログにはしばしば欠落が起こる
  - 手動による登録漏れ，システム障害，異なる環境から収集したログの混在など
  - 失われたイベントを無視したままでは，知識発見も成功しない
- 失われたイベントの回復問題
  - ただし，何の前提知識もないと不可能なので，ビジネス上のルールや制約を使う (Process specification)

Example

Figure1より引用

前提知識



(a) process specification

※ループを含む可能性がある

ログ

ID	Sequence of events
1	<A, B, C, D, E, G>
2	<A, B, C, E, G>
3	<A, B, C, D, G>

(b) event log of process execution

正常に記録されている

本当は ABCDEG かな？

ABCDEG ← Minimum recovery

ABCDEFBCDEG

ABCDEFBCDEFBCDEG

….

提案内容

- Linear time backtracking algorithm
- 欠落イベントに対するMinimum recoveryを探す問題が一般的にはNP-hardであることを示す
- ループをサポートした，枝刈り手法
- Top-kリカバリ
  - リカバリの候補をk個生成する