

# **Fast and Accurate Influence Maximization on Large Networks with Pruned Monte-Carlo Simulations**

(AAAI-14)

大坂 直人 (東京大学)

秋葉 拓哉 (東京大学)

吉田 悠一 (NII & PFI)

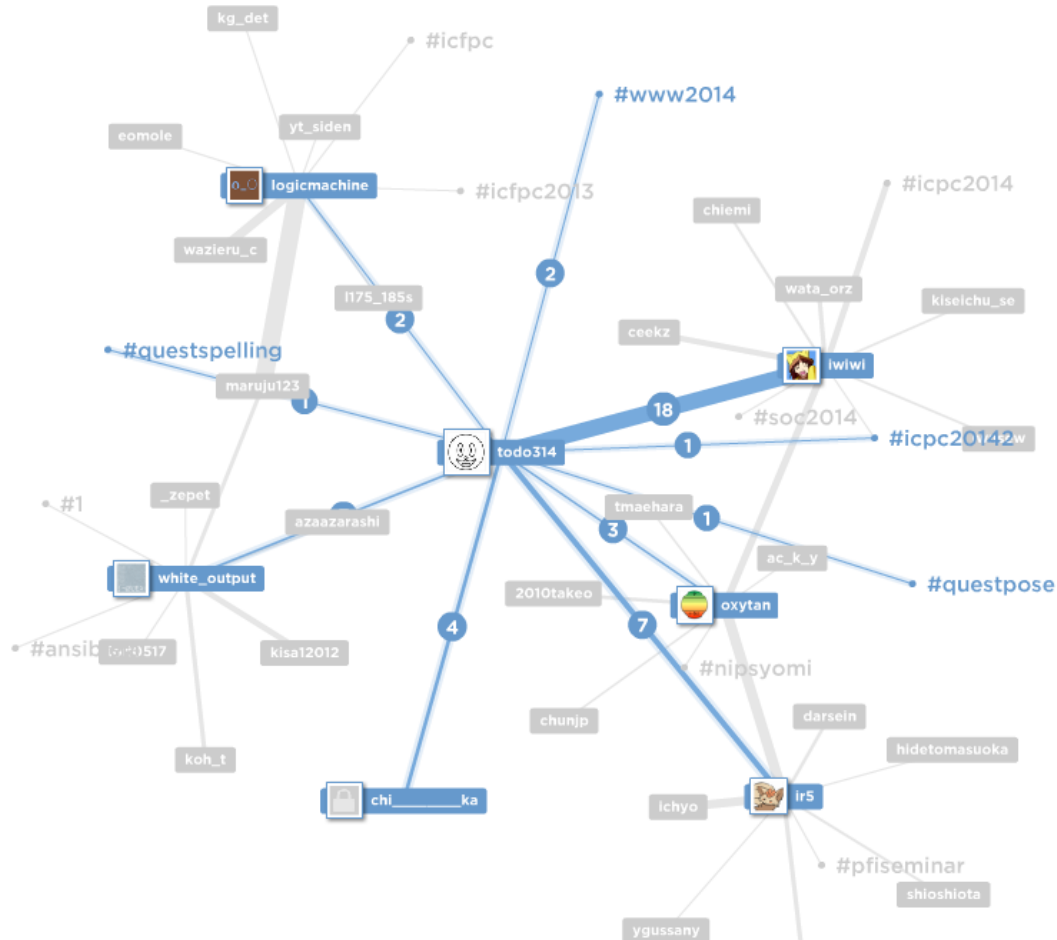
河原林 健一 (NII)

# 影響最大化

# Influence Maximization

# ネットワーク上の情報拡散

近傍の近傍の近傍の・・・

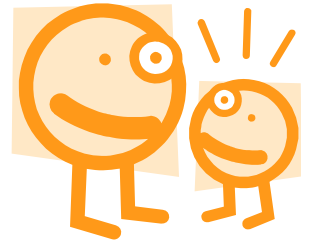


A mention network on Twitter generated by [mentionmapp.com](http://mentionmapp.com)

# バイラル (感染式) マーケティング

[Domingos, Richardson. KDD'01], [Richardson, Domingos. KDD'02]

- 「口コミ」を利用したマーケティング戦略
  - 少数に無料 or 割引商品を提供
  - 多数に宣伝効果
  - (提供コスト) < (宣伝による利益)だと嬉しい
- 例: Hotmail
  - 18ヶ月で12,000,000ユーザに増加
  - 「Hotmailで無料電子メールを入手しよう」



# 疑問

- 影響力の高い少数の集団を選択するには？
- 口コミ（情報拡散）のモデルは？

**影響最大化**

離散最適化問題

[Kempe, Kleinberg, Tardos. KDD'03]

ここから: アルゴリズムミックスな話題

# 影響最大化

(Influence Maximization)

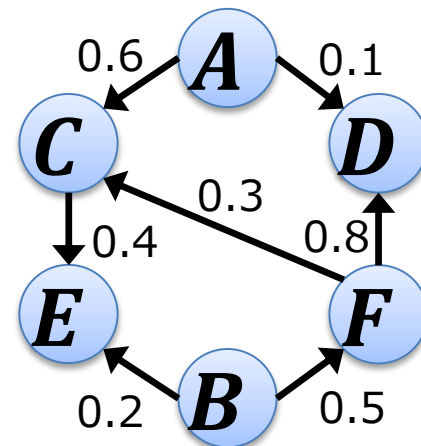
[Kempe, Kleinberg, Tardos. KDD'03]

## ■ 入力

- 有向グラフ  $G = (V, E)$
- 辺確率  $p_e$  ( $e \in E$ )
- シードサイズ  $k$

## ■ 問題

- maximize  $\sigma(S)$  ( $|S| \leq k$ )
  - $\sigma(\cdot)$ : 影響拡散 (情報拡散モデル依存)

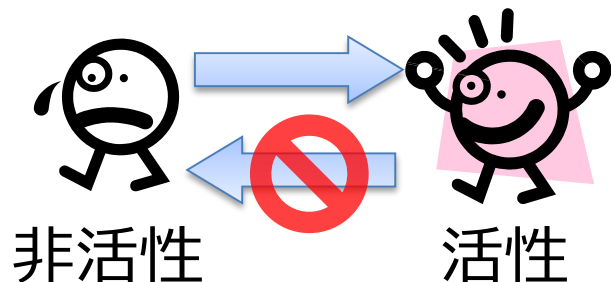


# 独立カスケードモデル

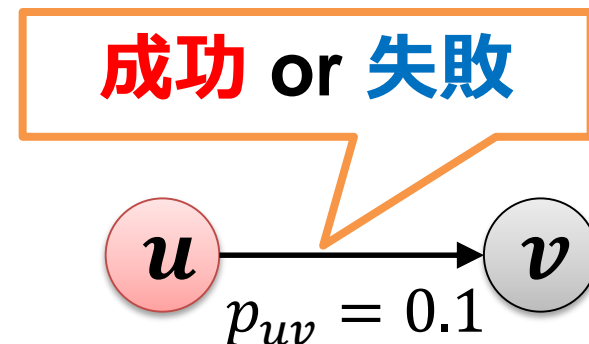
(Independent Cascade Model)

[Goldenberg, Libai, Muller. Marketing Letters'01]

- 頂点の状態: 非活性 or 活性

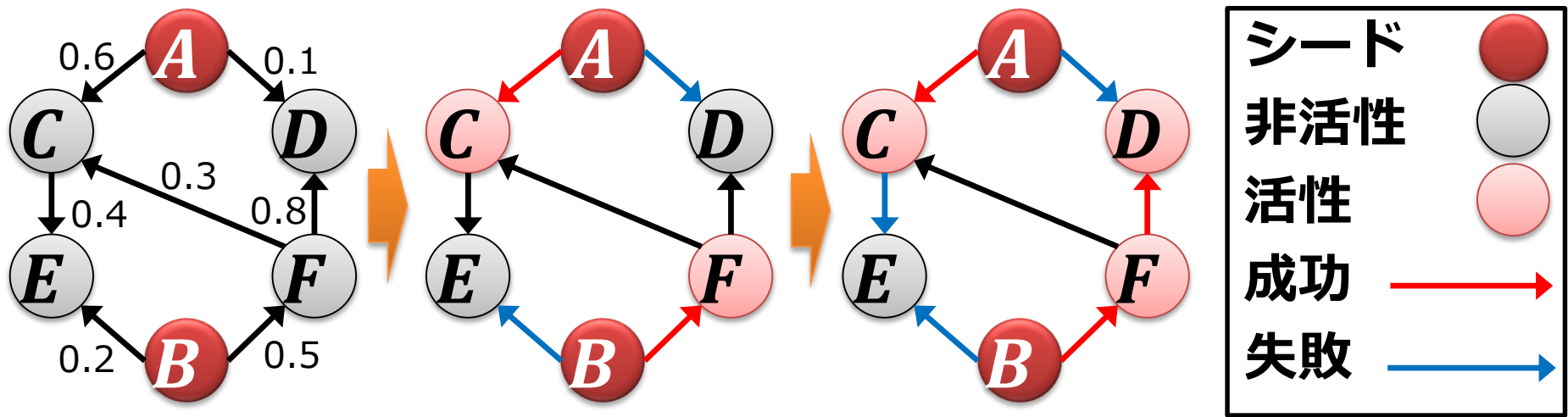


拡散過程



0.  $S \subseteq V$  (シード) 内の頂点を活性化
1. 活性頂点  $u$  は非活性頂点  $v$  を確率  $p_{uv}$  で活性化 (一回きり)
2. 新たな活性化がある限り  $1$  を反復

# 独立カスケードモデルの例



## ■ 影響拡散 $\sigma(S)$

- $S$ をシードとした時に  
活性化する頂点数の**期待値**



# 既存の結果

## 困難さ

影響最大化は  
**NP-hard**

[Kempe, Kleinberg, Tardos. KDD'03]

$\sigma(\cdot)$  の厳密計算は  
**#P-hard**

[Chen, Wang, Wang. KDD'10]

## アプローチ

貪欲アルゴリズム

[Kempe, Kleinberg, Tardos. KDD'03]

近似比  $\approx$  **63%**

Monte-Carlo  
シミュレーション  
 $\sigma(\cdot)$ を近似

# 貪欲+シミュレーション: 貪欲アルゴリズム

[Kempe, Kleinberg, Tardos. KDD'03]

$S \leftarrow \emptyset$

**while**  $|S| < k$  **do**

$t \leftarrow \arg \max_{v \in V} \sigma(S \cup \{v\}) - \sigma(S)$

$S \leftarrow S \cup \{t\}$

$\sigma(\cdot)$  の**劣モジュラ性**により

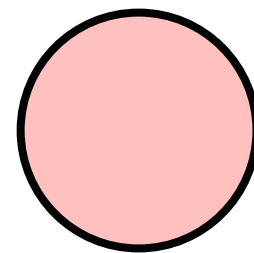
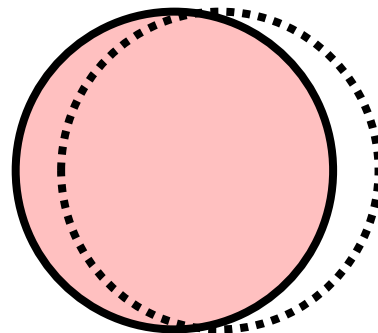
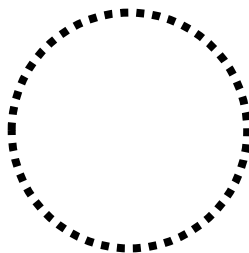
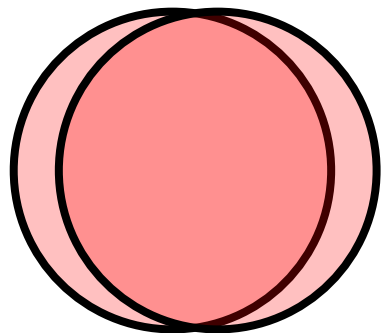
$$\sigma(S) \geq \left(1 - \frac{1}{e}\right) \text{OPT} \geq 0.63 \text{OPT}$$

[Nemhauser, Wolsey, Fisher.  
Mathematical Programming'78]

## 劣モジュラ

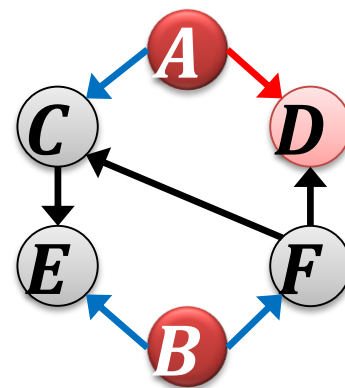
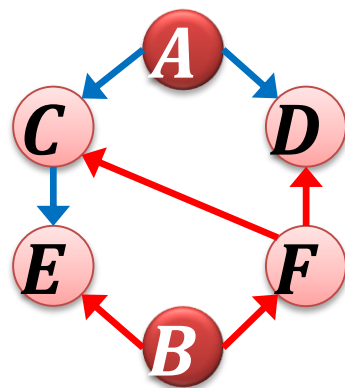
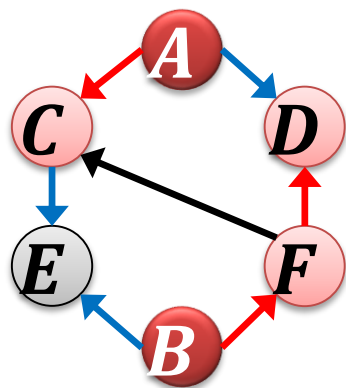
$$f(S + v) - f(S) \geq f(T + v) - f(T)$$

$$\forall S \subseteq T \subseteq V, v \in V$$



# 貪欲+シミュレーション: Monte-Carlo シミュレーション

- 拡散過程を繰り返しシミュレート
  - 10,000回くらい
- 活性頂点数の平均を出力



ほぼ最適な解を出力  $\left(1 - \frac{1}{e} - \varepsilon'\right)$   
何が問題か？

# 問題: 貪欲+シミュレーションは スケーラビリティに乏しい

貪欲アルゴリズム

$\sigma(\cdot)$ の計算回数:  $nk$

Monte-Carlo シミュレーション

$\sigma(\cdot)$ の計算時間:  $O(mR)$



総計算時間:  $O(knmR)$  ( $R \approx 10,000$ )

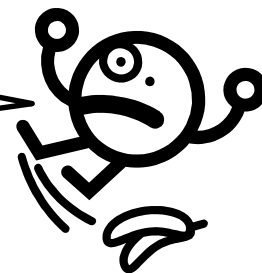
$n = |V| > 10^6$

$m = |E| > 10^7$



$k$ : シードの数

$R = \text{poly}(\varepsilon^{-1})$ : シミュレーション数

遅すぎる



# 既存手法

|    | 低品質   | 高品質   |
|----|---|---|
| 低速 |  <p>シミュレーション</p>  | <p>Greedy Approach<br/>[Kempe, Kleinberg, Tardos. KDD'03]</p> <p>CELF<br/>[Leskovec, Krause, Guestrin, Faloutsos, VanBriesen, Glance. KDD'07]</p> <p>StaticGreedyDU<br/>[Cheng, Shen, Huang, Zhang, Cheng. CIKM'13]</p> |
| 高速 | <p>DegreeDiscount<br/>[Chen, Wang, Yang. KDD'09]</p> <p>PMIA<br/>[Chen, Wang, Wang. KDD'10]</p> <p>SAEDV<br/>[Jiang, Song, Cong, Wang, Si, Xie. AAAI'11]</p> <p>IRIE<br/>[Jung, Heo, Chen. ICDM'12]</p> | <p>挑戦</p> <p><br/>ヒューリスティクス</p>   |

# 我々の貢献

- 枝刈りシミュレーションに基づく手法を提案
  - 高速
    - ヒューリスティクスと同等
    - **60,000,000辺** のグラフを **20分** で処理
  - 高精度
    - 理論的保証を有する

シミュレーションベースの手法の  
スケーラビリティが飛躍的に向上

# 提案手法

# 提案手法の概要

- 前処理: ランダムグラフの生成

↑ コインフリップテクニック

- 貪欲法

$S \leftarrow \emptyset$

**while**  $|S| < k$  **do**

$t \leftarrow \arg \max_{v \in V} \underline{\underline{\sigma(S \cup \{v\}) - \sigma(S)}}$

$S \leftarrow S \cup \{t\}$       ↑ 提案高速化手法



# 前処理: ランダムグラフの生成

## コインフリップテクニック

[Kempe, Kleinberg, Tardos. KDD'03]

影響拡散  $\sigma(S)$  の計算

||

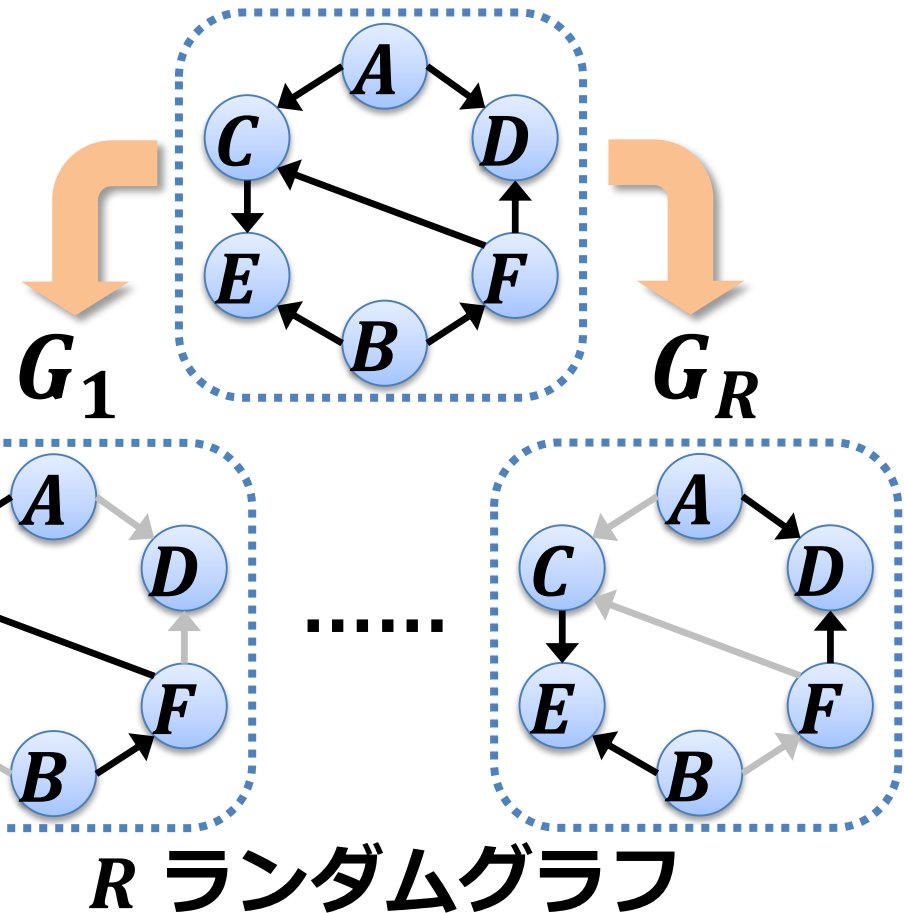
ランダムグラフ上で  $S$  から  
**到達可能**な頂点数の計算



各辺  $e$  を確率  $p_e$  で残す

残った辺: 試行**成功**  
消えた辺: 試行**失敗**

入力グラフ  $G$



# $\sigma(S)$ の近似

$$\sigma(S) \approx \frac{1}{R} \sum_{1 \leq i \leq R} \sigma_{G_i}(S)$$

$\sigma_{G_i}(S) = G_i$ 上で $S$ から  
**到達可能**な頂点数

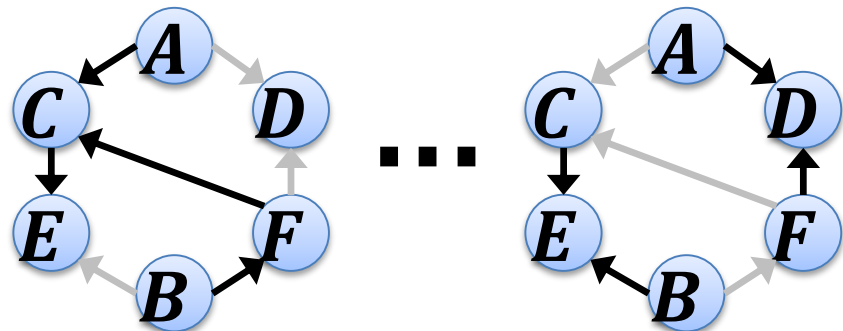
**挑戦**

右の表を可能な限り  
**高速**に求めたい！

**$R = 200$**

| $v$ | $\sigma_{G_1}(\{v\})$ | ... | $\sigma_{G_R}(\{v\})$ | $\sigma(\{v\})$ |
|-----|-----------------------|-----|-----------------------|-----------------|
| $A$ | 3                     | ... | 2                     | 2.4             |
| $B$ | 4                     | ... | 2                     | <b>2.8</b>      |
| $C$ | 2                     | ... | 2                     | 1.6             |
| $D$ | 1                     | ... | 1                     | 1               |
| $E$ | 1                     | ... | 1                     | 1               |
| $F$ | 3                     | ... | 2                     | 2.2             |

**$10^6$**



# 提案高速化手法

(各ランダムグラフに独立に適用)

## 1. Pruned BFS (枝刈り幅優先探索)

ソーシャルネットワークの構造的性質を利用

[Akiba, Iwata, Yoshida. SIGMOD'13]

[Yano, Akiba, Iwata, Yoshida. CIKM'13]

[Akiba, Iwata, Kawarabayashi, Kawata. ALENEX'14]

} 核

## 2. 不要な $\sigma_{G_i}(\cdot)$ 再計算の検知・回避

## 3. 標本平均近似法によるランダムグラフ数 $R$ の抑制

[Kimura, Saito, Nakano. AAAI'07], [Cheng, Shen, Huang, Zhang, Cheng. CIKM'13]

[Sheldon et al., UAI'10]

- 理論的根拠を与えた

$\sigma(\cdot)$  の見積もり精度に影響を与えない

# Pruned BFS

- アイデア: ほとんどのBFSは**冗長**
- **前処理**:  
次数最大の頂点  $H$  の**先祖**と**子孫**を計算
- **枝刈り** ( $v$  から BFS) :  
もし  $v$  が  $H$  の**先祖**なら,  $H$  の**子孫**を無視

(BFS中に訪れた頂点数)

# 2

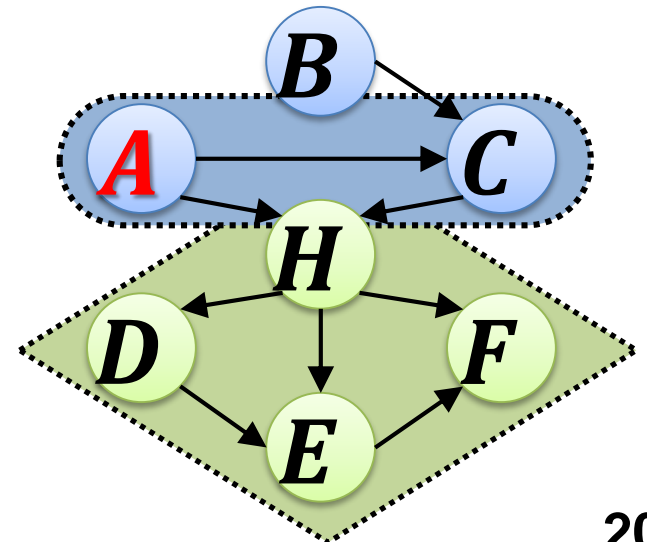
+

+

( $H$ の子孫の数)

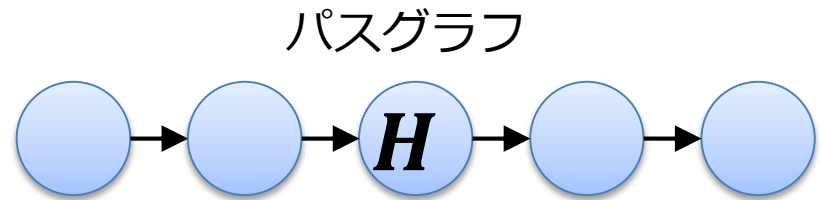
4

↑ 前計算



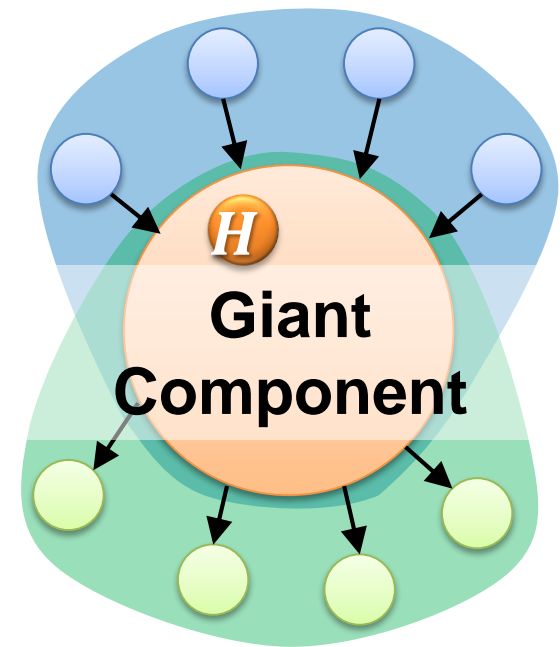
# Pruned BFS は効果的？

- パスグラフでは...
  - 非効率的  $\Theta(|V|^2)$



- ソーシャルネットワークなら...
  - 効率的
  - **ハブ** / **巨大連結成分**の存在

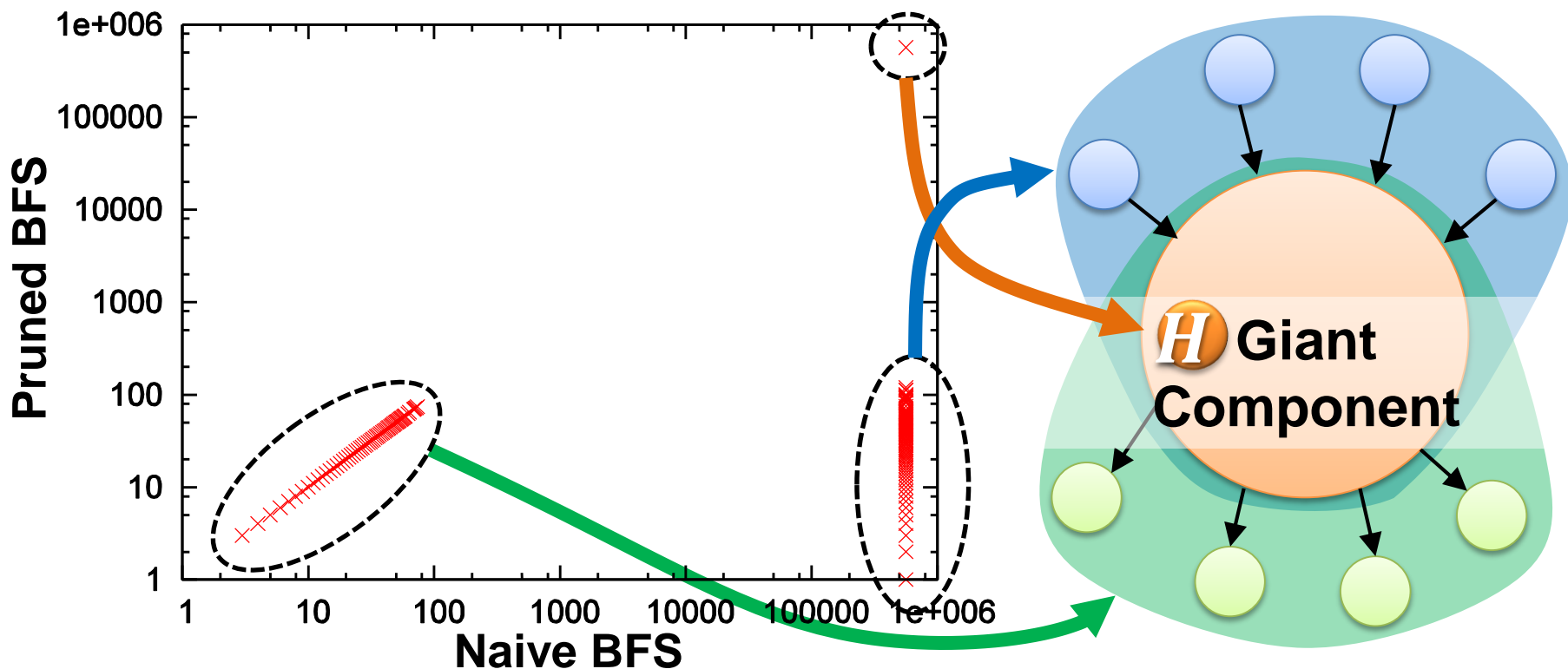
ソーシャルネットワーク



# ソーシャルネットワーク上の Pruned BFS の効果

(LiveJournal dataset,  $|V| = 4.8\text{M}$ ,  $|E| = 69\text{M}$ ,  $p_e = 0.1 \forall e$ )

## ■ Naive BFSとPruned BFSで訪れた頂点数の分布



## ■ 各頂点から訪れた平均頂点数

■ **400,000** (Naive BFS)  $\Rightarrow$  **6** (Pruned BFS)

# 高速化手法の直感的意味

- 前処理: ランダムグラフ  $G_1, \dots, G_R$

- 貪欲法

$S \leftarrow \emptyset$

**while**  $|S| < k$  **do**

$t \leftarrow \arg \max_{v \in V} \frac{1}{R} \sum_{1 \leq i \leq R} \sigma_{G_i}(S \cup \{v\}) - \sigma_{G_i}(S)$

$S \leftarrow S \cup \{t\}$

手法3  
標本平均近似法

手法2  
検知・回避

手法1  
Pruned BFS

**$k$** : **while**  $|S| < k$  **do**

**$n$** :  $\arg \max_{v \in V}$

**$R$** :  $\sum_{1 \leq i \leq R}$

**$m$** :  $\sigma_{G_i}(S \cup \{v\}) - \sigma_{G_i}(S)$

$O(knmR)$

# 実験結果



# 実行時間 [秒] の比較 (手法3有)

各辺について  $p_e = P$ . シードサイズ  $k = 50$

| データセット                        | 手法1 有<br>手法2 有 | 手法1 無<br>手法2 有 | 手法1 有<br>手法2 無 | 手法1 無<br>手法2 無 |
|-------------------------------|----------------|----------------|----------------|----------------|
| DBLP<br>( $P = 0.01$ )        | 27             | 26             | 149            | 158            |
| DBLP<br>( $P = 0.1$ )         | 54             | 3,036          | 306            | 3,275          |
| LiveJournal<br>( $P = 0.01$ ) | 327            | 1,934          | 2,176          | 3,820          |
| LiveJournal<br>( $P = 0.1$ )  | 634            | 272,518        | 2,426          | 272,973        |

400倍

10分 ⇐ 3日

| データセット       | $ V $ | $ E $ |
|--------------|-------|-------|
| DBLP         | 655K  | 2.0M  |
| Live Journal | 4.8M  | 69M   |

# 影響拡散

各辺について  $p_e = P$ . シードサイズ  $k = 50$

| データセット                        | 提案手法           | StaticGreedy<br>DU<br>[Cheng+'13] | IRIE<br>[Jung+'12] | PMIA<br>[Chen+'10] | SAEDV<br>[Jiang+'11] |
|-------------------------------|----------------|-----------------------------------|--------------------|--------------------|----------------------|
| DBLP<br>( $P = 0.01$ )        | <b>332</b>     | 330                               | 323                | 317                | 76                   |
| DBLP<br>( $P = 0.1$ )         | <b>100076</b>  | --                                | 99533              | 99505              | 99579                |
| LiveJournal<br>( $P = 0.01$ ) | <b>47527</b>   | --                                | 41906              | 40544              | 26066                |
| LiveJournal<br>( $P = 0.1$ )  | <b>1686629</b> | --                                | 1682436            | --                 | 1682242              |

| データセット       | $ V $ | $ E $ |
|--------------|-------|-------|
| DBLP         | 655K  | 2.0M  |
| Live Journal | 4.8M  | 69M   |

- 提案手法が最良

# 実行時間 [秒]

各辺について  $p_e = P$ . シードサイズ  $k = 50$

| データセット                        | 提案手法 | StaticGreedy<br>DU<br>[Cheng+'13] | IRIE<br>[Jung+'12] | PMIA<br>[Chen+'10] | SAEDV<br>[Jiang+'11] |
|-------------------------------|------|-----------------------------------|--------------------|--------------------|----------------------|
| DBLP<br>( $P = 0.01$ )        | 27   | 117                               | 77                 | 4                  | 388                  |
| DBLP<br>( $P = 0.1$ )         | 52   | OOM                               | 77                 | 289                | 388                  |
| LiveJournal<br>( $P = 0.01$ ) | 327  | OOM                               | 1,622              | 500                | 1,275                |
| LiveJournal<br>( $P = 0.1$ )  | 663  | OOM                               | 1,635              | OOM                | 1,294                |

- ヒューリスティクスと同等
- $P$  に対して頑健

| データセット       | $ V $ | $ E $ |
|--------------|-------|-------|
| DBLP         | 655K  | 2.0M  |
| Live Journal | 4.8M  | 69M   |

# 今後の展望

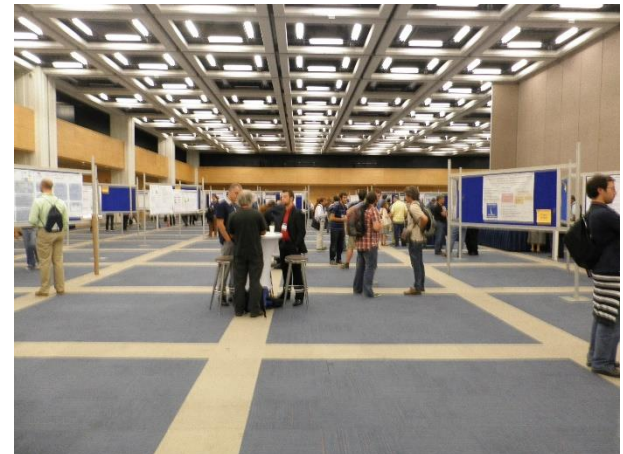
- 他モデルへの適用
- 並列化
- Pruned BFS の効率の解析

# 28th AAAI Conference on Artificial Intelligence (AAAI-14)

- 人工知能のトップ会議
- 7/27～31, Québec, Canada
- 採択論文: 約400本
  - 採択率28%
  - 日本からは9本
- 参加者: 約1000人
- 機械学習, ゲーム理論, 探索, 自然言語処理, プランニング, ビジョン, WEB, ...



口頭発表



ポスター発表

