2022/1/20

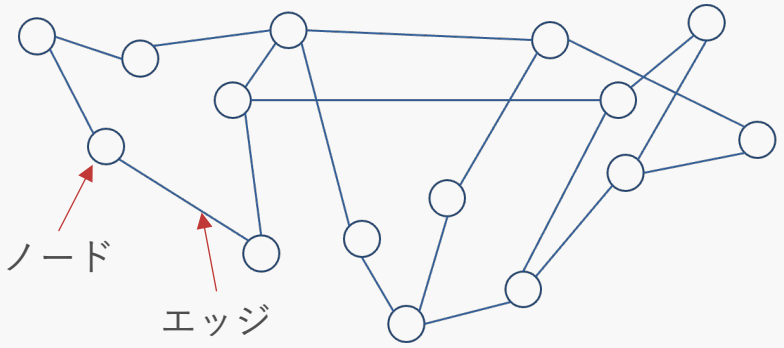
地理的に分散したグラフ上における非同期 Random Walk 処理システム

滝沢 駿

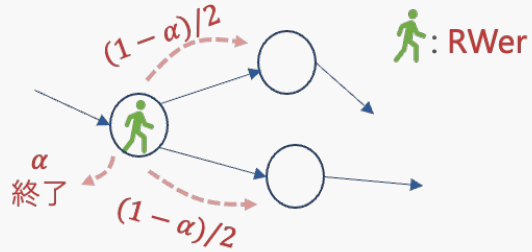
慶應義塾大学大学院開放環境科学専攻 修士 1 年

研究概要 (地理的に分散したグラフ上における非同期 RandomWalk 処理システム) 2

- **グラフ解析**
  - SNS : ノード→ユーザ, エッジ→フォロー
  - Web : ノード→サイト, エッジ→リンク
  - ✓ 影響力の高いノードの抽出
  - ✓ コミュニティの抽出
  - ✓ あるノードから見た他のノードの関連度

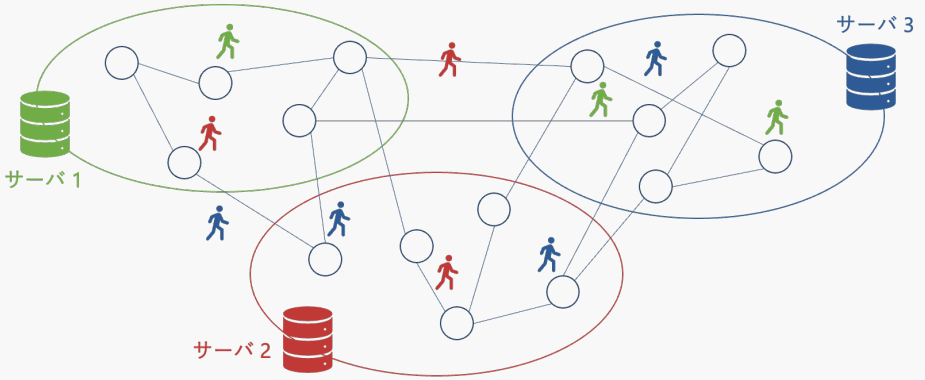


- **Random Walk (RW)**
  - 様々なグラフ解析の基礎となる演算
  - RW の経路情報はグラフ解析において重要

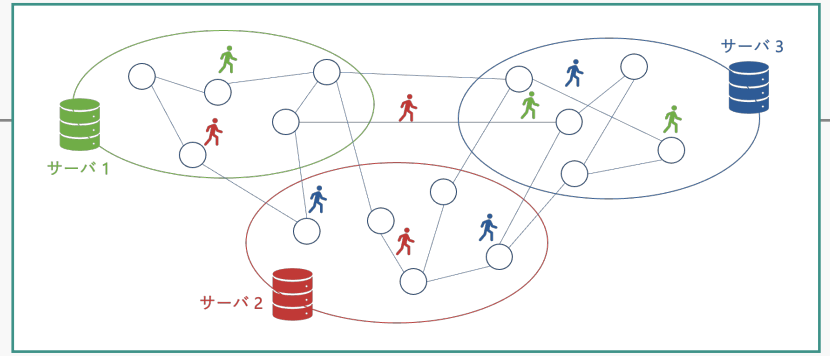


Random Walk
確率 α で終了し, 確率 $1-\alpha$ で
ランダムな隣接ノードへ遷移

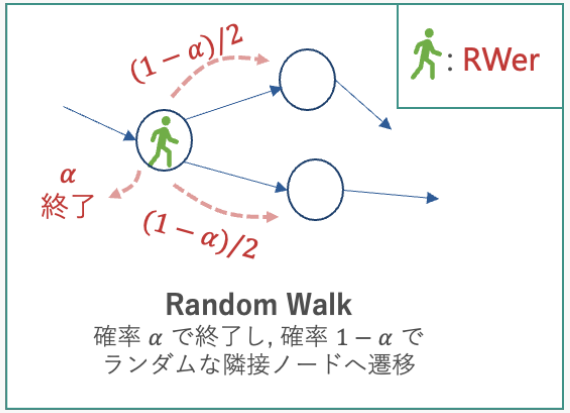
- **地理的分散環境でのグラフ解析 (RW)**
  - シングルマシンには限界がある
    - ➢ メモリ, CPU, 世界中のデータの保存 (地理的分散)
  - 巨大なグラフを複数のマシンで分割して保存
  - 複数のマシンが協力して RW を実行



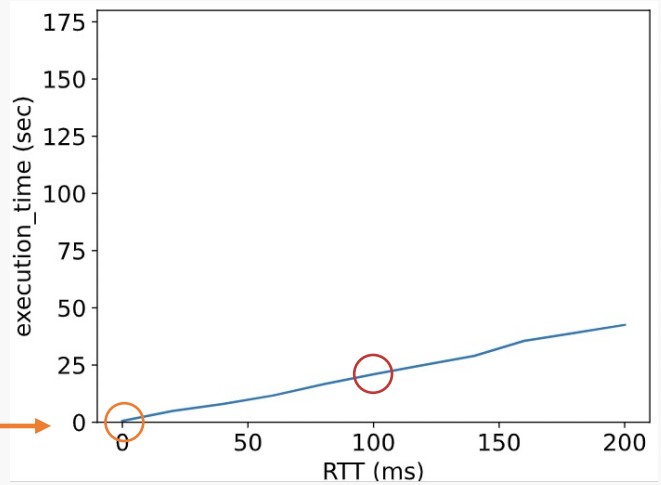
分散環境におけるグラフ上での Random Walk



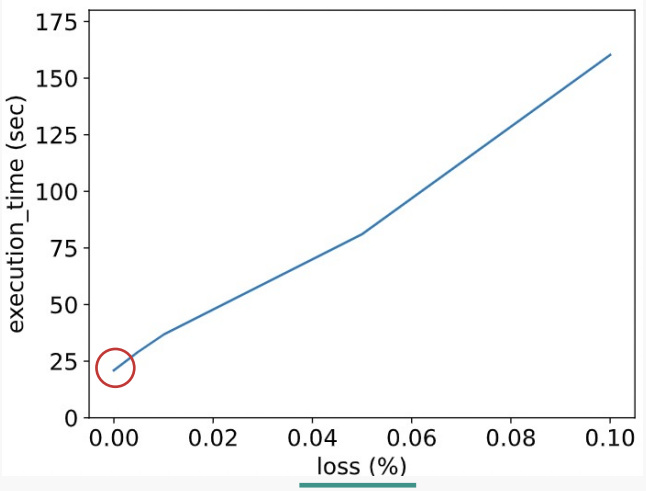
- **グラフ上での Random Walk (RW) の利便性**
  - e.g., 推薦システム, コミュニティ検出, 類似性推定
- **グラフデータは地理的に分散した環境下で保持**
  - 高 RTT, 高パケットロス率の通信環境
- **既存手法：単一データセンター内での処理に特化**
  - Bulk Synchronous Parallel (BSP, バルク同期並列) モデル
  - 地理的分散環境下では通信のスループットが大幅に低下



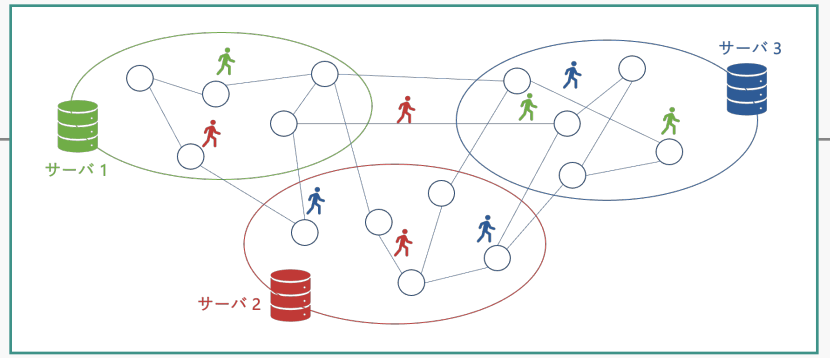
5 台のサーバでの
RW 5,000,000 回の実行時間
左 : パケロス率 = 0% 固定
右 : RTT = 100ms 固定



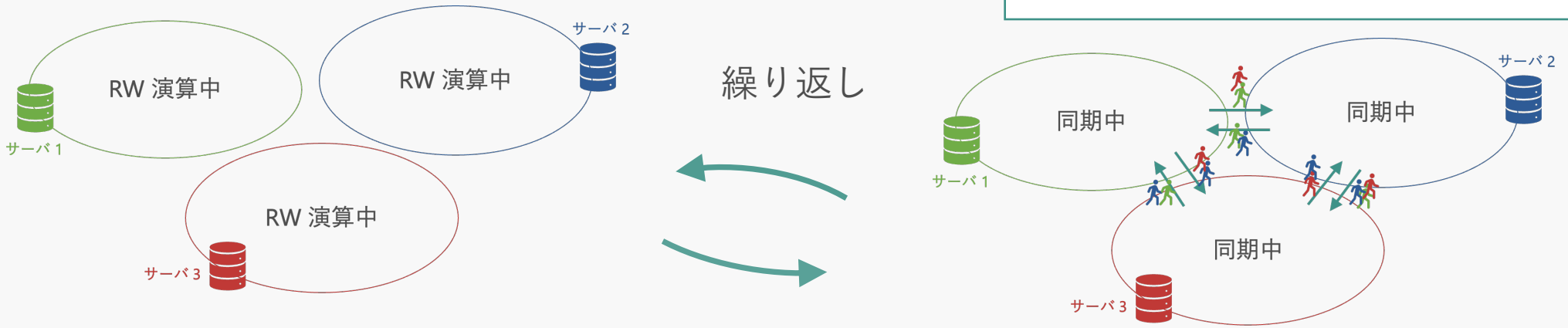
理想的な環境の場合は約 0.5 sec



関連研究：BSP モデルの RW 処理システム[1]



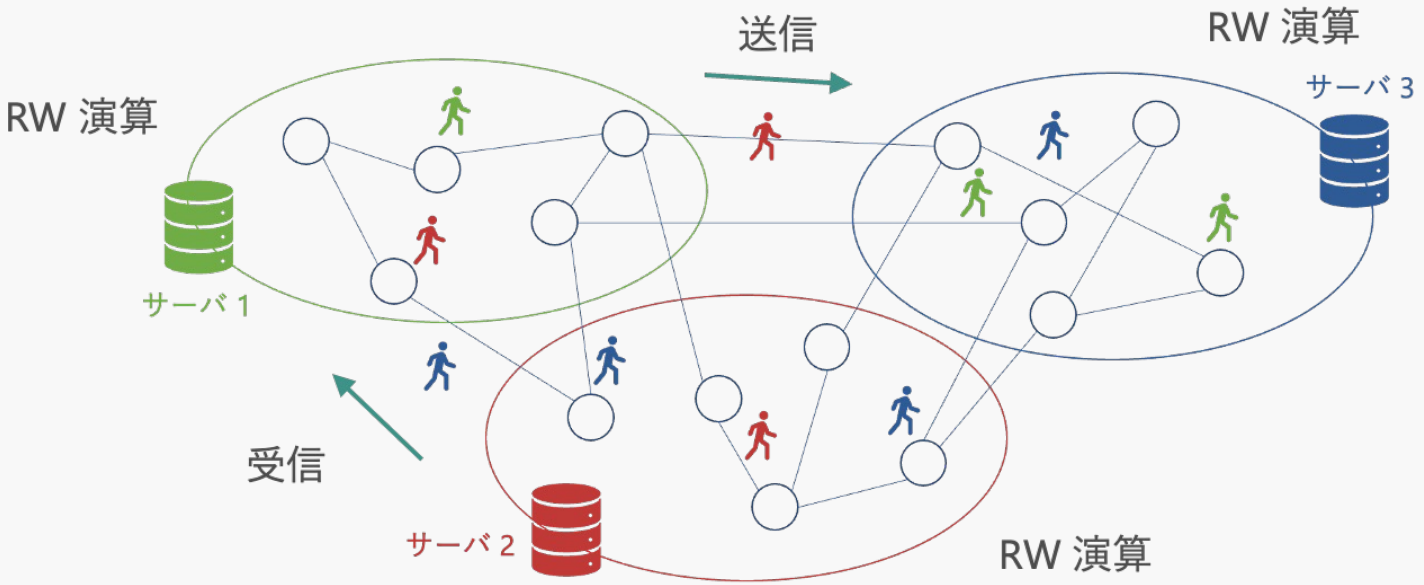
- **計算フェーズ ⇄ 通信フェーズ**



- **サーバ内での RW 演算に OpenMP, サーバ間同期に MPI（TCP 通信）を使用**
- **単一データセンター内での処理に特化**
  - RTT, パケロスを無視できる環境
- **計算負荷が均等になるようなグラフ分割**
  - 不均等なグラフ分割のとき, 早く演算が終わったサーバが同期のタイミングまで待つ必要があり, 偏りが発生する

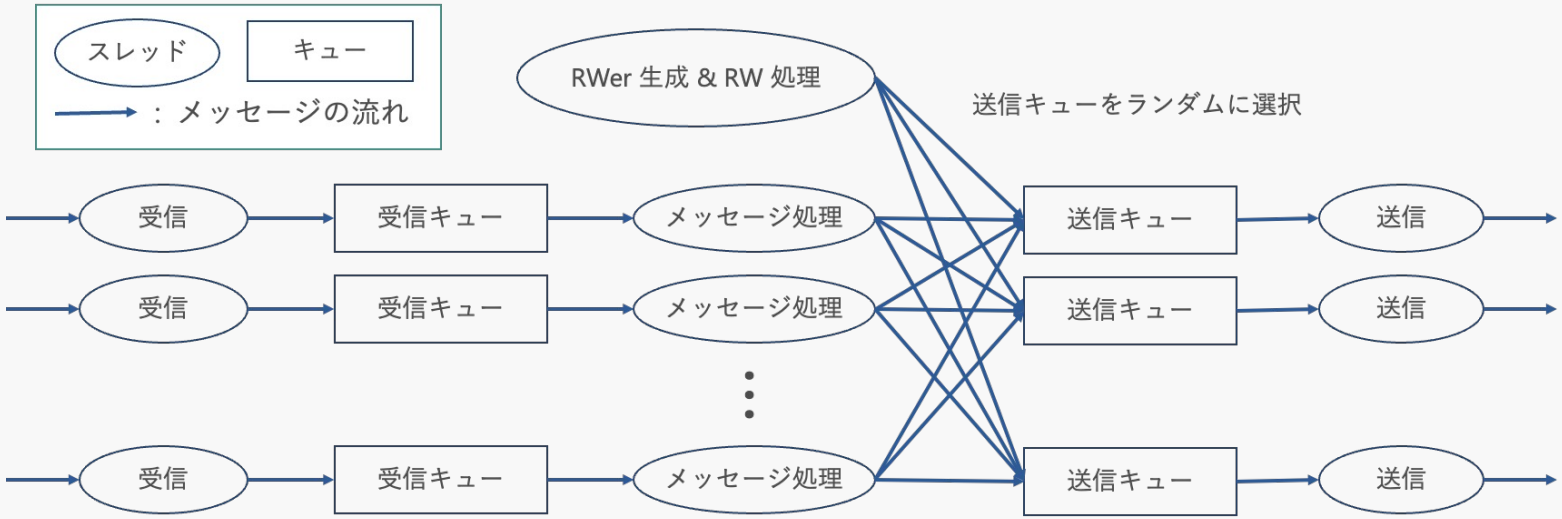
[2]Ke Yang, MingXing Zhang, Kang Chen, Xiaosong Ma, Yang Bai, and Yong Jiang. Knightking: A fast distributed graph random walk engine. In *Proceedings of the 27th ACM Symposium on Operating Systems Principles*, SOSP '19, page 524–537, New York, NY, USA, 2019. Association for Computing Machinery.

- **計算フェーズと通信フェーズを同時並行（非同期）に**
  - 同期待ちが発生しないので, グラフの分割が不均一であったとしても影響が少ない
- **1 パケット = 1 RWer の UDP 通信**
  - RW の計算単位は RWer
  - 高 RTT, 高パケットロス率であったとしても影響が少ない



- **送受信とメッセージ処理に CPU リソースを最大限割り当てる**
  - ポート番号毎に送受信キューとメッセージ処理, 送受信用スレッドを生成

※ 送信キューを送信先毎にすると, 送信先が偏ったときに並列に送信できなくなる



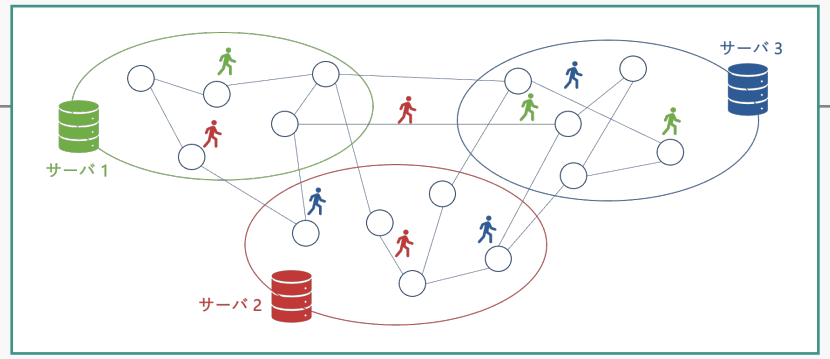
C++ で実装

- **メッセージ処理：受信キューから取り出す + RWer 復元 + RW 処理 + メッセージ生成 + 送信キューに入れる**

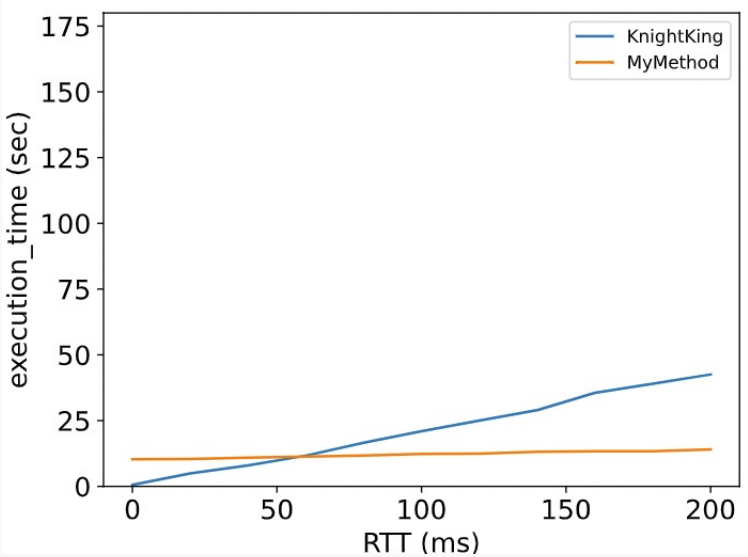
メッセージ内容

メッセージ ID, 送信先 IP アドレス,
RWer 情報 (ID, 起点ノード, 起点サーバの IP アドレス, 現在のノード, 経路長, 経路情報)

まとめ



- **背景**
  - 地理的に分散したグラフ上で RW を実行
- **既存手法の問題点**
  - 高 RTT, 高パケロス率で通信のスループットが大幅に低下
  - 同期のオーバーヘッド
- **提案手法のアプローチ**
  - RTT に依存しない UDP 通信を使用
  - RWer の独立性を活かした非同期処理
- **基本評価 (既存手法との比較)**
  - 提案手法は RTT の影響が小さい
  - 現時点では約 RTT 60ms で同程度
  - ✓ 日中間で RTT が大体 50 ~ 100 ms



パケロス率 = 0% 固定, RTT 変動