

2019年度 情報処理学会関西支部 支部大会  
セッションC: ネットワークサービス1  
C-06

# 複数の異なる時間間隔に基づく 構造化オーバーレイネットワーク構築手法の評価

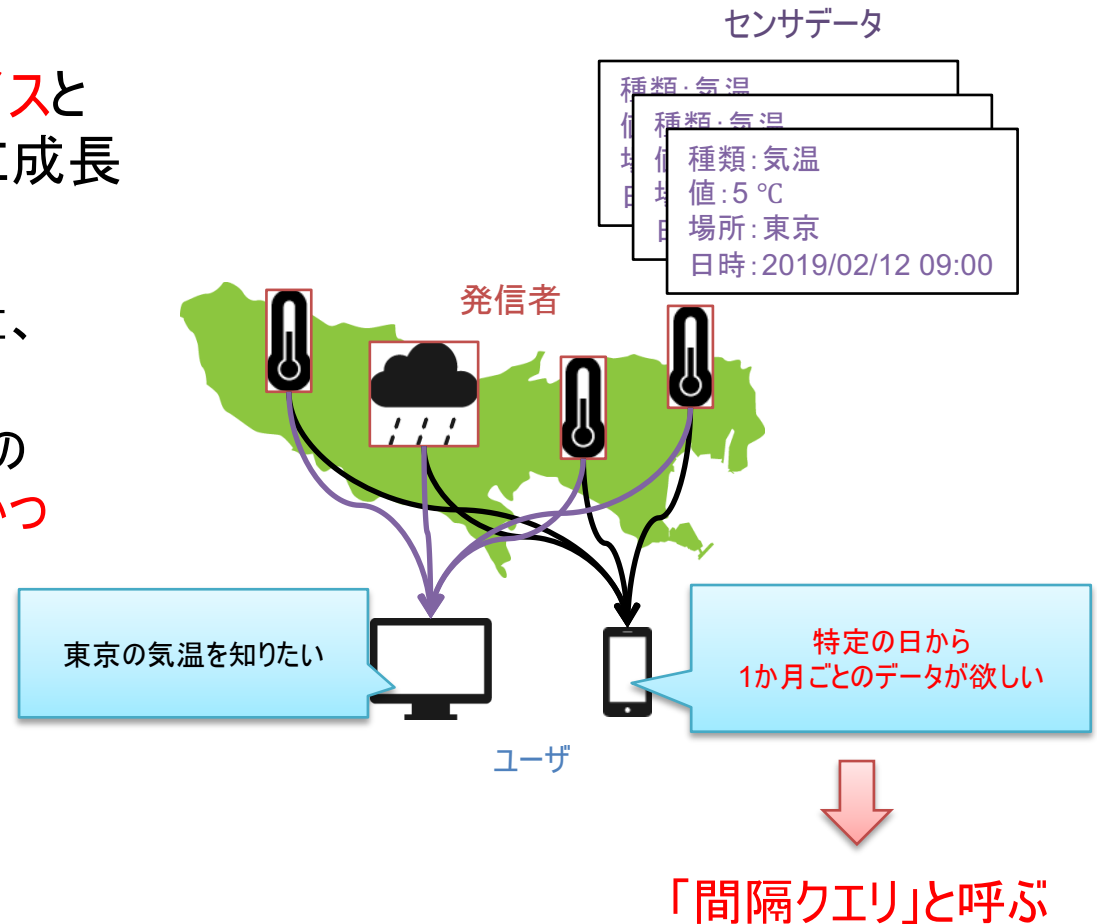
川上 朋也

奈良先端科学技術大学院大学

2019/09/23

# 背景

- IoT (モノのインターネット)デバイスとそれらの発信するデータが急速に成長
- センサデータは代表的な例
  - 気温、位置情報、消費電力量、音声、画像、映像など
  - 測定された地点や範囲、日時の情報に深く結びついた空間的かつ時間的なデータ
    - 関係する情報に基づいて、検索や利活用が行われる
    - 実際のユーザやシステムは特定の時間間隔を想定すると考えられる



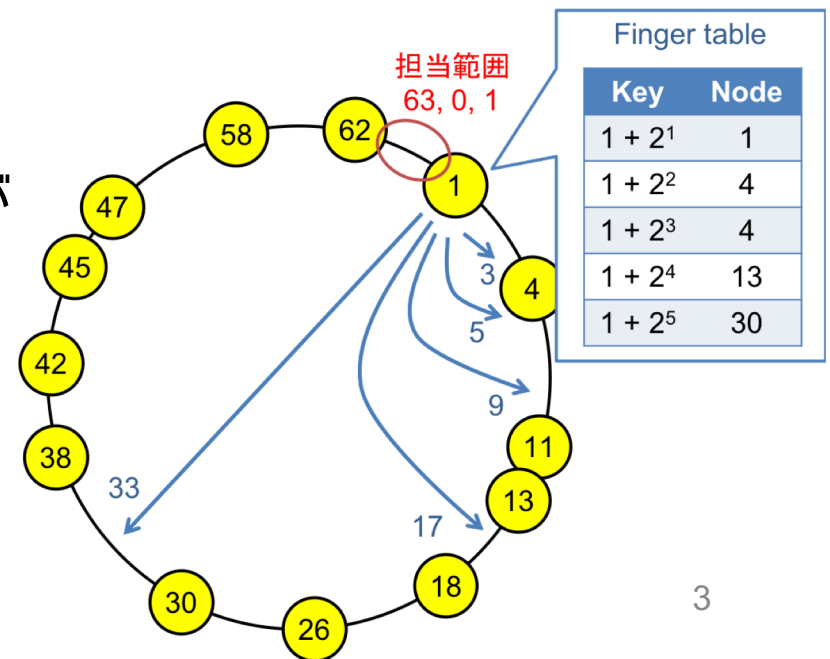
膨大な量のデバイス(ノード)やデータ、利用者を扱うために  
高いスケーラビリティを維持しつつ、データを効率的に検索可能な仕組みが必要

# 構造化オーバーレイネットワーク

- 複数のコンピュータ(ノード)で高いスケーラビリティを実現する技術
  - 分散ハッシュテーブル(DHT)、スキップグラフ、地理的オーバーレイネットワーク
- 特定の情報(キー)やルールに基づいて、ノード間に論理ネットワーク(仮想リンク)を構築
  - ID, IPアドレス、二次元座標など
- キーの値や範囲をクエリ(問合せ)で指定し、特定のノードを探索
- 各ノードは自身のルーティングテーブルに基づいて、対象ノードへ届くようにクエリを転送
- 各ノードは分担するキーの範囲(担当領域)があり、コンテンツの分散管理などに利用

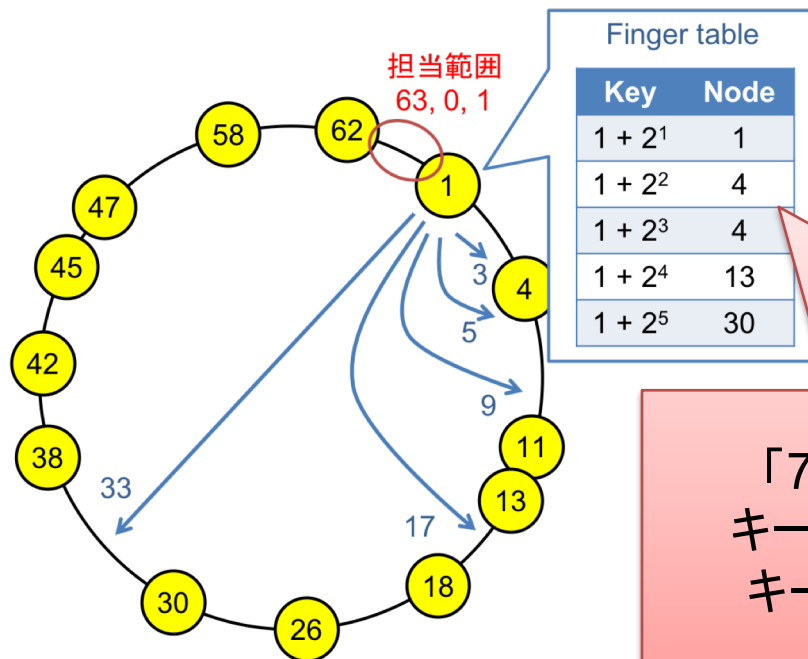
Chord (環状の代表的なDHT)

- 一次元のキー空間( $m$  ビット)
- Finger table: 各ノードから $2^1, 2^2, 2^3, \dots, 2^{m-1}$  ビット先までのショートカットリンク
- ノード探索のホップ数は  $O(\log N)$



# 問題点

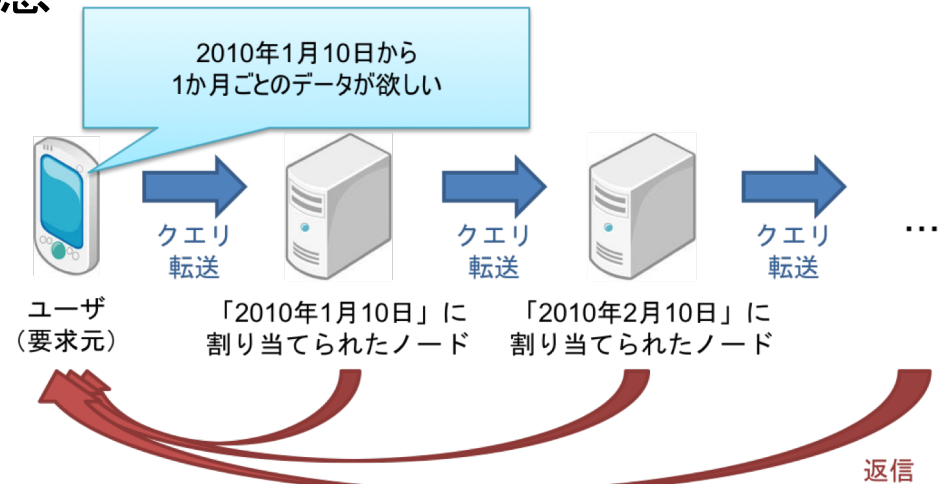
- 実際のユーザが多く要求する間隔クエリを既存手法は想定していない
- 次の間隔を担当するノードまで、多くのノードを経由してクエリを転送してしまう



「7日間(1週間)」がクエリとして指定されても、キーが7先のノードへのリンクを持っていないため、キーが5先のノードへクエリを転送することになる

# 目的

- 間隔クエリの効率的な処理を実現
  - 複数の異なる時間間隔を考慮
    - 年、月、週、日、時間など



- 実現するためのアプローチ
  - 複数の異なる時間間隔で、各ノードにショートカットリンクを割り当て
  - 対象ノードまでのホップ数を削減

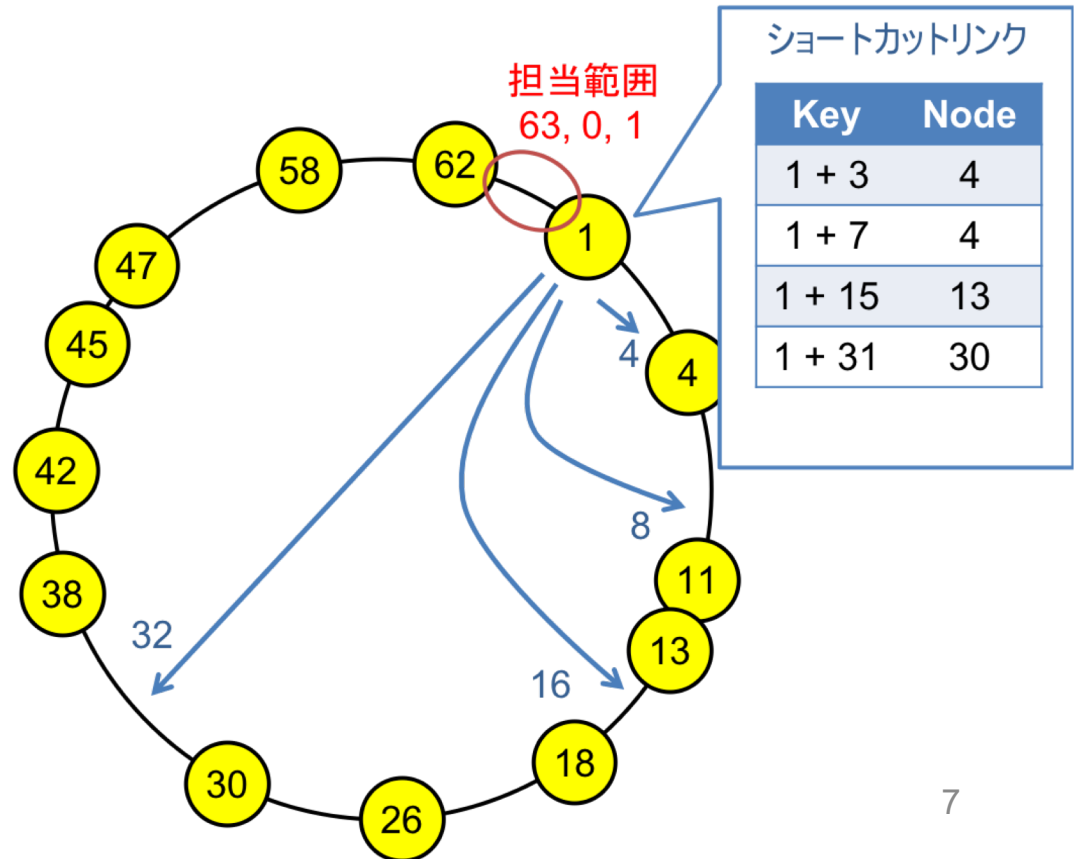
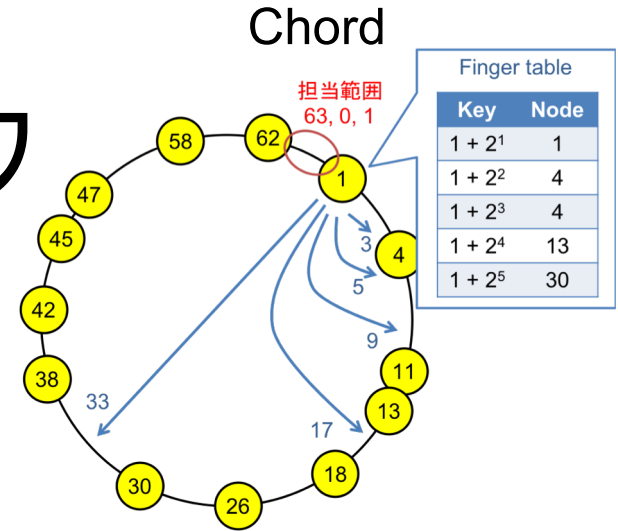
# 提案手法

- Chord同様、環状のトポロジを想定
- 時間に関する一次元の情報に基づいて、各ノードをキー空間へ割り当て
  - 一次元の情報は年や月、週、日、時間などを最小単位として表現
  - 協定世界時(UTC)やUNIX時間などと同様、キー空間は特定の長さ( $m$ ビット)を持つ
- 間隔クエリのホップ数を削減するため、各ノードは自身のキー位置から特定の間隔のショートカットリンクを構築
  - successorは1つ後ろのノードで、predecessorは1つ前のノード

種類	対象キー(間隔)	数
年	$2^0, 2^1, 2^2, \dots, 2^{m-1}$	$m$
月	6, 3, 1	3
日	15, 7 (曜日), 3, 1	4
時刻	12, 6, 3, 1	4
隣接	successor, predecessor	2

# ショートカットリンク

- 例として、最小単位を1日を設定
- キー空間の長さは64日間
- キー1に割り当てられたノードは以下の間隔のショートカットリンクを構築
  - 月: 1か月(31日)
  - 日: 15日、7日(曜日)、3日
  - 隣接: successor, predecessor
- 1か月後や1週間後の間隔クエリは1ホップで転送を実現
- ショートカットリンクの数(ルーティングテーブルの大きさ)は4で、Chordと比べて大きく変わらない



# シミュレーションによる評価

- キー空間の最小単位は1時間
- キー空間の長さは16ビット ( $2^{16} = 65536$ 時間)
- ノード数は1000～4000 (キーはランダム)
- 比較手法: Chordに類似のショートカットリンク(exponential)
- クエリパターン(1クエリで複数のキーを指定)
  - 12キー、720時間間隔(8640時間 = 360日)
  - 12キー、ランダム間隔(8640時間)
  - 52キー、168時間間隔(8736時間 = 52週)
  - 52キー、ランダム間隔(8736時間)
- 評価指標
  - ノード間のメッセージ数(通信負荷)、ホップ数(遅延)、ルーティングテーブルの大きさ(メンテナンスコスト)
  - 20回の試行で平均値を算出



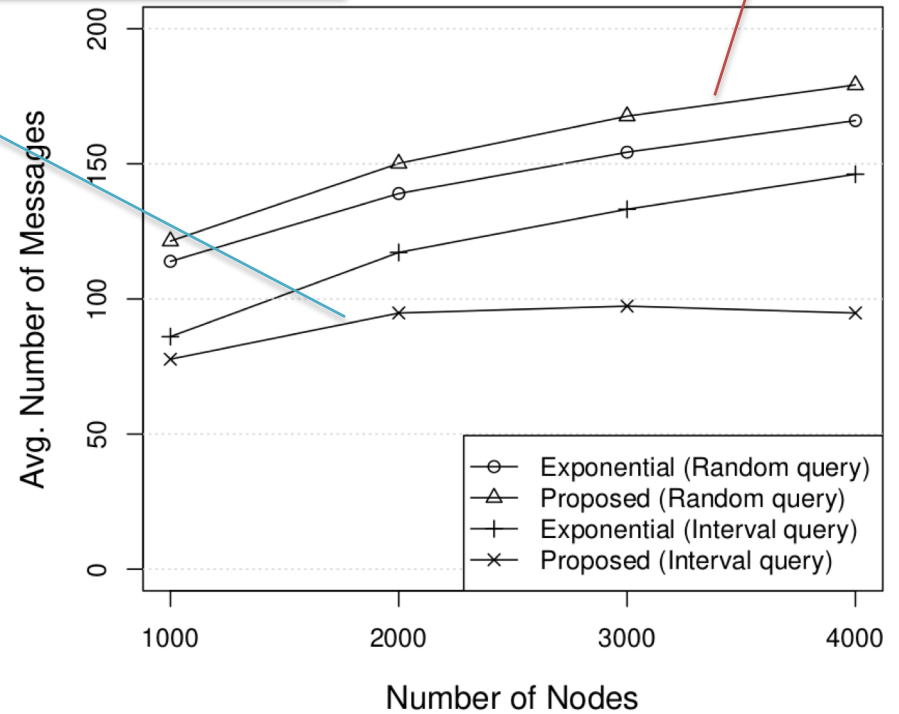
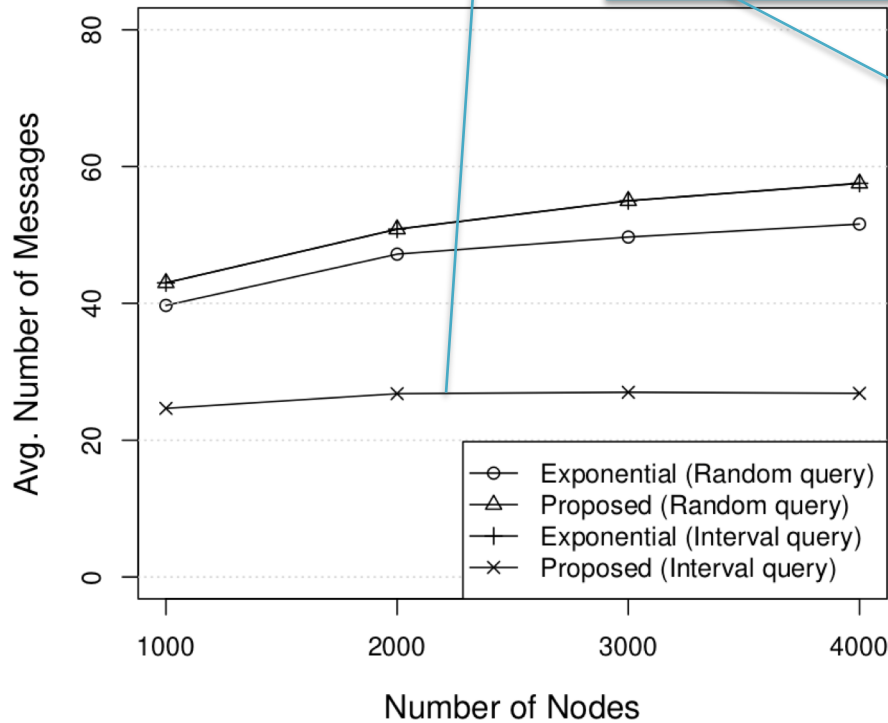
# 平均メッセージ数

クエリパターンがランダムの場合、  
提案手法は値がわずかに増加

クエリパターンが異なっても、間隔クエリでの  
提案手法の値が最小

12キー

52キー



実際のユーザが多く要求する間隔クエリに対して、  
提案手法は通信負荷を削減

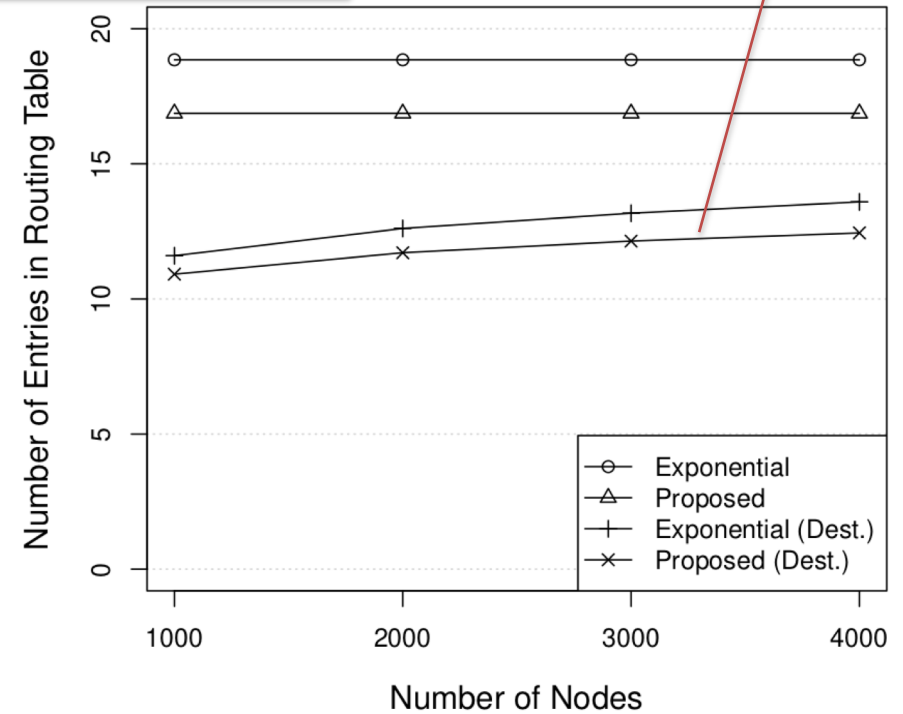
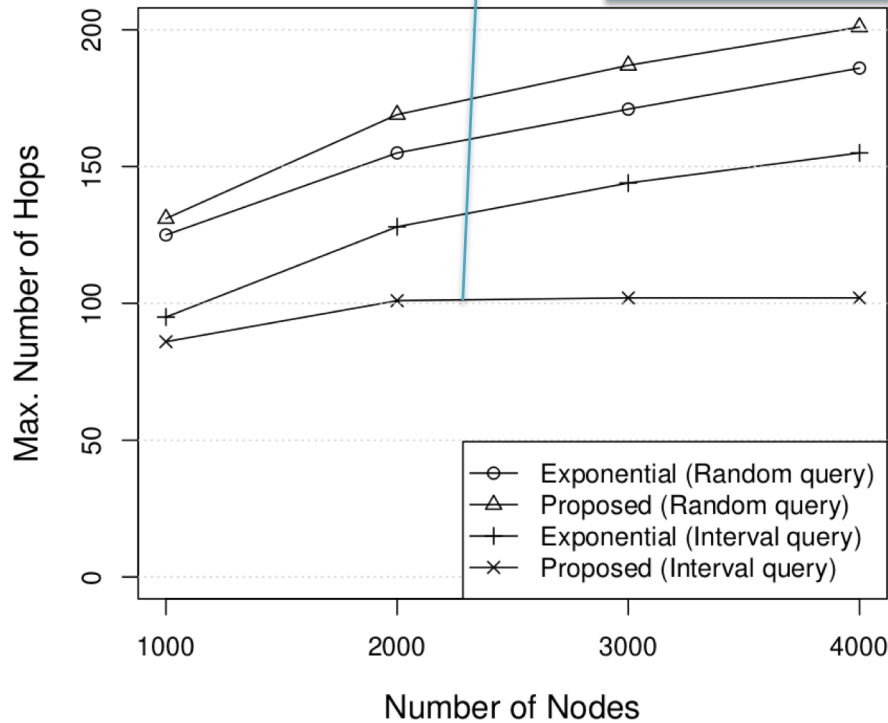
# ホップ数・ルーティングテーブルのコスト

エントリ数、宛て先数 (Dest.) とともに、  
提案手法の値が比較手法より小さい

最大ホップ数 (12)

提案手法では、ノード数による  
最大ホップ数が収束

ルーティングテーブルの平均サイズ



多数のノードが存在しても遅延を軽減し、  
定期的なメンテナンスが必要なルーティングテーブルのコストも抑制

# まとめ

- 複数の異なる時間間隔に基づく  
構造化オーバーレイネットワーク構築手法を提案・評価
  - 環状のトポロジを想定し、時間に関する一次元の情報に基づいて、各ノードをキー空間へ割り当て
  - 実際のユーザが多く要求する間隔クエリのホップ数を削減するため、各ノードは自身の位置から特定の間隔のショートカットリンクを構築
  - シミュレーションにより、提案手法が間隔クエリでメッセージ数や最大ホップ数、ルーティングテーブルサイズを削減できることを確認
- 今後の課題
  - ノード間の負荷の偏りを軽減するため、仮想ノードの仕組みの導入
  - オーバレイネットワークの到達性や耐障害性、信頼性の向上のため、割り当てノードやデータを複製する仕組みを検討