

NAIST において取り組みたい研究

1 はじめに

1.1 取り組みたい研究分野

私は NAIST でオーバーレイネットワークに関する研究に取り組みたいと考えている。

1.2 研究背景

オーバーレイネットワークは IP ネットワークなどのアンダーレイの上位に構築されるアプリケーション層のネットワークである。オーバーレイはアンダーレイにはない機能を持つネットワークを構築することができる。

2000 年以降、構造化オーバーレイによって実現する分散ハッシュ表 (distributed hash table, DHT) に関する研究が盛んに行われてきた。DHT とはハッシュ表を複数のノードで分散管理する技術で、ノードの参加や離脱が頻繁に生じるようなネットワーク上でのファイルシステムの構築やメッセージングを可能にする。大規模な分散システムの耐障害性やスケーラビリティを向上させるために必要不可欠な技術となっている。特にビッグデータ分析において DHT が果たす役割は非常に大きい。

DHT を実現するアルゴリズムの代表的なものに Chord [1], Kademlia [2] がある。また、DHT アルゴリズムの根本を見直し、その設計方針を明確にした柔軟な経路表 (flexible routing tables, FRT) [3] という DHT の設計手法がある。柔軟な経路表によって Chord, Kademlia をそれぞれ再設計した FRT-Chord, FRT-XOR も提案されている。これらの DHT ではハッシュ関数で決まるノード ID に基づいてトポロジを決定し、ノード数 N に対して $O(\log N)$ の経路長でのルーティングを可能にしている。

構造化オーバーレイはアンダーレイのトポロジや通信帯域、ネットワーク近接性などを無視したトラフィックを発生させてしまうという問題点がある。これに対処するための拡張アルゴリズムも数多く提案されている。

2 これまでに行った研究

2.1 オーバレイに関する研究

オーバーレイではシステムを構成しているノードの通信帯域や処理能力、信頼性などを考慮して適切な役割分担を行うべきである。

私は FRT-Chord に基づき、ネクストホップのノード状態を考慮する構造化オーバーレイを提案してきた [4]。これは $O(\log N)$ の経路長とノード状態の考慮を両立するような経

路表を構築し、高効率なフォワーディングを可能にするオーバーレイである。実際に、各ノードの最大経路表サイズを考慮する構造化オーバーレイを Overlay Weaver [5] 上に実装し、シミュレーション実験を行うことで、フォワーディング効率が向上することを示した。

2.2 防災情報システムに関する研究

オーバーレイと直接関係はないが、岐阜県情報技術研究所と共同で地理情報システム (geographic information system, GIS) の応用に関する研究も行ってきた [6]。この研究では、GIS への災害情報の登録を効率化するためにスマートフォンの特徴を生かしたアプリを開発した。

従来は県の専門家だけが災害情報の登録を行っていたが、この研究の成果によって登録手順が容易になり、一般の県民も防災リポータとして災害情報を発信できるようになった。GIS の利用が容易になったことで用途が広がり、県民自らが県民らに情報を発信できる基盤になりつつある。

3 今後の研究

3.1 方針

構造化オーバーレイではアンダーレイのトポロジを考慮し、不要なトラフィックを削減することが望ましい。そのため、グループ間通信を抑制するアルゴリズムや通信遅延を考慮するアルゴリズムなどが提案されてきたが、複数の状態を同時に考慮するアルゴリズムは提案されていない。

NAIST ではこれまでのオーバーレイに関する研究を活かし、ノードの属するグループと通信遅延の両方を同時に考慮する構造化オーバーレイの実現を目標に研究を進めようと考えている。

私は経路表構築とルーティングの両者が密接に関係してこそ理想的なオーバーレイが実現できると考えている。そこで、グループ情報を考慮した経路表と通信遅延を考慮した経路表を構築し、これらを用いて適切なルーティングを行うことで複数の状態の考慮を両立しようと考えている。

3.2 提案手法

FRT-Chord に基づいて DHT を設計する。図 1 に提案手法の概要図を示す。

ノードとオブジェクトにハッシュ関数で決まる ID を与え、consistent hashing に基づいて同一の ID 空間 $P = \{x \mid 0 \leq x < 2^m, x \in \mathbb{N}\}$ 上に配置する。ID 間の距離は次のように定義する。

$$d(x_1, x_2) = x_2 - x_1 \pmod{2^m} \quad (x_1, x_2 \in P) \quad (1)$$

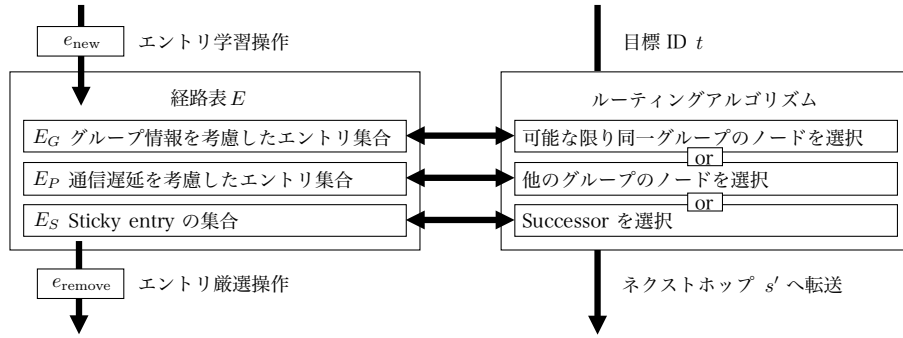


図 1 提案手法の概要図

各ノードは経路表 $E = E_S \cup E_G \cup E_P = \{e_i\}$ ($i = 0, \dots, |E| - 1$) を持つ。経路表のエントリはノード ID と IP アドレスの対で、ネクストホップの候補である。 E_S は Chord における successor list や predecessor に相当し、到達性を保証する役割を果たす。 E_G, E_P はそれぞれグループ情報を考慮したエントリ集合と通信遅延を考慮したエントリ集合で、構築方法は以降に述べる。

3.2.1 経路表の構築

次のエントリ学習操作とエントリ厳選操作を繰り返すことによって、グループ情報を考慮するように E_G を、通信遅延を考慮するように E_P を、そして $O(\log N)$ の経路長を実現するように E を改善する。

エントリ学習操作 エントリ学習操作では、ノード s が

- 同一のグループに所属する未知のノードと通信したとき新しいエントリ e_{new} を E_G に、
- 同一ではないグループに所属する未知のノードと通信したとき新しいエントリ e_{new} を E_P に、

追加する。この時、各エントリがノード s からの距離 $d(s, e_i)$ の昇順に並ぶように追加を行う。

エントリ厳選操作 エントリ学習操作で経路表サイズ $|E|$ が最大エントリ数 L を越えようとした時、エントリ厳選操作を行う。

$O(\log N)$ の経路長とグループ情報、通信遅延のすべてを考慮し、 E_G もしくは E_P から最も不要なエントリ e_{remove} を選択し、削除する。

3.2.2 ルーティングアルゴリズム

ノード s から目標 ID t へのメッセージはエントリ (ネクストホップ) $s' = \operatorname{argmin}_{e_i \in E} d(e_i, t)$ へのフォワーディングを再帰的に繰り返すことで到達する。このとき、 E_G を用いた同一グループ間のフォワーディングで可能な限り目標 ID t への残りの距離を削減し、最終的に E_P や E_S を用いてメッセージを到達させるようにする。

4 まとめ

私がこれまでに行ってきた研究と NAIST で取り組みたい研究について述べた。今後の研究においては、提案手法のエントリ厳選操作とルーティングアルゴリズムの具体的な手順の決定、そしてこれらの評価に取り組んでいきたい。

この研究に取り組むにあたって、NAIST に進学する理由は主に次の 2 つである。

一つは NAIST の刺激的な環境である。NAIST は大学院大学であり、「〇〇を研究したい」という明確な目標を持った学生が集まる。先生方の熱心なご指導を受けながら、周りの活気あふれる学生と切磋琢磨し、充実した研究活動を行うことができると考えている。

もう一つは NAIST の持つ研究設備である。大規模クラウド実験基盤システムでは、設計したオーバレイを実際に動かし、評価することができる。

私は NAIST の素晴らしい環境の中で、より効率的なオーバレイの実現を目指して研究を行いたいと考えている。

参考文献

- [1] I. Stoica, R. Morris, et al., “Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications,” *ACM SIGCOMM '01*, pp. 149–160, 2001.
- [2] P. Maymounkov, D. Mazieres, “Kademlia: A Peer to Peer Information System Based on the XOR Metric,” *IPTPS '01*, pp. 53–65, 2002.
- [3] H. Nagao, K. Shudo, “Flexible Routing Tables: Designing Routing Algorithms for Overlays Based on a Total Order on a Routing Table Set,” *IEEE P2P 2011*, pp. 71–81, 2011.
- [4] 白石 裕輝, 安田 真, “構造化オーバレイネットワークにおけるノード状態を考慮した経路表構成手法,” 情報処理学会全国大会論文集, Vol. 78, No. 3, pp. 163–164, 2016.
- [5] K. Shudo, Y. Tanaka, et al., “Overlay Weaver: An Overlay Construction Toolkit,” *Comput. Commun.*, Vol. 31, No. 2, pp. 402–412, 2008.
- [6] 白石 裕輝, 西中 智樹他, “スマートフォン向け岐阜県防災情報システムアプリの開発,” *FIT2014*, Vol. 13, No. 4, pp. 69–74, 2014.