

DHT における負荷分散を目的とした複製配置手法

A Replication Method for Load Balancing in DHT

高木邦孝 蘇洲 甲藤二郎
Kunitak Takagi Su Zhou Jiro Katto

早稲田大学理工学研究科
Graduate School of Science and Engineering, Waseda University

1. はじめに

近年 Peer-to-Peer(P2P)技術が注目されている。P2P を用いることによりサーバの負荷を増加させることなくコンテンツを多人数に送信することが可能である。その P2P 技術の一つとして分散ハッシュテーブル(DHT)がある[1,2]。

DHT では、システム共通の 1 つのハッシュ関数を用意し、ノードとコンテンツに一意の ID が割り当てられる。そして一般的にコンテンツは、コンテンツの ID に最も近い ID を持つノードが保有する。また DHT を用いることにより検索対象のコンテンツを保有しているノードを確実に発見することが可能となる。しかし、DHT ではノードの ID はランダムに割り当てられるため、ID が隣接するノードとの地理的な関係性がない。そのため、遅延や待ち時間などが大きくなる可能性がある。さらに、個々のノードの処理能力に関係なく、アクセス数が多いコンテンツを保持するノードに負荷がかかる問題点がある。そこで本稿では、以上の問題点を解決する複製配置手法を提案する。

2. 提案手法

2.1. クラスタ化

クラスタ化にはメトリックに地理的な近さを用い、図 1 に示すようにクラスタ ID を上位ビットに付加する。そうすることで、同じクラスタ内のノードとの遅延を減らすことができる。

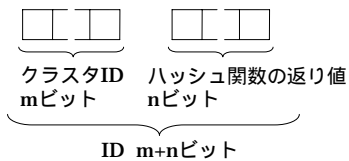


図 1 ID 空間

2.2. 複製配置

複製を ID からクラスタ ID を削った部分、つまり上位 m ビットを削った部分に該当するノード全てに配置する。そうすることで、複製を全てのクラスタに 1 つ作成できる。そして、コンテンツ取得の際には自分のクラスタ内のコンテンツを探索することで、自分のクラスタ内のみでルーティングが行われる可能性が高くなる。もし見つけることができなければ他のクラスタを探索する。

3. 評価

Overlay Weaver [3]を用いて実装を行った。比較対象として Pastry と Chord を取り上げ、その Pastry と Chord に提案方式を適用し、コンテンツ取得を行った際のホップ数の比較結果を図 2 に示す。自分のクラスタ内のコンテンツを

探索しているために、コンテンツを持っているノードの ID が自分の ID に近くなる可能性が高いため、ホップ数は少なくなっている。

次に従来手法としての Pastry、Chord と、それぞれに提案を適用しコンテンツ取得を 1000 回行った時の、アクセス数の分布を図 3 に示す。アクセスの集中を緩和し、アクセス数の偏りが小さくなっていることがわかる。

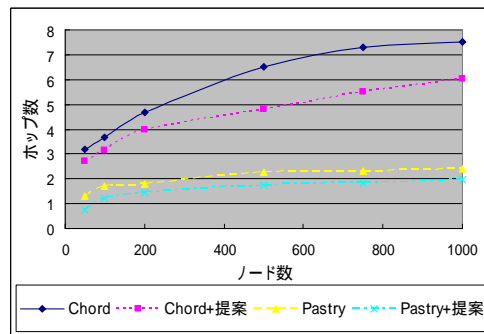


図 2 ホップ数の比較

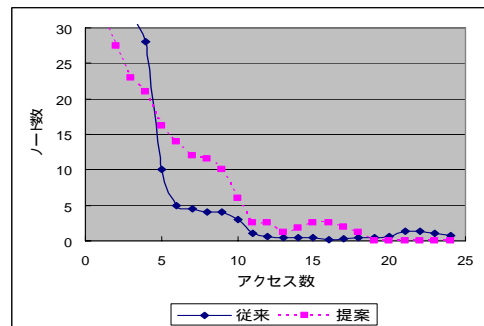


図 3 アクセス数の分布

4. まとめと今後の課題

地理的配置を考慮したクラスタ化を行い、複製を各クラスタに配置することで、ホップ数の減少とアクセス数の分散化を行うことができた。今後の課題としては、コンテンツの人気度を反映した複製配置やストレージの効率化について検討を進める予定である。

文献

- [1] Stoica, I., et al. "Chord: A scalable peer-to-peer lookup service for Internet applications" ACM SIGCOMM 2001.
- [2] A. Rowstron et al, "Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems" IFIP/ACM ICDS (Middleware), Nov.2001.
- [3] 首藤他, "オーバレイ構築ツールキット Overlay Weaver", 情報処理学会論文誌: Vol.47, No.SIG12 (ACS 15), pp.358-367, Sep.2006.