

アドホックルーティングにおける DHT の適用に関する検討

鳴海 寛之[†] 高橋 修[†]公立はこだて未来大学 システム情報科学部 情報アーキテクチャ学科[†]

〒041-8655 北海道函館市亀田中野町 116-2

1 はじめに

近年、無線通信技術の発達と移動無線端末（ノード）の小型化・高性能化によって、モバイルネットワーク環境が私たちの社会生活に普及しつつある。これに伴い、MANET (Mobile Ad-hoc NETWORK) に関する研究が活発に行われており、その研究分野のひとつにルーティングプロトコルに関する研究がある。MANET のルーティングプロトコルの代表的なものとして、リアクティブ型の DSR[1]と AODV[2]がある。これらのプロトコルは、経路構築の特性上、ノード数の多い環境においてエンド-エンド間の通信経路が長くなってしまった場合や、ノードが常に移動するような環境においては、データ到達性が低下する傾向にある。そのため実際に MANET において安定した通信を行うためには、ノード移動へ柔軟な対応能力やデータ到達性などといった、信頼性のさらなる向上が求められている。

そこで本研究では、インターネット上でピアツーピアシステムを構築する際に用いられる、効率的なデータ分散やスケーラビリティを実現する DHT (Distributed Hash Table) に着目し、DHT によるオーバーレイルーティングの特性を既存のアドホックルーティングプロトコルに適用することによって、従来よりもさらに高い信頼性を提供するアドホックルーティングプロトコルの実現について検討する。

2 関連研究

2.1 DHT

DHT は、2001 年頃に提案された、Pure-P2P ネットワークを構築するためのアルゴリズムの総称である。代表的なアルゴリズムに、Pastry[3]と Kademlia[4]がある。DHT は物理ネットワーク上にオーバーレイネットワークを構築する。そしてオーバーレイ ID 空間上に一様に分散配置したノードが協調動作することで ID 空間全体をカバーし、任意のノードへのルーティングが可能となる。DHT のオーバーレイルーティング方式には Iterative ルーティングと Recursive ルーティングがあり、Pastry は Recursive ルーティングを、Kademlia は Iterative ルーティングをそれぞれ採用している (図 1)。

・ Recursive ルーティング

Recursive ルーティングでは、ルーティングの開始ノードが適切なノードに転送を依頼し、依頼を受けたノードはさらに他のノードへ転送を依頼するという操作を繰り返すことで最終的な宛先まで転送するという方法をとる。

・ Iterative ルーティング

Iterative ルーティングでは、ルーティングの開始ノードがルーティングの最初から最後までを制御する。宛先ノードのアドレスを反復的にオーバーレイネットワーク上のノードに問い合わせ、アドレス取得後は宛先ノードと直接データのやり取りを行う。

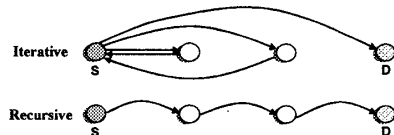


図 1 Iterative ルーティングと Recursive ルーティング

2.2 MANET オーバレイルーティング

MANET 上で DHT によるオーバーレイネットワークを構築し、オーバーレイルーティングを利用することで信頼性向上を図る関連研究として、MADPastry[5]および Ekta[6]がある。MADPastry ではアドホックルーティングプロトコルに AODV を用い、その上位層として

Pastry を用いてオーバーレイネットワークを構築している。また、Ekta ではアドホックルーティングプロトコルに DSR を使用し、DHT は同じく Pastry を用いている (表 1 参照)。これらの研究では、アドホックルーティングプロトコルの選択が汎用的ではなく、また、Pastry を利用することからオーバーレイルーティングによるオーバーヘッドが発生する。これらの点について解決することが課題となる。

3 提案方式

3.1 基本方式

本提案方式では、MANET 上に Kademlia を用いてオーバーレイネットワークを構築し、オーバーレイネットワーク上からアドホックルーティングを制御することで信頼性向上を図る。この制御は、Kademlia の Iterative ルーティングをベースとして近傍性や Recursive ルーティングの機構を組み込むことで実現される新たなオーバーレイルーティングアルゴリズムによって行われる。このアルゴリズムを新たなルーティングプロトコルとして実装する。これにより、ノード数が多いネットワーク環境下における長経路通信時の信頼性向上という課題を解決する。また、関連研究において達成されていない、プロトコル汎用性およびオーバーヘッドの課題を解決する。提案方式は、以下のような構成をとる。

(1) **Kademlia**: 本提案方式では DHT を用いて MANET 上にオーバーレイネットワークを構築する。各種 DHT アルゴリズムの特徴を検討した結果、Kademlia はノードのネットワークへの頻繁な参加/離脱を考慮し設計されたものであり、ノードがオーバーレイネットワークから離脱する際の特別な処理が必要ないという特徴を持つ点が、後述する近傍性を実現することに最も適していると判断し、本提案方式では Kademlia を用いることとした。

(2) **近傍性の実現**: 一般的なオーバーレイネットワークでは、ネットワークにおけるノードの位置をサービス提供者が定義できるが、DHT を用いて構築したオーバーレイネットワークでは、ノードはオーバーレイ ID 空間上に一様に分散配置される仕組みとなっている。このため、オーバーレイネットワーク上の隣接ノードが実際の物理ネットワークにおいて互いに遠くに位置するという状況が頻繁に起こり得る。つまり、MANET 上で DHT によるオーバーレイルーティングを行うと、通常のアドホックルーティングを利用する際に使用する経路と比べて“遠回り”する可能性が高い (図 2)。

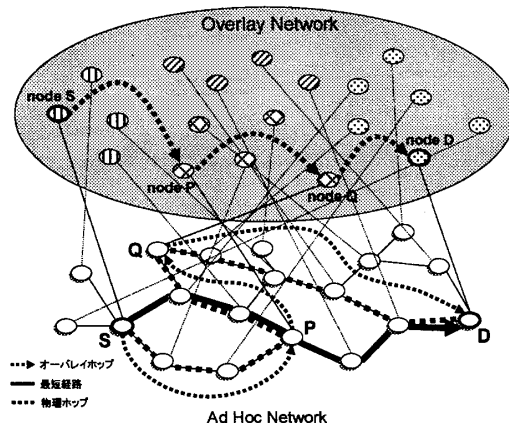


図 2 一様分散配置によるオーバーレイホップと物理ホップ

このため本提案では、DHT によるオーバーレイルーティングの特性を有効利用するため、図 3 で示すように、物理的に互いに近いノード

Study on DHT-based Ad Hoc Routing Protocol

† Hiroyuki Narumi · Future University-Hakodate

† Osamu Takahashi · Future University-Hakodate

をできる限り論理的にも近い位置に配置させるという近傍性を実現することが必要となる。本提案方式では、代表的手法である RLM (Random Landmarking) [7]を用いることで近傍性を実現する。RLMでは、オーバーレイネットワークを複数のクラスタに分割し、それぞれにクラスタ ID が割り当てられる。そして各クラスタ内には常時 landmark ノードと呼ばれるクラスタ内の代表者が存在し、このノードがクラスタ ID を現在クラスタ内に存在するノードに対して定期的に通知する。これにより、MANET 上でノードが移動すると、オーバーレイネットワーク上ではクラスタを跨ぐことになり、そこで通知されるクラスタ ID を元に、ノードは自分のノード ID を自立的に更新する。これによって、MANET 上でノードが移動した場合でも近傍性が維持される(図 4)。こうした近傍性を踏まえたノード配置を行うことで、オーバーレイネットワークのトポロジが MANET と似た構成となる。これにより、オーバーレイホップを構成する物理経路が(通常のアドホックルーティングがとるような)最短経路に近い経路をとるようになる。そして DHT の Recursive ルーティングの特性を利用すると、1つの長経路を複数の短経路(図 3 では 3 つ)に分割することができる。これによって、ノード移動に伴う経路切断が従来よりも発生しにくくなり、信頼性を向上できる。

(3) **Kademlia の Recursive ルーティング化**: 本提案方式では DHT に Kademlia を用いるが、図 3 に示すように、オーバーレイルーティングの方式には Recursive ルーティングを用いる必要がある。この課題は、Iterative ルーティングを必要とするテーブルメンテナンス等の Kademlia のコア処理以外の、データ転送などの処理のみに Recursive ルーティングの機構を導入することによって解決する。これにより、ノードの高速移動への対応能力などといった Kademlia の特徴を継承したまま Recursive ルーティングを行うことができるようになり、本研究の目的であるオーバーレイルーティングの特性を利用した、アドホックルーティングの信頼性向上を図ることができる。

3.2 レイヤ構成

TCP/IP 参照モデルにおける本提案プロトコルの位置づけを図 5 に示す。本提案プロトコルは、既存のアドホックルーティングプロトコルの拡張機能という位置づけであるため、それらの上位に位置する。アドホックルーティングプロトコルの上位に近傍性を実現する RLM と Kademlia が動作する。本提案プロトコルはアドホックルーティングをオーバーレイネットワーク上から制御する方式をとるため、下位のアドホックルーティングプロトコルに対して汎用性を持たせることができ、DSR や AODV など(リアクティブ型)を任意に選択することが可能である。

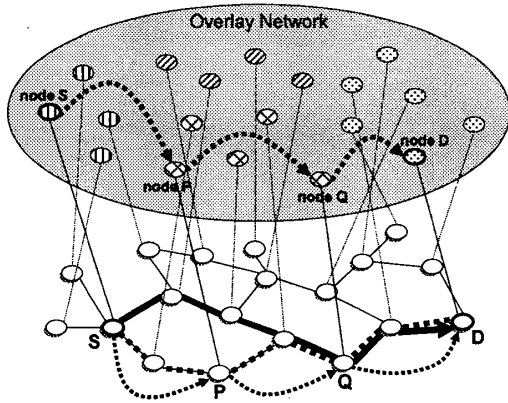


図 3 提案方式

4 提案方式の基本評価と考察

表 1 に示すように、提案方式では、2つの関連研究とは異なるアプローチとして、Kademlia に RLM を適用し近傍性を実現する。RLM ではノード ID を更新する際に一度オーバーレイネットワークから離脱し、再度ネットワークに参加するという手順を踏む必要がある。そこで前述した離脱処理が不必要であるという Kademlia の特性を利用

することにより、MADPastry よりもノードの高速移動に柔軟に対応できるという見込みをもつことができる。また、Kademlia はオーバーレイネットワークを維持するための専用の処理も必要としなという特長もあり、基本的にネットワーク維持のためのオーバーヘッドが発生しない。これに対して、Pastry はネットワーク維持の専用処理によってネットワーク上に制御メッセージが流れる。このことから、本提案方式は MADPastry などよりもオーバーヘッドを低減できる見込みがある。

表 1 関連研究と提案方式の比較

	MADPastry	Ekta	提案方式
DHT	Pastry	Pastry	Kademlia
アドホックルーティングプロトコル	AODV	DSR	リアクティブ型
近傍性の実現の有無	RLMにより実現	無し	RLMにより実現
特徴	AODVを利用しているため、比較的高速なノード移動に対応できるが、Pastryを利用するためノード離脱時の処理が必要となり、メッセージオーバーヘッドが発生する。	十分な近傍性を実現していないため、ノード数が多い場合に経路切断が発生しやすく、データ到達率に影響する。Pastryを利用するためメッセージオーバーヘッドが発生する。	Kademliaを利用するため基本的にメッセージオーバーヘッドが発生しない。近傍性の実現によってノード数が多い場合にも安定したルーティングが可能。ノード離脱の処理が必要ないため、ノードの高速移動にも対応可能。

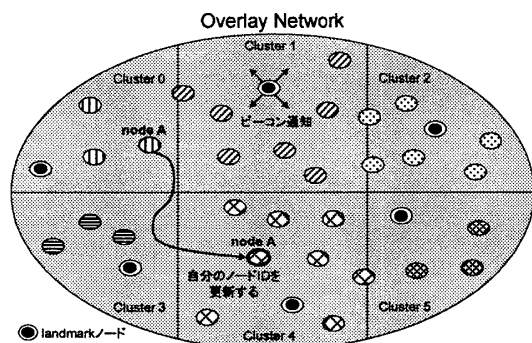


図 4 RLM による近傍性の実現

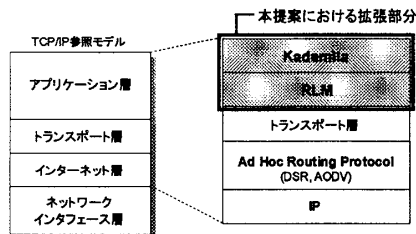


図 5 本提案プロトコルの位置づけ

5 おわりに

アドホックルーティングの信頼性向上に関して、本研究では Kademlia が持つ DHT のオーバーレイルーティングの特性に着目し、提案プロトコルの設計および基本評価を行った。今後は本提案プロトコルの実装・評価を行う。

参考文献

- [1] DSR (Dynamic Source Routing) : RFC4728
http://www.ietf.org/rfc/rfc4728.txt
- [2] AODV (Ad hoc On-Demand Distance Vector) : RFC3561
http://www.ietf.org/rfc/rfc3561.txt
- [3] A. Rowstron and P. Druschel. "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems". In Proc. of Middleware. November 2001.
- [4] Petar Maymounkov and David Mazières, "Kademlia: A Peer-to-peer Information System Based on the XOR Metric". IPTPS02. March 2002
- [5] Thomas Zahn and Jochen Schiller. "DHT-based Unicast for Mobile Ad Hoc Networks". In Proc. of IEEE Percom. March 2006.
- [6] H. Pucha, S.M. Das, and Y.C. Hu. "Ekta: An Efficient DHT Substrate for Distributed Applications in Mobile Ad Hoc Networks". In Proc. of IEEE WMCSA. December 2004.
- [7] R. Winter, T. Zahn, and J. Schiller. "Random Landmarking in Mobile, Topology-Aware Peer-to-Peer Networks". In Proc. of FTDCS. May 2004.