

# トラヒック状況を考慮したアドホックルーティングプロトコルの検討

森崎明<sup>†</sup> 伊藤将志<sup>††</sup> 渡邊晃<sup>†</sup>

無線 LAN を標準搭載した携帯端末の普及に伴い、無線端末のみでネットワークを構築するモバイルアドホックネットワーク(MANET : Mobile Ad-hoc Network)の研究が注目されている。MANET で提案されている多くのアドホックルーティングプロトコルは、経路生成の際に経路上のトラヒック状態が考慮されていないため、中継ホップ数が最短であれば比較的負荷の高い経路でも選択してしまうという課題がある。本論文では OLSR(Optimized Link State Routing)を拡張することにより、経路上のトラヒックを考慮した経路生成が可能なアドホックルーティングプロトコルを提案する。

## Researches on an Ad-hoc Routing Protocol considering Traffic Conditions

AKIRA MORISAKI<sup>†</sup> MASASHI ITO<sup>††</sup>  
AKIRA WATANABE<sup>†</sup>

With the spread of mobile nodes, the study of MANET (Mobile Ad-hoc Network) that can build networks only with mobile nodes is paid much attention. However, most of ad-hoc routing protocols have not considered about traffic conditions in the network. We propose an ad-hoc routing protocol considering traffic condition by extending OLSR (Optimized Link State Routing).

### 1. はじめに

無線 LAN を標準搭載した携帯端末が急速に普及してきている。これに伴い、無線端末(以後、ノード)のみで自律的にネットワークを構築するモバイルアドホックネット

ワーク(MANET : Mobile Ad-hoc Network)の研究が注目されている。

MANET のルーティングには無線通信に特化したルーティングプロトコルが使用され、一般にアドホックルーティングプロトコルと呼ばれている。今まで多くのアドホックルーティングプロトコルが提案されているが、ほとんどが経路生成の際に中継ホップ数が最短となる経路(以後、最短経路)を選択する。しかし、MANET では集中的に通信の中継を行うノードが発生する可能性が高い。このような負荷の高いノードを含む最短経路では遅延が増えたり、バッテリーが消費され、リンクが切断される可能性がある。リンクの切断が発生すると経路再構築が必要となり、スループットが大きく低下する。このため、単純に最短の経路が最善な経路であるとは限らない。

スループットの向上を目的とするアドホックルーティングプロトコルの研究には以下のものが挙げられる。ABR (Associativity-Based Long-lived Routing) [10]の経路選択では、リンク切断が長時間起こらない安定した経路を選択する。各ノードは一定間隔ごとに隣接ノードへビーコンを送信する。より多くのビーコンを受信したノードからなるリンクは持続性が高いと期待されるため、安定した経路により通信を行うことができる。しかし、ノードの移動が少ない環境では、ビーコンの受信回数に差が生じないため、スループットの向上が期待できない経路が選択される可能性がある。

ETR(Estimated-TCP-Throughput Maximization based Routing)[11]は DSR を拡張することにより、宛先への複数の経路候補に対してスループットを予測し、スループットの高い経路を選択する。スループットは Floyd らが提案したモデル式を使って計算される。モデル式には遅延(RTT: Round-Trip Time)と往復パケット喪失率(RTPL: Round-Trip Packet Loss ratio)の情報が必要であり、これらの情報を収集するために新たな制御メッセージを設け、一定間隔で送信する。しかし、制御メッセージにより、ネットワークのオーバーヘッドが高くなるという課題がある。

本論文ではアドホックルーティングプロトコルの一方式 OLSR(Optimized Link State Routing)を拡張することによって、経路ごとの負荷を計算し、負荷の低い経路を選択するアドホックルーティングプロトコルを提案する。

以下、2 章では MANET のルーティングプロトコルの分類を示し、3 章で OLSR の概要について説明する。4 章では OLSR の拡張方法を説明し、最後に 5 章でまとめを行う。

### 2. アドホックルーティングプロトコルの分類

MANET では電波到達範囲外のノードと通信するために中継機能を持ち、ノードの移動によるリンク接続状態の変化に迅速に対応する必要がある。MANET には様々な用途が考えられ、用途に応じたルーティングプロトコルが必要である。これまで様々なアドホックルーティングプロトコルが検討されているが、全ての環境に適するプロ

<sup>†</sup>名城大学大学院理工学研究科

Graduate School of Science and Technology, Meijo University

<sup>††</sup>株式会社東芝研究開発センター

Research and Development Center, Toshiba Corporation

トコルは開発されていない。

これまでに開発されたアドホックルーティングプロトコルは、表 1 に示すように 3 つの型に分類することができる。これらは、その特徴が活かせる環境によって使い分けられる。

表 1 MANET のルーティングプロトコル

分類	プロトコル例
Proactive 型	OLSR, DSDV, TBRPF
Reactive 型	AODV, DSR, TORA, ABR
Hybrid 型	ZRP

### (1) Proactive 型

Proactive 型のルーティングプロトコルは、通信の要求が発生する前からルーティングテーブルを生成しておく方式で、通信の要求が発生すると即座に通信を開始できる。各ノードはルーティング情報を格納するためのテーブルを 1 つ以上持ち、ネットワークプロトコルの変化に応じてネットワーク全体に経路の更新情報を配信する。ルーティングに必要なテーブル数と、ネットワークの構造の変化を知らせるブロードキャスト方式の違いにより、いくつかのプロトコルが存在する。Proactive 型のルーティングプロトコルの特徴として、無通信時にも制御パケットが流れるため、消費電力は大きくなるが、通信を開始する際に遅延が発生しないことから、通信頻度の高いネットワークに適することが挙げられる。

### (2) Reactive 型

Reactive 型のルーティングプロトコルは、オンデマンド型のプロトコルである。すなわち、あるノードにおいて宛先ノードへの経路が必要になった時点で、ネットワーク内で経路探索プロセスを始動する。このプロセスは経路が見つかるか、利用可能なすべての経路パターンを試し終えると終了する。いったん経路が発見され、確立すると宛先へのアクセスができなくなるか経路が不要になるまでは、その経路が維持される。Reactive 型のルーティングプロトコルの特徴として、通信時に経路を決定するまでの遅延が発生が、オンデマンドで経路を構築するために、ノードの移動が頻繁なネットワークに適することが挙げられる。

### (3) Hybrid 型

Hybrid 型のルーティングプロトコルは、Proactive 型と Reactive 型の両方の長所を取り入れた複合プロトコルである。ネットワーク内を複数のゾーンに分割し、ゾーン内では Proactive 型のプロトコルを使用し、定期的な経路情報の更新はゾーン内のノードについてのみ行う。宛先ノードが送信元のゾーン外にある場合は Reactive 型のプロトコルを用いて経路を構築する。Hybrid 型ではこのように両方の特徴を生かすことがで

きるが、ノードが密集するような場合においてはゾーン内の管理すべきノードが多くなり、トポロジー管理が難しいという課題がある。

本論文では、最短経路の中から、トラヒック量が最少となる経路を選択し、スループットの低下を防止する方式を提案する。

経路上のトラヒック量はルーティングプロトコルが働いている間にも刻々と変化していく。MANET のルーティングプロトコルを拡張する上で、この変化に対応していくことが重要である。Reactive 型のプロトコルでは一度確立した経路は、宛先へのアクセスができなくなるか経路が不要になるまでは再計算が行われない。そのため、トラヒック量の変化に対応するのは難しい。一方、Proactive 型のプロトコルでは定期的にルーティングテーブルを更新することができる。よって、本論文では、Proactive 型のルーティングプロトコルを検討対象とし、その中の代表的かつ最も普及している OLSR を提案方式の対象とした。

## 3. OLSR

OLSR(Optimized Link State Routing)は INRIA のプロジェクト Hipercion[12]で提案された MANET を構築する Proactive 型のルーティングプロトコルである。以下に OLSR の概要を示す。

### 3.1 隣接ノードの発見

各ノードは HELLO メッセージを定期的にブロードキャストする。HELLO メッセージの送信間隔のデフォルト値は 2 秒である。HELLO メッセージには自身のアドレス、シーケンス番号、隣接ノードのアドレスなどの情報が入っている。このため、HELLO メッセージを受信したノードは隣接ノードのアドレス及び隣接ノードの更に隣接ノード、すなわち 2 ホップ先のノード(以後、2 ホップ隣接ノード)のアドレスを得ることができる。また、受信した HELLO メッセージの隣接ノードアドレスの中に自身のアドレスが含まれていれば、自分が送信した HELLO メッセージを隣接ノードが受信したことが確認できる。このことは自身と隣接ノード間で双方向に HELLO メッセージの送受信が可能ということであり、このようなリンクを双方向リンクと呼ぶ。一方、受信した HELLO メッセージの隣接ノードアドレスの中に自身のアドレスが含まれていなければそのリンクは非双方向リンクの状態と認識される。これらリンクの状態も HELLO メッセージに含めて送信される。

図 1 ではノード A の HELLO メッセージをノード B が受信し、ノード C は受信失敗となっている。その後、ノード B からの HELLO メッセージがノード A と C によって受信され、その中にノード A のアドレスが含まれていることを検知することにより、ノード A は AB 間のリンクを双方向リンクと認識する。また、ノード C はノード A を 2 ホップ隣接ノードと認識する。さらにその後、C から送信された HELLO メッセージ

がノード A で受信されると、その中にはノード A のアドレスが含まれていないため、ノード A は AC 間のリンクを非双向リンクと認識する。

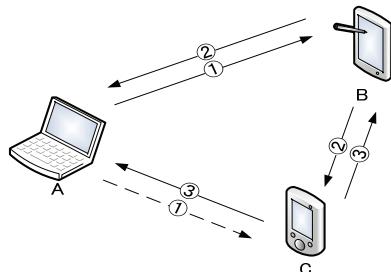


図 1 HELLO メッセージの送受信

### 3.2 OLSRのフラッディング方式

OLSR の最大の特徴として、効率の良いフラッディングが挙げられる。フラッディングとは、各ノードが自身の情報をネットワーク内の全てのノードへ配信することである。通常のフラッディングでは、送信元ノードはメッセージを隣接ノードへブロードキャストする。それを受信した隣接ノードはブロードキャストを繰り返し、すべてのノードにメッセージを中継する。同じメッセージを重複して受信した場合は、そのメッセージを破棄する。しかし、通常のフラッディングでは、ブロードキャストの総数が増大し、同一メッセージの重複受信も増大する。

OLSR では必要最低限の中継ノード(MPR : Multipoint Relay)を定義し、この中でのみフラッディング動作を行うことにより、すべてのノードにメッセージを届ける。各ノードは自身の MPR を選択すると、その情報を HELLO メッセージで隣接ノードに通知する。これを受信した各ノードは自身を MPR として選択しているノードを認識できる。このようなノードを MPR セレクタと呼ぶ。これにより、各ノードは自身の MPR セレクタからのメッセージのみを中継する。このようにして、ブロードキャストの総数を減少させ、同一メッセージの重複受信を減少させる。

### 3.3 トポロジー情報の配達

OLSR では、トポロジー情報を定期的に TC(Topology Control)メッセージによってフラッディングする。TC メッセージを生成するのは MPR のみである。TC メッセージの送信間隔はデフォルト値で 5 秒である。TC メッセージには自身のアドレス、シーケンス番号、自身の MPR セレクタのアドレスなどの情報が入っている。TC メッセージによって配達されるトポロジー情報は全てのリンクから構成されるトポロジーではなく、各ノードの MPR セレクタから構成されるトポロジーのみである。

### 3.4 その他のメッセージ

OLSR には、HELLO メッセージ、TC メッセージ以外に MID(Multiple Interface Declaration)メッセージと HNA(Host and Network Association)メッセージがある。MID メッセージはノードが複数のインターフェースを有する場合にのみ使用され、HNA メッセージはノードがゲートウェイとして機能する場合に使用される補助的なメッセージである。本論文の提案方式では MID メッセージ、HNA メッセージに手を加えないため、これら制御メッセージの説明は省略する。

### 3.5 各ノードが持つ情報

各ノードは図 2 に示す 7 つのテーブルからなるリポジトリを持つ。これらのテーブルは隣接ノードだけに届く HELLO メッセージ、ネットワーク全体にフラッディングされる TC メッセージによって生成される。

リンク集合はローカルノード自身のアドレス、隣接ノードのアドレス、リンクが双向とみなされる時間、レコードの生存時間から構成される。隣接ノード集合は隣接ノードのアドレス、リンクが双向か非双向であるかの状態、MPR として選択されるための指標(willingness)から構成される。2 ホップ隣接ノード集合は隣接ノードのアドレスと双向の 2 ホップ隣接ノードのアドレス、レコードの生存時間から構成される。MPR 集合は MPR として選択されたノードのアドレスとレコードの生存時間から構成される。MPR セレクタ集合は MPR セレクタとして選択されたノードのアドレスとレコードの生存時間から構成される。トポロジー集合は宛先となるノードのアドレス、宛先へ 1 ホップで到達できるノードのアドレス、レコードの生存時間から構成される。複製集合は受信したメッセージの重複した処理を避けるために設けられるテーブルである。

図 2 を用いてノードの動作を説明する。HELLO メッセージを受信したノードはリポジトリ内のリンク集合、2 ホップ隣接ノード集合、MPR セレクタ集合、複製集合を更新する。また、リンク集合、2 ホップ隣接ノード集合の更新に伴い、隣接ノード集合と MPR 集合も更新する。一方、TC メッセージを受信したノードはトポロジー集合と複製集合を更新する。これらの更新されたテーブルを基に新しい HELLO メッセージ及び TC メッセージを生成する。また、TC メッセージは MPR セレクタ集合と複製集合を基に中継される。さらに、隣接ノード集合、2 ホップ隣接ノード集合、トポロジー集合の情報を基にルーティングテーブル(以後、RT)を生成する。

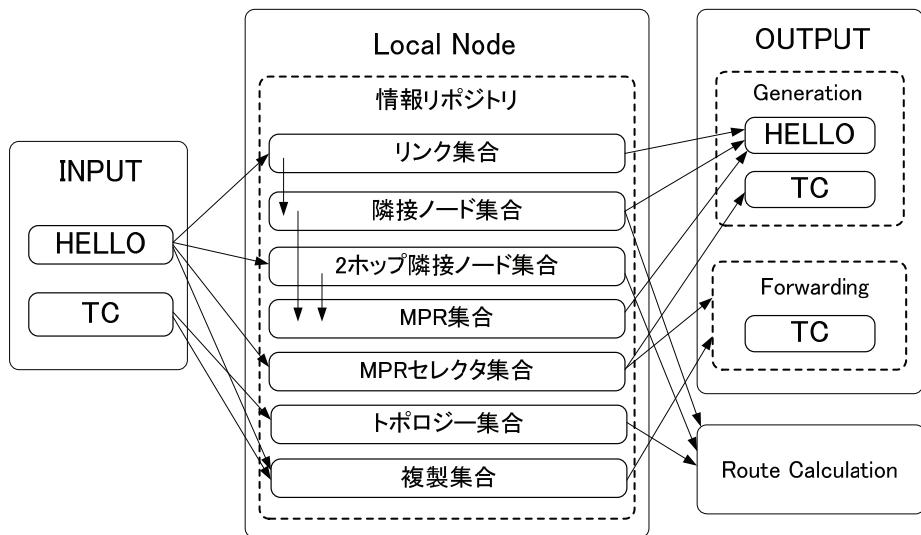


図 2 制御メッセージと情報リポジトリの関係

### 3.6 経路計算

OLSR の RT は、宛先ノード(Dest), Dest への次ホップノード(Next), Dest までのホップ数(hop)から構成され、各 Dest に対して 1 つの経路を保持する。以下に OLSR の経路生成の方法を示す。図 3 はノード s が持つ RT にノード a~d までの経路が既に作成された状態から、ノード e への経路を新たに追加する過程を示している。Dest が e となるレコードの Next には e の隣接ノードである c, d のうち最初に発見されたノード c のレコードの Next の値(a)が設定される。ノード a~d の RT においても同様の方法で e への経路が決まり、図 3 に示すように、s→a→c→e という 1 つの最短経路が完成する。

しかし、この方法では単純に最初に発見された最短経路が選ばれる。この経路が図 3 のように負荷が高く通信状態が悪いリンクから成る経路であった場合、ノードの処理によるオーバーヘッドやパケット損失によるスループットの低下が発生する。

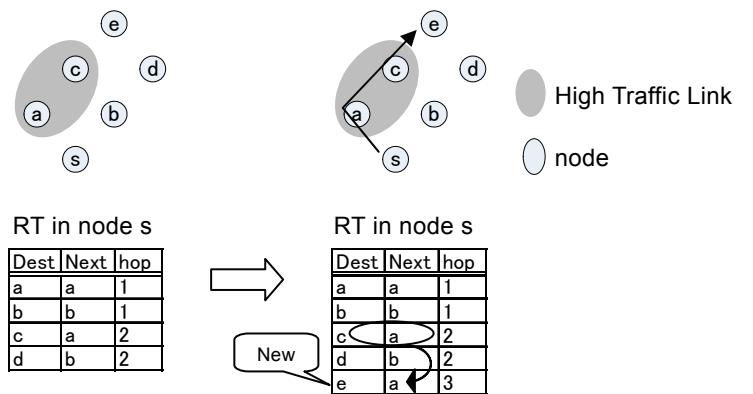


図 3 OLSR による RT 生成方法

### 4. 拡張OLSR

提案システムでは既存の OLSR に対して、トラヒックを考慮した経路選択を行う。そのため、リポジトリ内の関連テーブル（リンク集合、隣接ノード集合、2 ホップ隣接ノード集合、トポロジー集合）にトラヒック量の項目を追加する。また、宛先ノードへの複数の最短経路の合計トラヒックを計算した経路計算テーブル(RCT : Route Calculation Table)を新たにリポジトリに追加する。RCT は、Dest, Dest への経路(Route), hop, Dest への経路の合計トラヒック(Traffic)から構成される。さらに、各ノードは HELLO メッセージ及び TC メッセージを送信するときに、トラヒック情報をメッセージに付加する。これらを受信したノードはリポジトリ内のテーブルをトラヒック情報と共に更新する。最終的な RT はトラヒック情報を含むリポジトリの内容から生成する。

図 4 は OLSR における制御メッセージの処理の流れを基に拡張 OLSR の動作を具体的に示したものである。吹出し①～⑤では以下に示す拡張動作を行う。

#### ① 制御メッセージの送信

HELLO メッセージ及び TC メッセージの送信元ノードは自身のトラヒック量をメッセージに付加し、送信する。

#### ② リンク集合の更新

HELLO メッセージの送信元ノードと一致する隣接ノードのレコードに送信元ノードのトラヒック量を記録する。一致するレコードが存在しないときは、新たに送信元ノードを隣接ノードとするレコードを作成する。

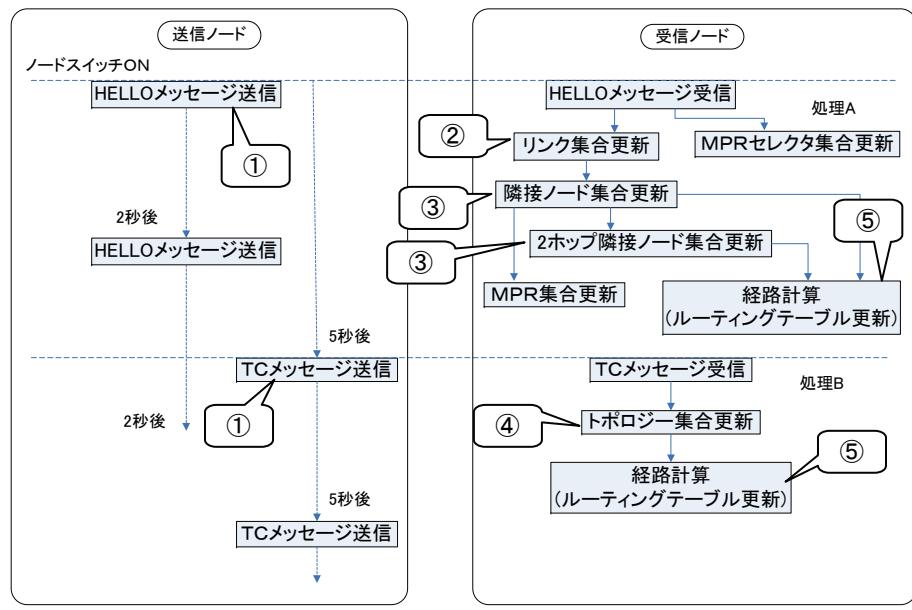


図 4 拡張 OLSR の動作

- ③ 隣接ノード集合と 2 ホップ隣接ノード集合の更新  
②の更新と対応する隣接ノードのレコードにそのトラヒック量を記録する。
- ④ トポロジー集合の更新  
TC メッセージの送信元ノードと一致する宛先ノードのレコードに送信元ノードのトラヒック量を記録する。一致する宛先ノードが存在しないときは、新たに送信元ノードを宛先ノードとするレコードを作成する。
- ⑤ 経路計算  
経路計算( RT 更新)に先立ち、RCT を生成する。RCT は 3.6 項の経路計算と同様に作成されていくが、宛先ノードへの複数の経路を保持するため、これらの経路を区別するために経路全体を記録する。RCT の生成が完成すると、RCT の中から最少のトラヒック量を持つ経路を抽出し RT を生成する。

図 5 に具体的な拡張 OLSR の経路生成方法を示す。図 5 は、図 3 と同じようにノード s に着目した経路生成方法を示している。各ノード横の数字は、ノードのトラヒック量を表す。

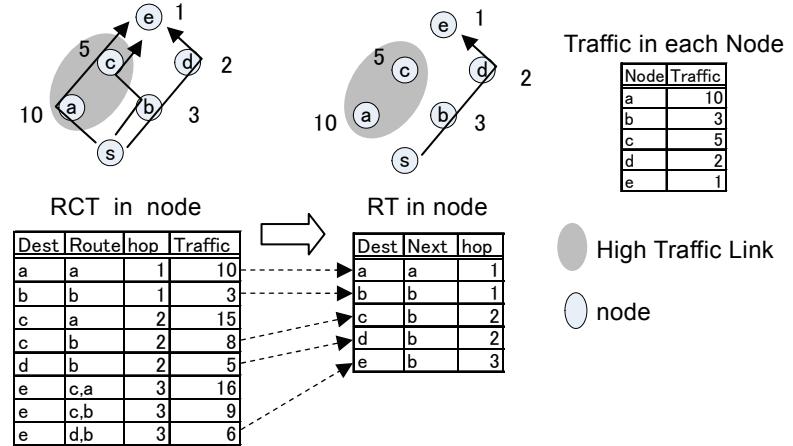


図 5 拡張 OLSR による RT 生成方法

HELLO メッセージ、TC メッセージから最短経路候補を RCT に複数生成し、経路ごとの合計トラヒック量を計算する。Route には Dest へ到達するまでの全てのノードが記述される。hop が 1 及び 2 の Dest の経路は、それぞれ隣接ノード集合、2 ホップ隣接ノード集合を参照することにより計算される。hop が 3 以上の Dest の経路は次のようにして計算する。トポロジー集合から e の隣接ノードは c と d であるため、RCT で Dest が c と d となるレコードを参照し、Route の値に c 及び d を付加することにより、e の Route が [c, a], [c, b], [d, b] と決まる。RCT が決定したら、同一宛先の中から最小トラヒックの経路を抽出する。さらに、抽出されたレコードの Route から右端のノード([c, b] の場合は b)を Next の値として RT を生成する。ノード a~d においても同様の方法で RT を生成することにより、図 5 に示すように s→b→d→e というトラヒックの高いリンクを避けた経路が完成する。

ある瞬間ににおいて、新しい通信セッションが開始されると、その時のトラヒック量が最少となる最適な経路が生成される。この通信によって経路全体のトラヒック量は変化する。この影響により、今まで使用していた経路は最適な経路ではなくなる可能性がある。このように、新たな通信がトラヒックに影響を与え、頻繁に経路が切り替わることが考えられる。この解決策として、セッションが終了するまでは同じ経路を使い続ける方法が考えられる。

## 5. むすび

MANET では集中的に通信の中継を行う負荷の高いノードが発生する。既存のアドホックルーティングプロトコルの多くは単純に最短経路を選択しているため、負荷の高い経路を選択してしまう可能性があり、スループットの低下が起きる。本論文では、既存の OLSR を拡張し、トラヒック量を経路選択の指標にすることにより、負荷の低い経路を選択できるアドホックルーティングプロトコルの実現方法を示した。今後は検討結果に基づきシミュレーションを実施し、動作検証を行う。また、電池の消耗度などトラヒック以外の要素を考慮した経路選択ができるような検討を行う。

### 参考文献

- [1] S. Corson : “Mobile Ad hoc Networking (MANET) : Routing Protocol Performance Issues and Evaluation Considerations”, RFC 2501 (1999)
- [2] T. Clausen, Ed. : “Optimized Link State Routing Protocol (OLSR)”, RFC 3626 (2003)
- [3] D. Johnson : “The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4”, RFC 4728 (2007)
- [4] C. Perkins : “Ad hoc On-Demand Distance Vector (AODV) Routing”, RFC 3561 (2003)
- [5] R. Ogier : “Topology Dissemination Based on Reverse-Path Forwarding (TBRPF)”, RFC 3684 (2004)
- [6] Royer, E.M.; Chai-Keong Toh : “A Review of Current Routing Protocols for Ad hoc Mobile Wireless Networks”, IEEE Personal Communications Pages:46 – 55 (Apr 1999)
- [7] C-K.Toh 著, 構造計画研究所 訳：“アドホックモバイルワイヤレスネットワーク”, 共立出版株式会社 (2003)
- [8] 間瀬 憲一, 阪田 史郎 : “アドホック・メッシュネットワーク”, コロナ社 (2007)
- [9] Douglas S. J. De Couto, Daniel Aguayo, Benjamin A. Chambers, Robert Morris : “Performance of multihop wireless networks: shortest path is not enough”, ACM SIGCOMM Computer Communication Review Pages:83 – 88 (Jan. 2003)
- [10] Toh, C.-K.: “Associativity-Based Routing for Ad-Hoc Mobile Networks”, Wireless Personal Communications, Vol.4 No.2 Pages:103 - 139 (1997)
- [11] 高橋 ひとみ, 斎藤 匠人, 間 博人, 戸辺 義人, 德田 英幸 : “MANET における TCP スループット推定による経路選択機構の実環境評価”, 情報処理学会論文誌 Vol. 46 No.12 Pages:2857 - 2870 (Dec. 2005)
- [12] Hipercam : <http://www.lix.polytechnique.fr/hipercom/>

# トラヒック状況を考慮したアドホック ルーティングプロトコルの検討

Researches on an Ad-hoc Routing Protocol  
considering Traffic Conditions

名城大学大学院 理工学研究科 情報工学専攻  
森崎 明 伊藤 将志 渡邊 晃

# はじめに

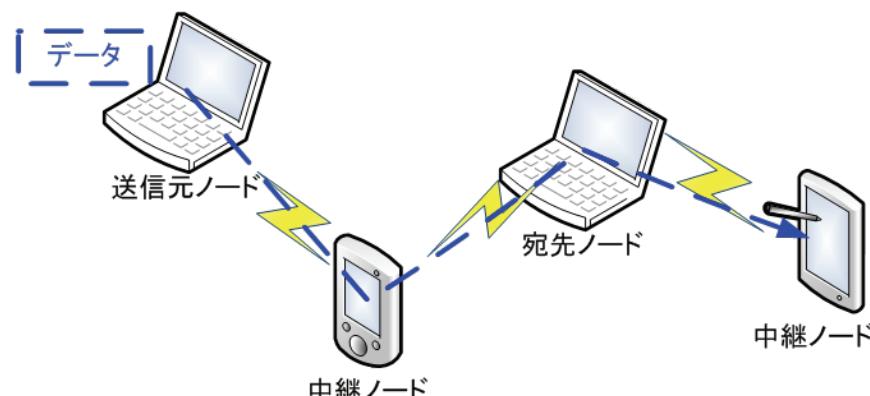
無線LANの普及に伴い, **MANET (Mobile Ad-hoc Network)** の研究が注目されている

## ■ MANET

- アクセスポイントを必要としない
- 無線通信機能を備えたノードのみで構成されるネットワーク
- すべてのノードは中継機能をもつ
- 遠隔のノードとはマルチホップ通信を行う
- 特有のルーティングプロトコルを必要とする

## ■ 利用形態

- 災害時などでインフラを利用できない場面での通信
- 会議時, イベント会場などの一時的な通信



# アドホックルーティングプロトコル

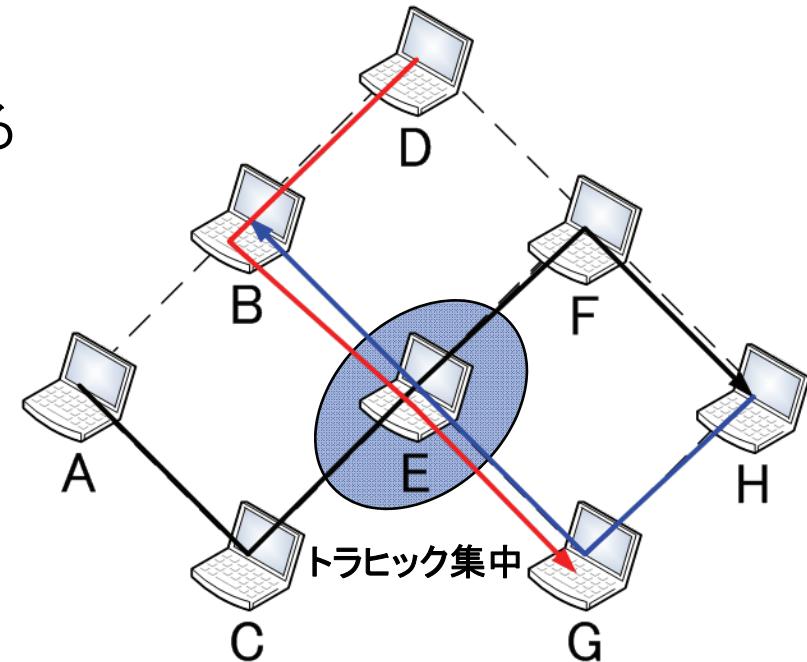
- ノードはリンク接続状態の変化へ迅速に対応
- ノードはマルチホップ通信を行うための中継機能を持つ

分類	特徴
プロアクティブ型	<ul style="list-style-type: none"><li>・一定間隔ごとに経路を生成することにより、通信を即座に開始できる</li><li>・通信の発生頻度が高く、ネットワーク構成が頻繁に変化しないネットワークに有効</li></ul> <p>例) OLSR (Optimized Link State Routing)</p>
リアクティブ型	<ul style="list-style-type: none"><li>・通信開始時にのみ経路を生成し、相手と通信できなくなるまで同じ経路を利用する</li><li>・通信の発生頻度が低く、ネットワーク構成が頻繁に変化するネットワークに有効</li></ul> <p>例) AODV (Ad hoc On-Demand Distance Vector)</p>

# アドホックルーティングプロトコルの課題

- ほとんどのアドホックルーティングプロトコルは、中継ホップ数を指標とした最短経路を選択する
- しかし、単純に選択された最短経路は実際は最善な経路とは限らない
  - トラヒックが集中する可能性がある  
パケットロスが多発

↓  
スループットが大きく低下する

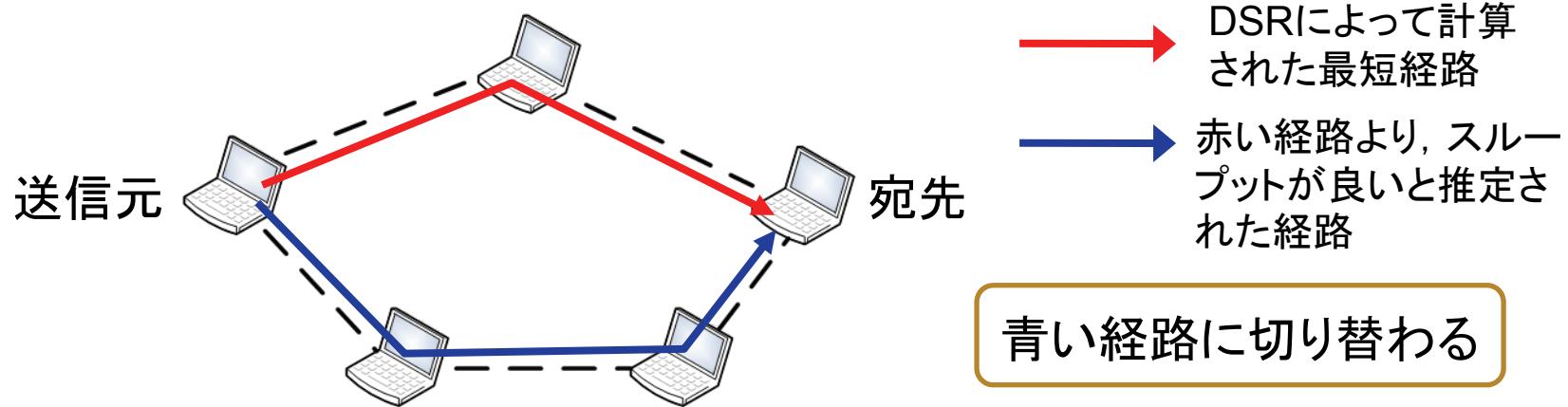


スループットの向上を目的とするアドホックルーティングプロトコルが研究されている

# 関連研究

## ■ ETR (Estimated-TCP-Throughput Maximization based Routing)

- リアクティブ型のDSR (Dynamic Source Routing)を拡張
- DSRによって計算された最短経路から、スループットが良いと推定される経路に切り替える
- スループットの算出に必要な遅延と往復パケット損失率は、データ送信開始時から一定間隔で送信されるRTPLM(Round-Trip Packet Loss ratio Measurement)要求とその応答によって収集



一定間隔でRTPLM要求の送信を行うため、ネットワークへのオーバーヘッドが高くなる

# 提案方式

## ■ 経路選択方法

- 経路上の各ノードのトラヒック情報を基に、最少トラヒックとなる最短経路を選択

## ■ 検討の対象

### 着目点

- ネットワーク全体の各ノードのトラヒック情報を新たな制御メッセージを設けることなく、収集する

### 対象の決定

- プロアクティブ型は定期的に配達される制御メッセージによってネットワーク全体のトポロジーを把握する
- 定期的に配達される制御メッセージを使ってネットワーク全体の各ノードのトラヒック情報を収集する

プロアクティブ型を検討対象とし、その中の代表的でかつ最も普及しているOLSRを提案方式の対象とする

# OLSRの概要

## ■ 各ノードは**HELLO**, **TC**メッセージの送受信によりRTを生成

### ■ **HELLO**メッセージ

- 各ノードが持つ情報を通知するために、2秒毎に隣接ノードへブロードキャスト

### ■ **TC**メッセージ

- ネットワークトポジーを通知するためにMPRにより、5秒毎にネットワーク全体にフラッディング

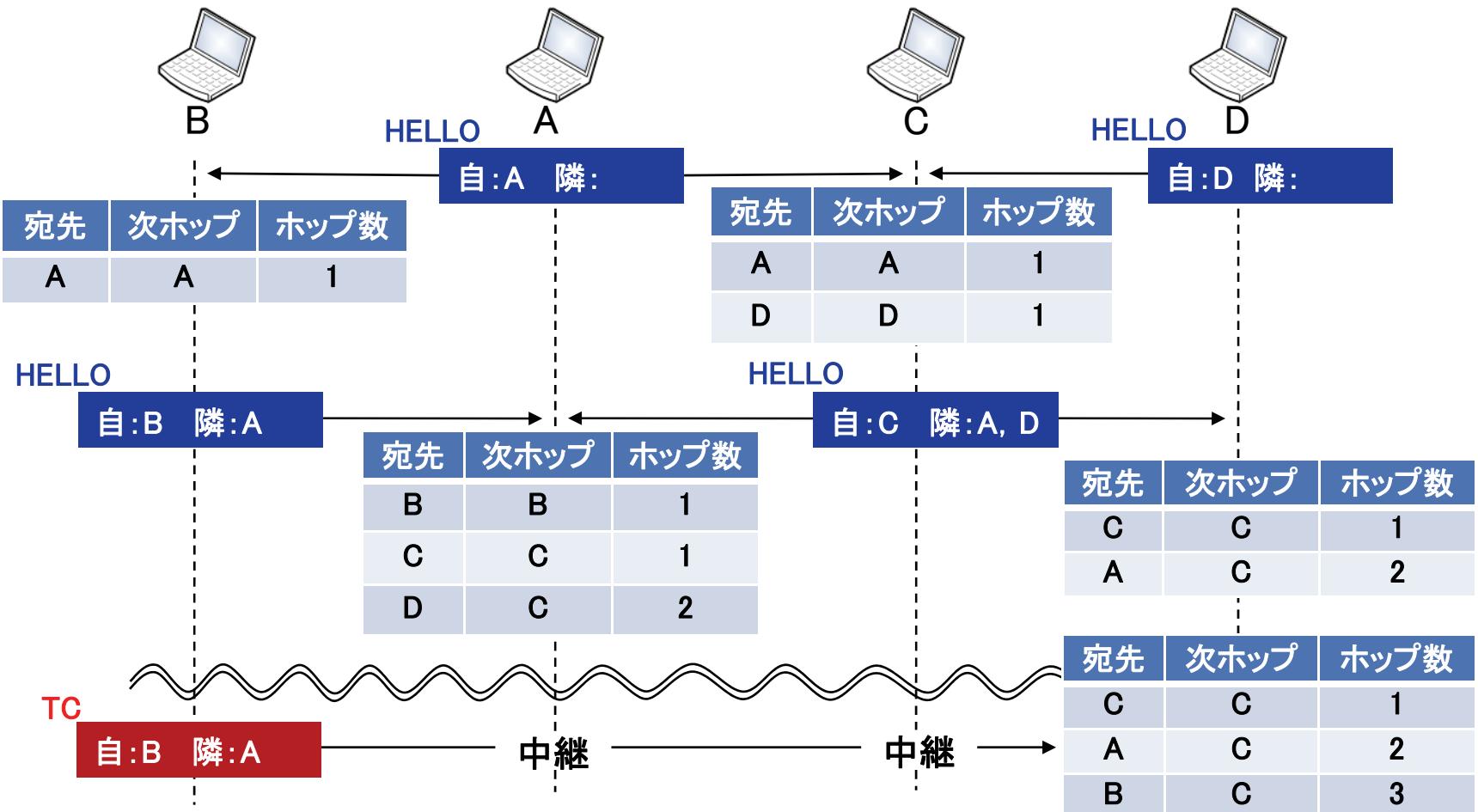
## ■ 特徴

- 規模が大きく、密度の高い環境に適する
- 必要最低限のノード「MPR」を使って効率的なフラッディングを行う

# 制御メッセージの送受信

## ■ HELLO, TCメッセージの送受信

- HELLO, TCとともに自身のアドレスと隣接ノードのアドレスから構成



# OLSRの経路選択

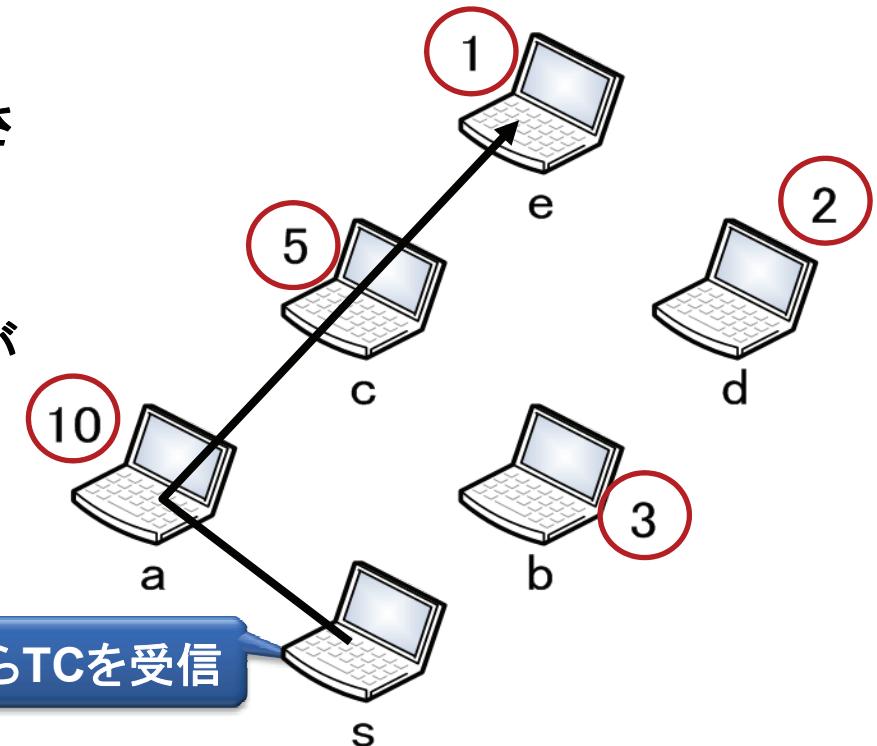
ノードsからノードeへの経路生成

- sはeからTCメッセージを受信し、eへの経路をRTに追加
- eへの経路の次ホップには既に生成されている経路のうちeの隣接ノードで、最初に見つかるcの次ホップaを選択
- 同様に各ノードでRTにeまでの経路が生成され、一つの経路が完成

ノードsのRT

宛先	次ホップ	ホップ数
a	a	1
b	b	1
c	a	2
d	b	2
e	a	3

New



この経路はトラヒック状態の悪いaを含んでおり、最善な経路ではない

# OLSRの拡張方法

- HELLOメッセージとTCメッセージにトラヒック情報を追加する
- 宛先への複数の最短経路の合計トラヒック情報を計算した新たな経路計算テーブル(RCT:Route Calculation Table)を定義
- RCTで生成された複数の最短経路の中から最もトラヒック情報の良い経路を選択し、これを基にRTを生成

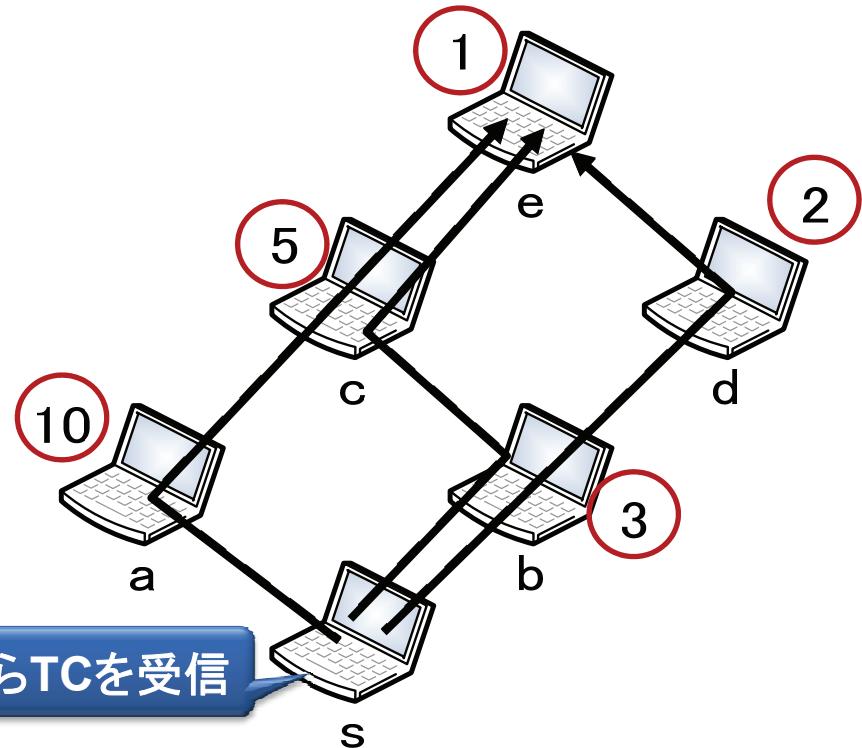
# 拡張OLSRの動作 ①

ノードsからノードeへの経路生成

- sはeからTCメッセージを受信し、RCTにeへの最短経路を複数追加
- 中継ノード欄にはeへの経路のすべての中間ノードを記述
- トラヒック欄には経路のトラヒック情報を記述

ノードsのRCT

宛先	中継ノード	ホップ数	トラヒック
a	a	1	10
b	b	1	3
c	a	2	15
c	b	2	8
d	b	2	5
e	c, a	3	16
e	c, b	3	9
e	d, b	3	6



New

同様に各ノードでRCTを生成し、複数の最短経路候補が完成

# 拡張OLSRの動作 ②

ノードsからノードeへの経路生成

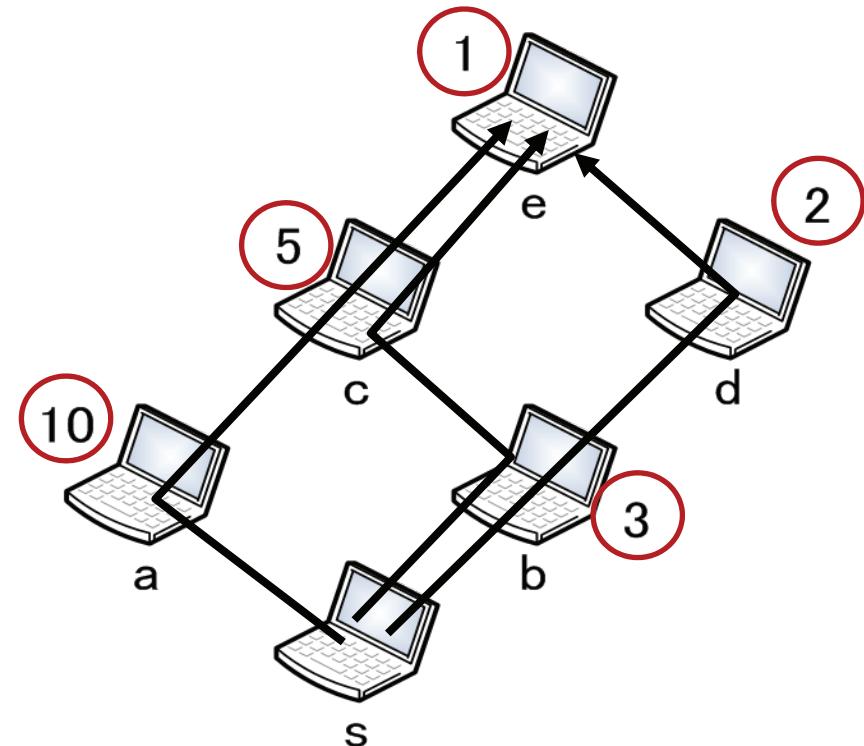
- RCTの中からトラヒック情報の良い経路を選択してRTを生成

ノードsのRCT

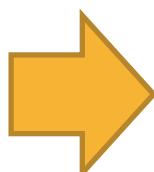
宛先	中継ノード	ホップ数	トラヒック
a	a	1	10
b	b	1	3
c	a	2	15
c	b	2	8
d	b	2	5
e	c, a	3	16
e	c, b	3	9
e	d, b	3	6

ノードsのRT

宛先	次ホップ	ホップ数
a	a	1
b	b	1
c	b	2
d	b	2
e	b	3



同様に各ノードでRTを生成し、トラヒック情報の良い最適な経路が完成



# 拡張OLSRの実現方法 ①

## ■ 情報リポジトリ

- **HELLO**, **TC**メッセージを送受信することにより構築

### ■ リンク集合

- 直接電波の届くノードの集合

### ■ 隣接ノード集合

- 隣接ノードのアドレスやそのノードの再送信の積極度から成る集合

### ■ 2ホップ隣接ノード集合

- 隣接ノード集合のさらに1ホップ先に存在するノードの集合

### ■ MPR集合

- MPRとして選択された隣接ノードの集合

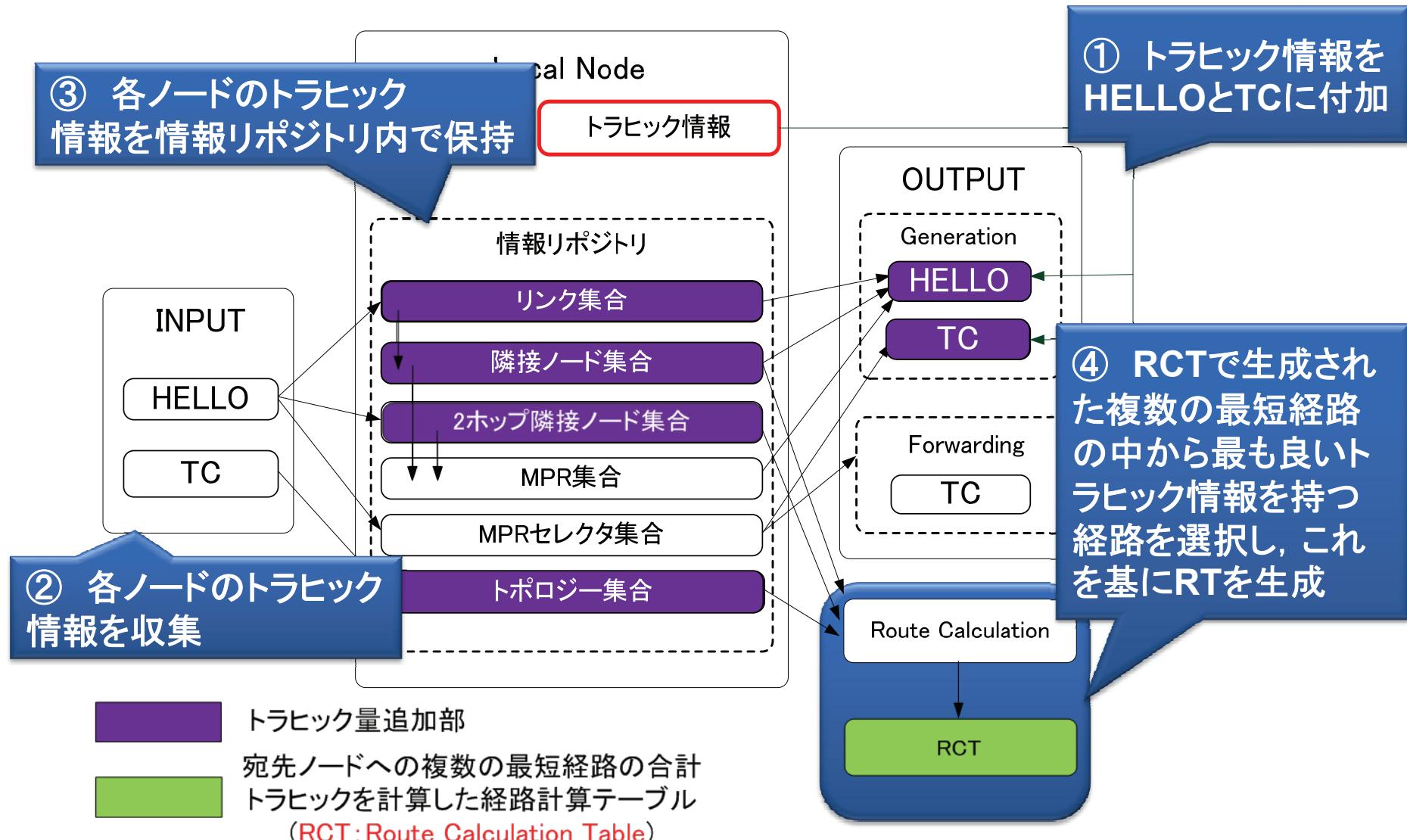
### ■ MPRセレクタ集合

- 自身をMPRとして選択しているノードの集合

### ■ トポロジー集合

- 3ホップ以上のノードとその隣接ノードを含むネットワークトポロジーの集合

# 拡張OLSRの実現方法 ②



# まとめ

## ■ 発表内容

- 集中的に通信の中継を行うノードが発生する環境において、スループットの向上を目的とする
- OLSRを拡張することにより、トラヒック状態を考慮した経路選択が可能なアドホックルーティングプロトコルを検討し、その実現方法を示した

## ■ 今後の予定

- 検討結果に基づきネットワークシミュレータns-2を用いてシミュレーションを実施し、動作検証を行う
- トラヒック状態以外の経路選択指標を検討する（例：電池の消耗度）

■ 補足

# 拡張OLSRの実現方法 ③

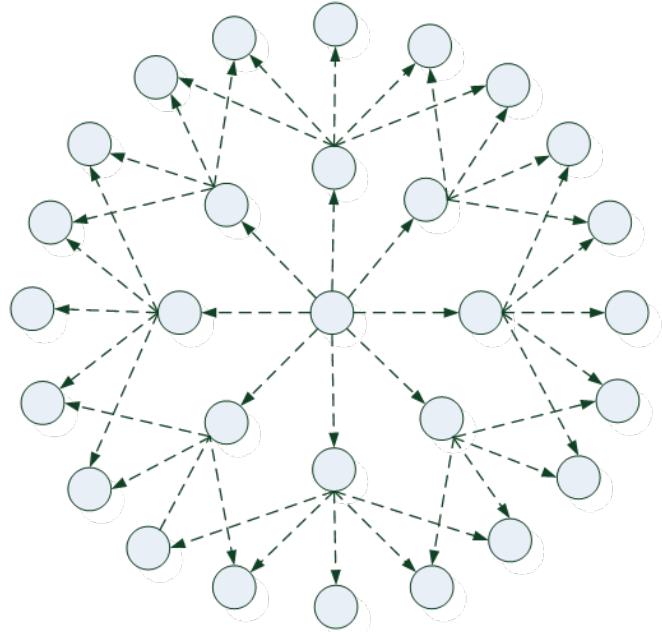
## トラヒック情報の取得

- MAC層で1秒毎に送受信のトラヒック量を求める

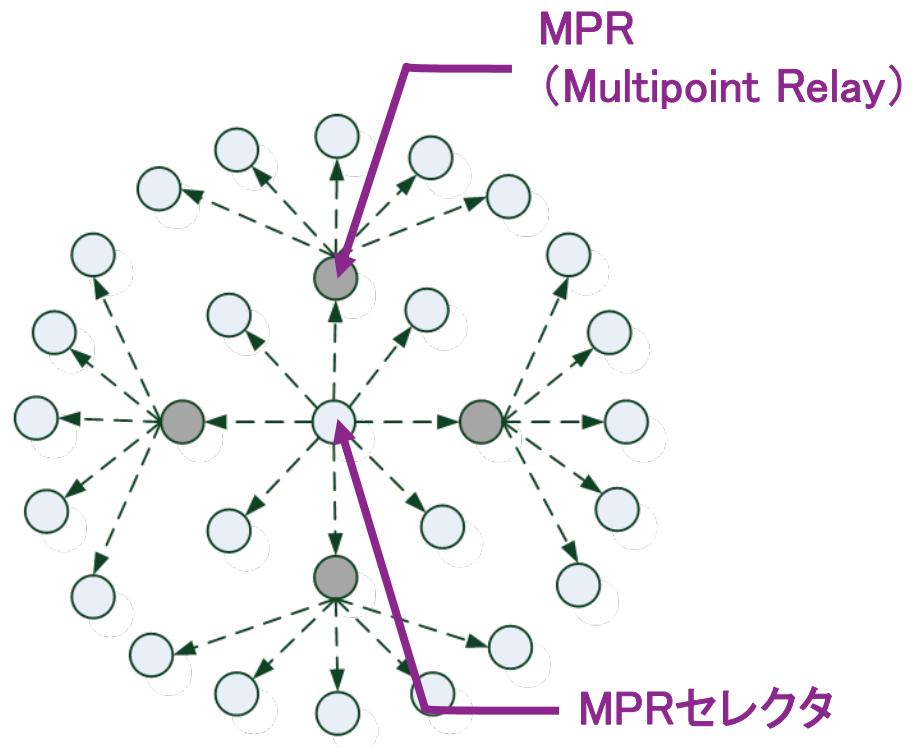
トラヒック量 = 制御信号のパケットサイズ  
+ データパケットのサイズ

- ・制御信号のパケット:ACK, RTS, CTS
- ・ データパケット : データ, HELLO, TC

# OLSRのフラッディング



通常のフラッディング



OLSRのフラッディング

# OLSRのパケットフォーマット

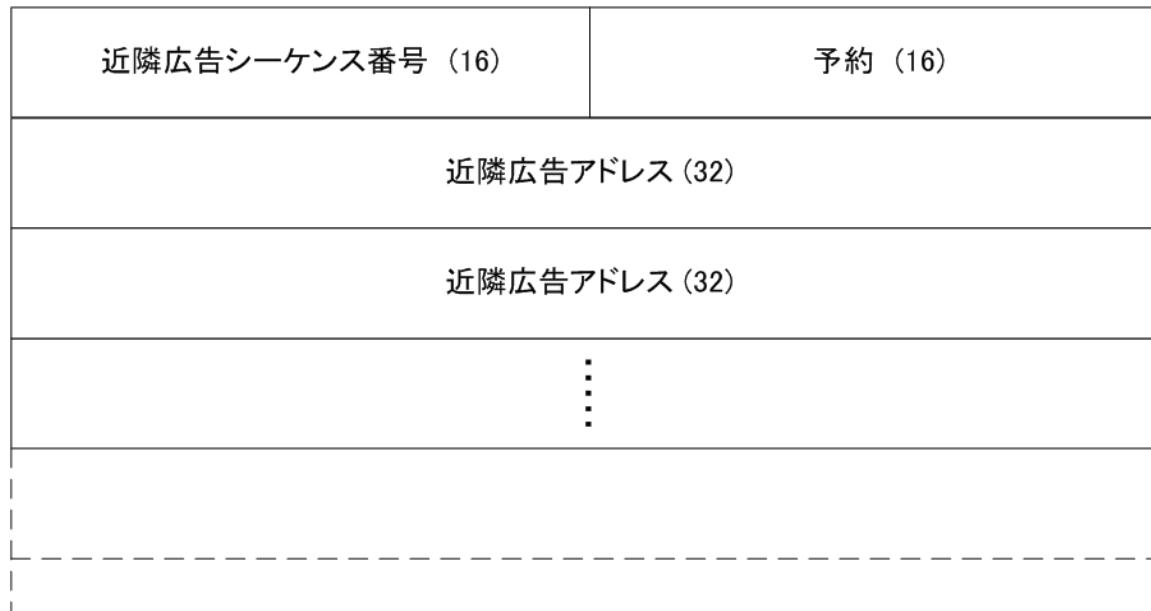
パケット長 (16)		パケットシーケンス番号 (16)
メッセージタイプ (8)	有効時間 (8)	メッセージサイズ (16)
発信元アドレス (32)		
TTL (8)	ホップ数 (8)	メッセージシーケンス番号 (16)
メッセージ		
メッセージタイプ (8)	有効時間 (8)	メッセージサイズ (16)
発信元アドレス (32)		
TTL (8)	ホップ数 (8)	メッセージシーケンス番号 (16)
メッセージ		

IPやUDPヘッダを取り除いた形で、OLSRのパケットは、「パケットヘッダ」と複数の「メッセージヘッダ」から成り立っている

# HELLOメッセージフォーマット

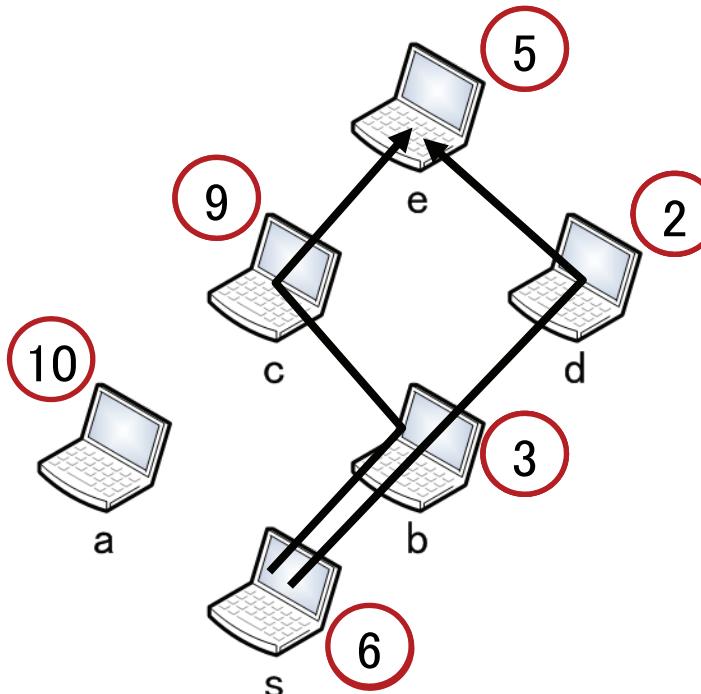
予約 (16)	HELLO発生間隔 (8)	Willingness (8)
リンクコード (8)	予約 (8)	リンクメッセージサイズ (16)
隣接ノードのインターフェースアドレス (32)		
隣接ノードのインターフェースアドレス (32)		
⋮		
リンクコード (8)	予約 (8)	リンクメッセージサイズ (16)
隣接ノードのインターフェースアドレス (32)		
隣接ノードのインターフェースアドレス (32)		

# TCメッセージフォーマット



# 考えられる問題点

- 通信によって経路のトラヒックが増え、更新時に経路が頻繁に切り替わってしまう



## 対応策

- 一度経路が決まったら、セッションが終わるまで同じ経路を使用する

# OLSRのI/O図

