

# 通信状態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> 渡辺晃<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）の研究が期待されている。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 TCP や UDP のトラヒック状況が考慮されていない。そのため、最短経路が複数存在する場合にはどの経路を選ぶかは実装に依存したものとなっている。本論文では OLSR（Optimized Link State Routing）を拡張することにより、TCP, UDP それぞれの特性を活か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提案する。

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

AKIRA MORISAKI<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 expected. Most of ad-hoc routing protocols have not considered about TCP and UDP traffic conditions in the network. Therefore, when there are some shortest routes, it is depended on implementation which route is chosen. We propose an ad-hoc routing protocol can make use of TCP and UDP characteristic by extending OLSR (Optimized Link State Routing).

### 1. はじめに

無線 LAN は、配線が不要で端末が自由に移動できるなどの利便性からネットワークへの接続方法として需要が高まってきている。最近では公共の場だけでなく、家庭内でも無線でネットワークを組む例が増加している。

無線 LAN を構築する方法には、端末が必ず AP (Access Point) を介して通信を行うインフラストラクチャモードによる方法と、端末同士で直接通信を行うアドホックモードによる方法がある。後者は、災害時や会議時、イベント会場などで一時的な無線ネットワークを構築できるモバイルアドホックネットワーク（MANET：Mobile Ad-hoc Network）[1]に応用されている。MANET は、あらゆる無線端末が中継端末となり得るため、その場でネットワークを構築することができるという特徴がある。近年では、インフラストラクチャモードの AP 間を MANET の技術で結合する無線メッシュネットワークの研究にも注目が集まっている[2]-[6]。

MANET を構築するには、各端末がアドホックルーティングプロトコルを用いてルーティングテーブルを生成する必要がある。アドホックルーティングプロトコルは、IETF (Internet Engineering Task Force) により、今まで多くの方式が提案されているが[7]-[13]、経路生成の際に中継ホップ数が最短となる経路（最短経路）を探索することが目的となっており、最短経路が複数存在する場合にどの経路を選ぶかは実装に任せられている。そのため、トラヒックが集中した中継ノードが発生すると、パケットロスが多発し、結果的にスループットが低下してしまうという課題がある[14]。

複数経路の中から適切な経路を選択することを目的としたアドホックルーティングプロトコルの研究として以下のものが挙げられる。ABR (Associativity-Based Long-lived Routing) [15]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出ないため、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) [16]は DSR (Dynamic Source Routing Protocol) [8]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 TCP 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。TCP スループットは所定の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、この方式は TCP スループットだけに着目されており、UDP スループットは考慮されていない。また、新たな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。

本稿では、MANET のアドホックルーティングプロトコルの一種である OLSR を拡張することによって、経路上の通信状態を考慮したプロトコル PD-OLSR (Protocol Dependent-OLSR) を提案する。具体的には、経路情報を収集する制御メッセージに、TCP と UDP のトラヒック状況を別々に追加し、TCP, UDP 用のルーティングテー

<sup>†</sup>名城大学大学院理工学研究科  
Graduate School of Science and Technology, Meijo University

ブルを別々に生成する。これにより、TCPとUDPの特性を活かしたルーティングを実現する。

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PD-OLSRの経路生成方法を説明する。5章ではPD-OLSRの実装方法を示し、最後に6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移動可能ノードと通信するため、各ノードは中継機能を持ち、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存在する。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トコルは存在しない。

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種類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[17]-[18]。

### 2.1 Proactive型

Proactive型のルーティングプロトコルでは、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、ネットワークトポロジー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ことで、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。

### 2.2 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は、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に遅延が発生してしまうが、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。

表1 MANETのルーティングプロトコルの分類

分類	方式・特徴	特徴	例
Proactive型	通信要求が発生する前からルーティングテーブルを生成	通信頻度の高いネットワークに適する	OLSR DSDV TBRPF
Reactive型	通信要求が発生した際にネットワーク内で経路探索プロセスが始動	ノードの移動が頻繁なネットワークに適する	AODV DSR TORA ABR
Hybrid型	ネットワーク内を複数のゾーンに分割し、ゾーンの内外でProactive型とReactive型を使い分けて経路を構築	Proactive型とReactive型の両方の特徴を活かすことができる	ZRP

### 2.3 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活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。

## 3. OLSR

プロアクティブ型のプロトコルは、ルーティングテーブルを定期的に更新するために送受信される制御メッセージを改造することにより、シンプルに経路上の通信状況を計算することができる。そこで、プロアクティブ型の代表的かつ最も普及しているOLSR(Optimized Link State Routing)[7]を提案方式のベースとする。以下にOLSRの原理と経路生成方法について説明する。

### 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(デフォルト送信間隔2秒)に隣接ノードにブロードキャストする。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。

### 3.2 OLSR のフラッディング方式

OLSR の最大の特徴は、効率の良いフラッディングを実現できることである。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動作である。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。この方法では、ノード数が多くなるとブロードキャストによるパケット数が急激に増大し、トラヒックを圧迫する。

OLSR では必要最低限の中継ノード（MPR : Multipoint Relay）を選択し、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。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、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 メッセージが使用されるような環境は想定しないため、これらの説明は省略する。

### 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、リンクが双方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、リンクが双方向か非双方向であるかの状態、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方向の 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生成する。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 と略す）を生成する。

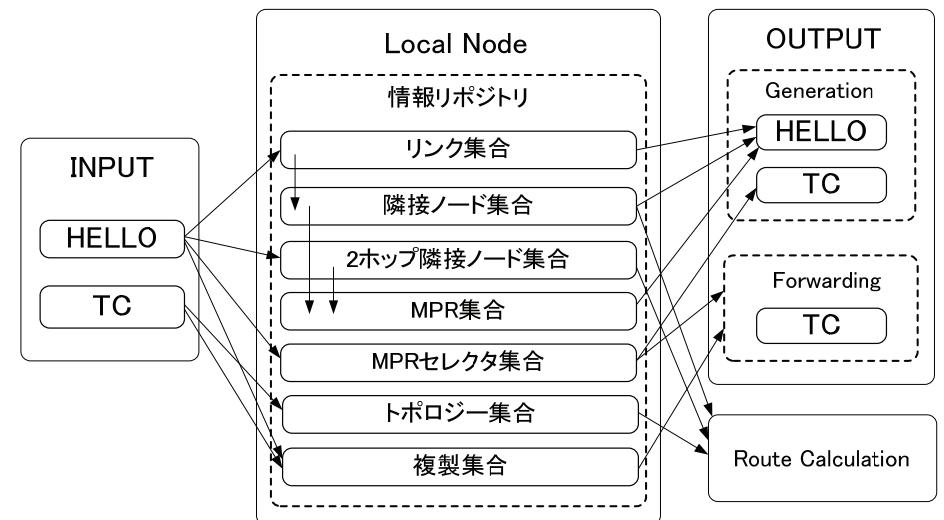


図 2 制御メッセージと情報リポジトリの関係

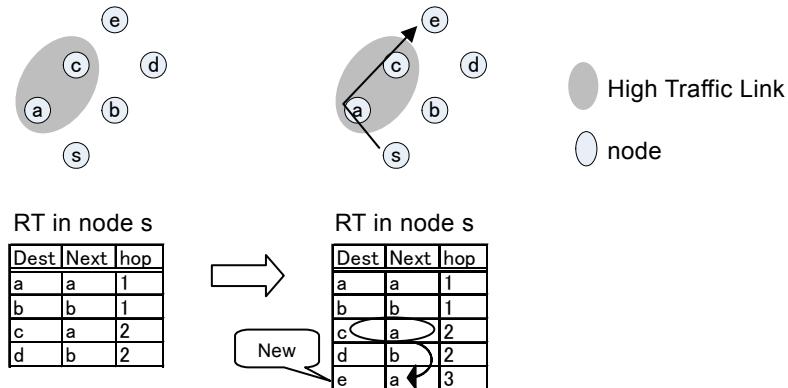


図 3 OLSR による RT 生成方法

### 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 の経路生成の手順を示す。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。

このように OLSR では、単純に最初に発見された最短経路が選ばれる。すなわち、選択される経路は実装に依存したものとなる。この経路が図 3 に示す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、パケットロスが多発しスループットが低下する。

## 4. PD - OLSR の提案

本論文では OLSR を基にして、通信状態を考慮した経路選択を行う PD - OLSR を提案する。TCP/IP では UDP と TCP という特性が全く異なる 2 種類のトラヒックが存在する。UDP は、端末側が意図した流量のトラヒックがそのままネットワークへ送出される。ネットワーク内でパケットロスが発生してもそれは変わらない。これに対して、

TCP は輻輳制御の機能により、順調に ACK が返ってくれば、ウインドウサイズを大きくし、帯域を有効に使おうとする。パケットロスが発生すると、ネットワークに輻輳が発生したものと判断し、ウインドウサイズを小さくする。このようにしてウインドウサイズが適切な大きさに調整され、ネットワークがさらに輻輳することを防止する。このような特性の違いから、ネットワーク上のトラヒックは、以下のようになる。まず送信された UDP パケットの合計より UDP が占めるトラヒック量が定まり、残りの余裕がある帯域分を複数の TCP セッションが分け合う。UDP のパケットロスはそのまま消滅するが、TCP の場合は再送処理を行いながら、スループットが最大になるように輻輳制御が働く。TCP の効率は、TCP の輻輳制御がうまく機能するかどうかによって決まる[20]-[22]。

以上の考えに基づき、UDP 通信においては、単純に現在の UDP トラヒックの少ない経路を選ぶ。一方、TCP 通信においては、空いている帯域を現在の TCP セッション数で割り、その値が大きい方が帯域に余裕があるとみなし、そちらの経路を選ぶ。PD-OLSR では既存の OLSR に対して、以下のよう拡張を行う。HELLO メッセージにその送信元ノードの UDP トラヒック量と TCP セッション数を追加する。各ノードは HELLO メッセージの送受信によって、隣接ノードが有する UDP トラヒック量と TCP セッション数を収集する。次に宛先ノードへの複数の最短経路と次ホップノードの通信状況情報 (UDP トラヒック量、TCP セッション数) を有する経路計算テーブル (RCT : Route Calculation Table) を新たに定義する。RCT で生成された複数の最短経路の中から、UDP 通信用と TCP 通信用に最も適した経路を別々に選択する。経路を選択する方法は以下の通りである。UDP 通信用の経路は、次ホップノードの最小 UDP トラヒック量である経路を選択する。TCP 通信用の経路は、次ホップノードにおいて、新しく開始される TCP セッションに割り当て可能な帯域 NSB (New Session Bandwidth) を以下の式で求め、NSB が最大となる経路を選択する。

$$NSB = \frac{X - UDPTraffic}{TCPsession + 1} \dots \dots \dots (1)$$

X はトラヒックの使用可能帯域、UDPTraffic は次ホップノードの UDP トラヒック量、TCPsession は次ホップノードの TCP セッション数を表す。以上的方法により、UDP, TCP の特性を活かした経路選択ができ、通信状態の悪い経路を避けた最短経路を選択することができる。

図 4 に PD-OLSR の経路生成手順を示す。図 4 は、図 3 と同様にノード 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 UDP トラヒック量と TCP セッション数を表す。例えば、UDP : 5 とは UDP トラヒックが 5Mbps の帯域を占め、TCP : 1 とは TCP セッション数が 1 つ存在していることを示す。

HELLO メッセージ、TC メッセージを受信すると、RCT に最短経路候補を複数生成する。Route には Dest へ到達までの全てのノードが記述される。hop が 1 及び 2 の Dest への経路は、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できる。hop が 3 以上の Dest へ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 の UDPTraffic には Next の UDP トラヒック量が記述される。また、RCT の TCPSession には Next が確立している TCP セッション数が記述される。RCT が決定すると、最小 UDPTraffic となる経路を UDP 通信用の RT へ抽出する。更に、(1) 式により NSB を求め、最大 NSB となる経路を TCP 通信用の RT へ抽出する。例えば、使用可能帯域 X=54Mbps の場合において、Dest が e の経路 [c, a], [c, b], [d, b] の NSB は、それぞれ  $(54-10)/2 \Rightarrow 22$ ,  $(54-3)/4 \Rightarrow 12.8$ ,  $(54-3)/4 \Rightarrow 12.8$  となる。この結果、RT へ抽出される経路は最大 TCP スループットが期待できる [c, a] が抽出され、Next が a と決まる。ノード a~d においても同様の方法で UDP 通信用、TCP 通信用の RT を別々に生成することにより、図 4 に示すような UDP 通信用経路 s→b→d→e (青線) という高いトラヒックを避けた経路が完成する。また、TCP 通信用経路 s→a →c→e (赤線) という TCP 輻輳制御がうまく機能することが期待できる経路が完成する。

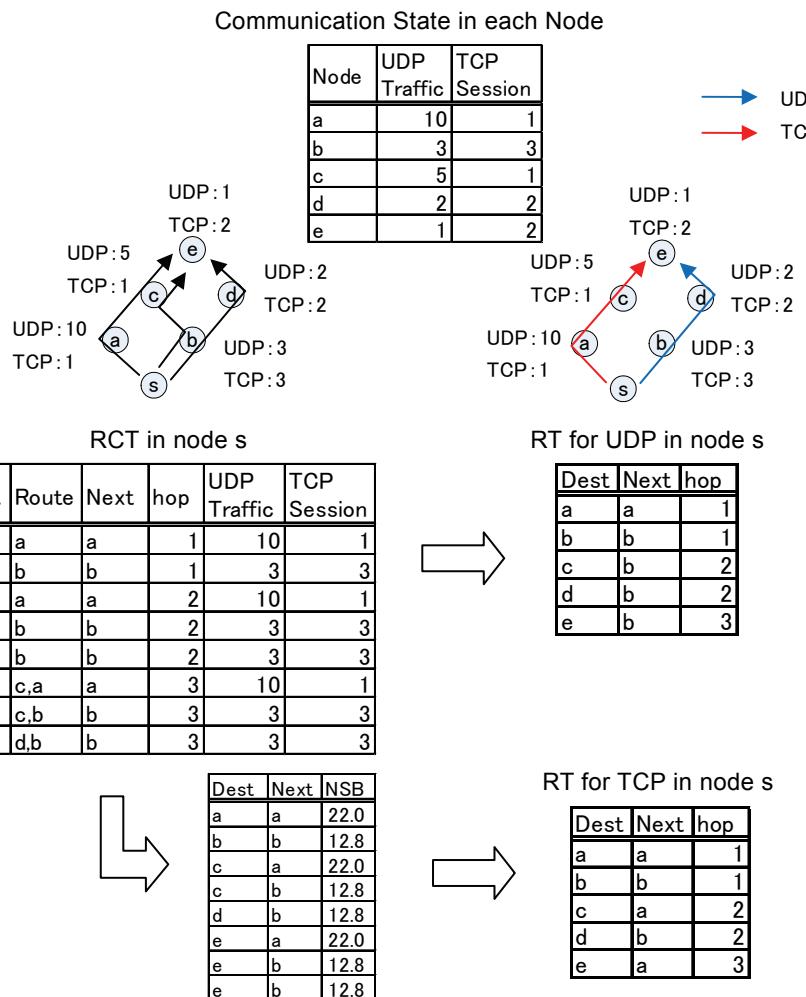


図 4 PD-OLSR による RT 生成方法

## 5. 実装方法

図 5 に PD-OLSR の実装方法を示す。図 2 の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）にトラヒック量とセッション数の項目を追加する。また、宛先ノードへの複数の最短経路において、次ホップノードが有する UDP トラヒック量と TCP セッション数を計算した RCT 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 への次ホップ（Next），Next の経路の UDP トラヒック量（UDPTraffic），TCP セッション数（TCPSession）から構成される。

PD-OLSR における制御メッセージの処理の流れは以下の通りである。

### ① 制御メッセージの送信

HELLO メッセージの送信元ノードは自身の UDP トラヒック量と TCP セッション数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 UDP トラヒック量と TCP セッション数を記録する。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。

### 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 UDP トラヒック量と TCP セッション数を記録する。

### ④ 経路計算

経路計算（RT 更新）プロセスに先立ち、RCT を生成する。RCT は 4 章で述べた方法で作成されていく。RCT が完成すると、その中から UDP 通信用 RT と TCP 通信用の経路 RT を生成する。

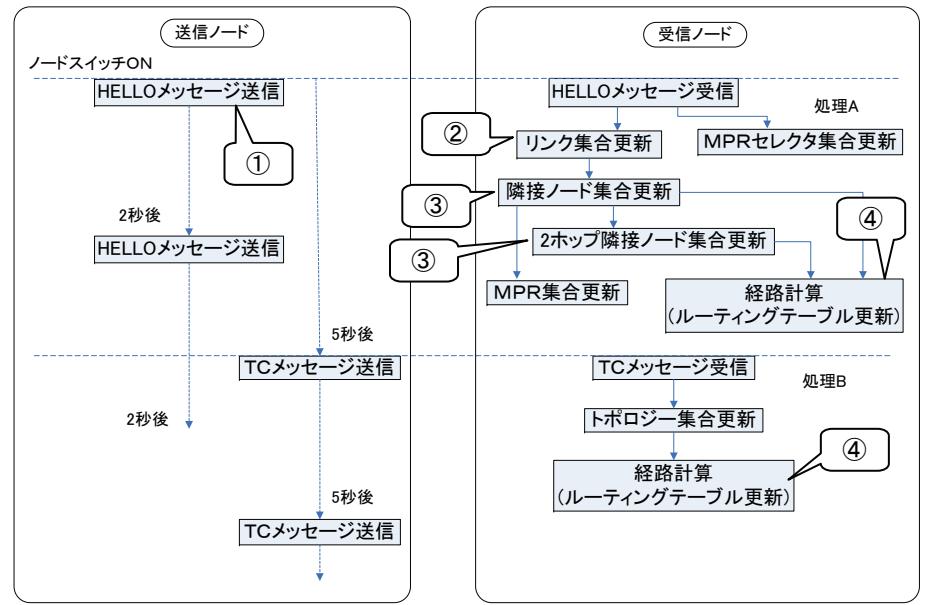


図 5 PD-OLSR の実装方法

## 6. まとめ

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い、スループットの低下が起きる可能性がある。本論文では、既存の OLSR を拡張し、UDP トラヒックと TCP セッション数を経路選択の指標にすることにより、UDP と TCP 用に別々のルーティングテーブルを生成する PD-OLSR を提案した。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。

## 参考文献

- [1] S. Corson : “Mobile Ad hoc Networking (MANET) : Routing Protocol Performance Issues and Evaluation Considerations”, RFC 2501 (1999)
- [2] MetroMesh : <http://www.tropos.com/>
- [3] MeshCruzer : <http://www.thinktube.com/>
- [4] Packethop : <http://www.packethop.com/>

- [5] Y.Amir, C.Danilov, M.Hilsdale : "Fast : Handoff for Seamless Wireless Mesh Networks", ACM MobiSys (2006)
- [6] V.Navda, A.Kashyap, S.R.Das : "De-sign and evaluation of iMesh: an infrastructure-mode wireless mesh network,World of Wireless Mobile and Multimedia Networks", pp.164–170 (2005)
- [7] T. Clausen, Ed. : "Optimized Link State Routing Protocol (OLSR)", RFC 3626 (2003)
- [8] D. Johnson : "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4", RFC 4728 (2007)
- [9] C. Perkins: "Ad hoc On-Demand Distance Vector (AODV) Routing", RFC 3561 (2003)
- [10] R. Ogier : "Topology Dissemination Based on Reverse-Path Forwarding (TBRPF)", RFC 3684 (2004)
- [11] Zygmunt J. Haas, Marc R. Pearlman, Prince Samar : "The Zone Routing Protocol (ZRP) for Ad Hoc Networks", IETF MANET Working Group Internet Draft, draft-ietf-manet-zone-zrp-04.txt (expire January 2003)
- [12] Charles E. Perkins, Pravin Bhagwat : "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", ACM SIGCOMM, vol.24, no.4 (Oct. 1994)
- [13] V.Park, S.Corson : "Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification", IETF MANET Working Group Internet Draft, draft-ietf-manet-TORA-spec-04.txt (Jul. 2001)
- [14] 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)
- [15] Toh, C.-K. : "Associativity-Based Routing for Ad-Hoc Mobile Networks", Wireless Personal Communications, Vol.4 No.2 Pages:103 - 139 (1997)
- [16] 高橋 ひとみ, 斎藤 匡人, 間 博人, 戸辺 義人, 徳田 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)
- [17] 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)
- [18] 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)
- [19] 間瀬 壽一, 阪田 史郎 : "アドホック・メッシュネットワーク", コロナ社 (2007)
- [20] 伊藤 将志, 鹿間 敏弘, 渡邊 晃 : "無線メッシュネットワークにおけるゲートウェイ分散方式の提案と評価", DICOMO2008 Vol.2008, No.1, pp.1873-1879, (Jul. 2008)
- [21] Masashi Ito, Toshihiro Shikama, Akira Watanabe : "A Proposal of Gateway Decentralization Method in Wireless Mesh Networks and Its Evaluation", ISITA2008 (Dec. 2008)
- [22] Masashi Ito, Toshihiro Shikama, Akira Watanabe : "Proposal and Evaluation of Multiple Gateways Distribution Method for Wireless Mesh Network", ICUIMC (Jan. 2009)

# 通信状態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**

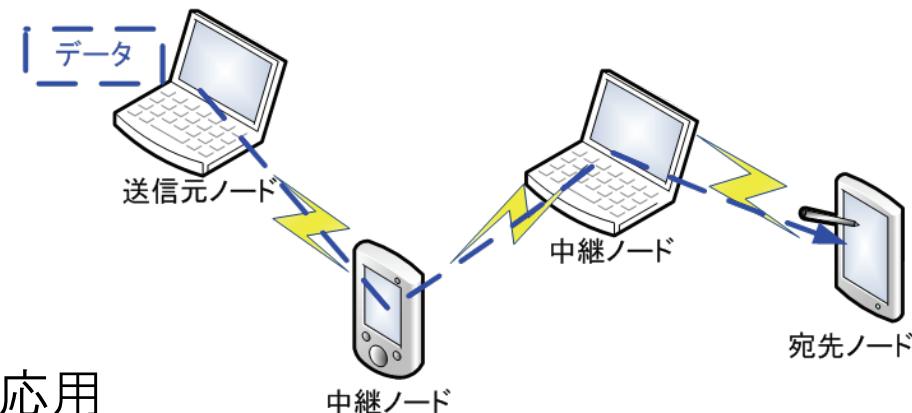
名城大学大学院 理工学研究科  
森崎 明 渡邊 晃



# 研究背景

# 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
  - 特有のルーティングプロトコルによって
    - 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
- 利用形態
  - インフラを利用できない災害時での通信
  - 会議時、イベント会場などの一時的な通信
  - 無線メッシュネットワークへの応用



# アドホックルーティングプロトコル

分類	特徴
プロアクティブ型	<ul style="list-style-type: none"><li>通信要求が発生する前からルーティングテーブルを生成</li><li>通信頻度の高いネットワークに適する</li></ul> <p>例) OLSR (Optimized Link State Routing)</p>
リアクティブ型	<ul style="list-style-type: none"><li>通信要求が発生した際にネットワーク内で経路探索プロセスが始動</li><li>ノードの移動が頻繁なネットワークに適する</li></ul> <p>例) AODV (Ad hoc On-Demand Distance Vector)</p>

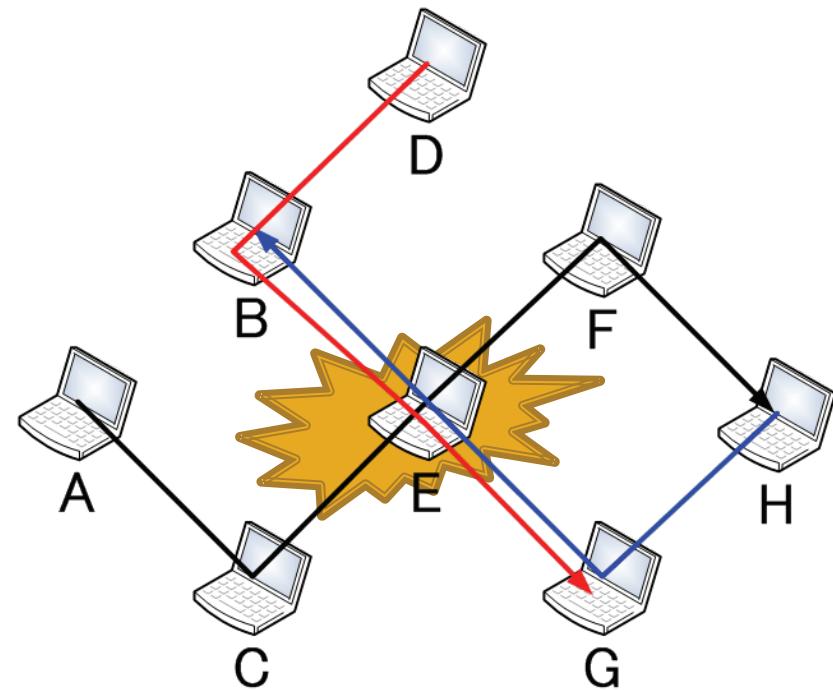
# アドホックルーティングプロトコルの課題

- ほとんどのアドホックルーティングプロトコルは、中継ホップ数が最小となる最短経路を選択する
- 最短経路が複数存在する場合はどの経路を選択するかは実装に任されている
  - トラヒックが集中する(負荷が高い)ノードを通る経路の場合  
パケットロスが多発



スループットが低下

**単純に選択される最短経路は  
実際は最善な経路とは限らない**



# OLSRの概要

- 各ノードは`HELLO`, `TC`メッセージの送受信によりネットワークに存在するノードを知りRT (Routing Table) 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 (Topology Control)` メッセージ
    - ネットワークトポロジーを通知するためにMPR (Multipoint Relay) により, 5秒毎にネットワーク全体にフラッディング  
MPR : フラッディングを効率良く行なうための必要最低限のノード

# OLSRの経路生成

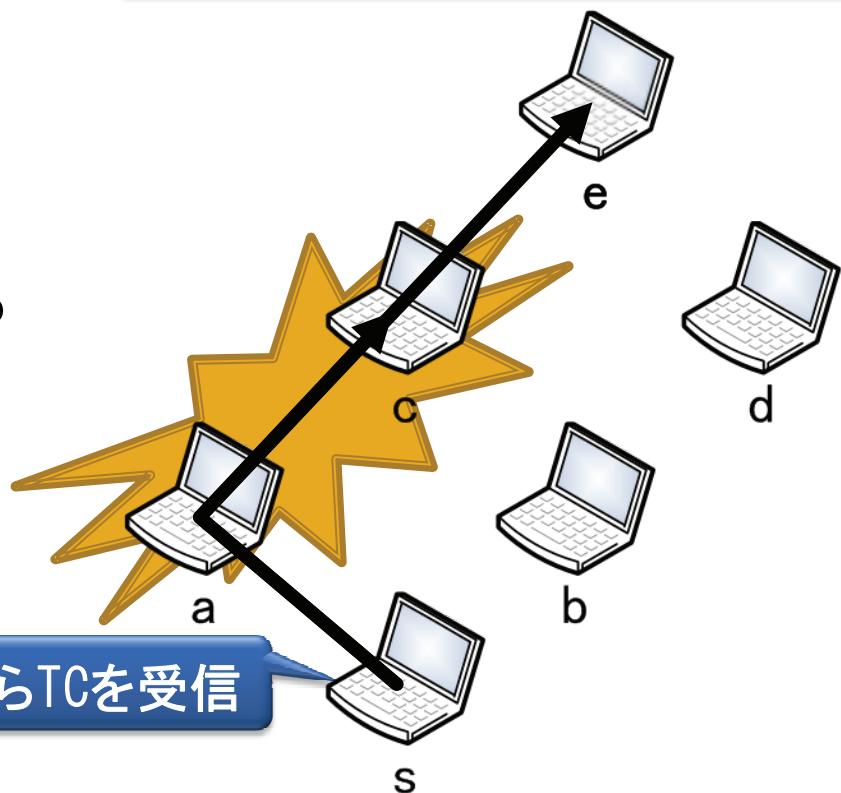
ノードsからノードeへの経路生成

- ① sはeからTCを受信し、宛先をeとする経路をRTに追加
- ② 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
- ③ 同様にしてノードa～dでRT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



aやcが負荷の高いノードの場合、この経路は最善な経路ではない

# 提案方式

# 提案方式

## ■ 目的

- 負荷の高いノードを避け、スループットが向上するルーティング

## ■ PD-OLSR (Protocol Dependent-OLSR)

- OLSRがベース
- TCPとUDPの混在環境において、特性が全く異なる2種類のトラヒックに対して別々の経路を生成
  - 負荷の高いノードを避ける
  - TCP・UDPの特性に適した経路で通信

# TCP・UDPの特性

## ■ UDP

- 端末側が意図した流量のトラヒックがそのままネットワークへ送出

## ■ TCP

- 輻輳制御によって順調にACKが返ってこればウィンドウサイズを拡大し帯域を使い切ろうとする

## ■ つまり

- TCPとUDP通信が混在するネットワークのトラヒックは

送出されるUDPパケットの合計からUDPが占めるトラヒック量が定まり、残りの余裕のある帯域分を複数のTCPセッションが分け合う

# PD-OLSR

- 各ノードはある時間の以下の情報を計算し、 HELLO, TCでネットワーク全体に広告
  - TCPセッション数
  - UDPトラヒック
- UDP信用経路の指標
  - 広告されたUDPトラヒックが小さいノードを次ホップノードに選択
- TCP信用経路の指標
  - 各ノードにおいて、新しいTCPセッションに割り当て可能な帯域  
(NSB : New Session Bandwidth)を求める
  - NSB値が大きいほど、帯域に余裕があるとみし、そのノードを経路の次ホップノードとして選択

$$NSB = \frac{X - UDP\text{Traffic}}{TCP\text{Session} + 1}$$

X : トラヒックの使用可能帯域

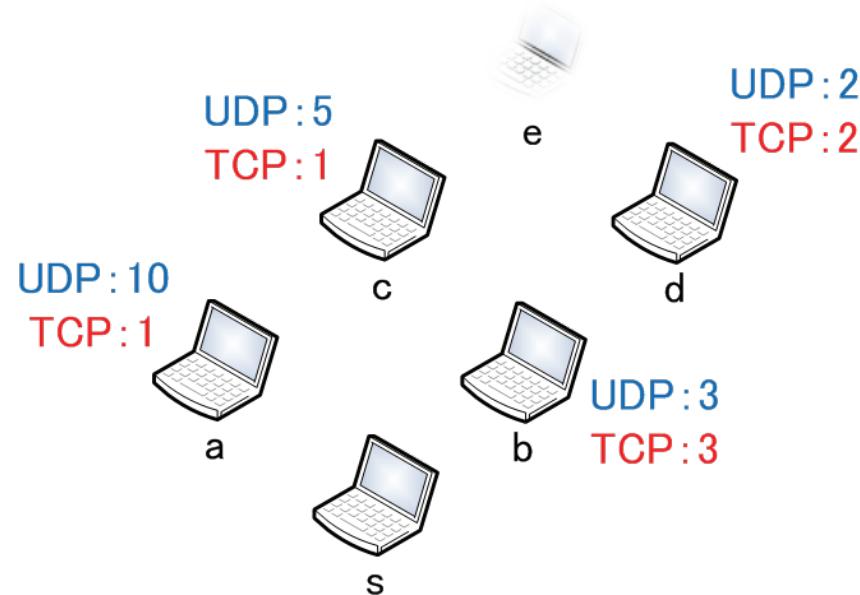
TCP Session : あるノードでセッション確立  
しているセッション数

UDPTraffic : あるノードのUDPトラヒック

# PD-OLSRの経路生成 ①

ノードsからノードeへの経路生成

- 各ノードは自身で計算したTCP Session と UDP TrafficをHELLOとTCによって、全ノードへ告知



# PD-OLSRの経路生成 ②

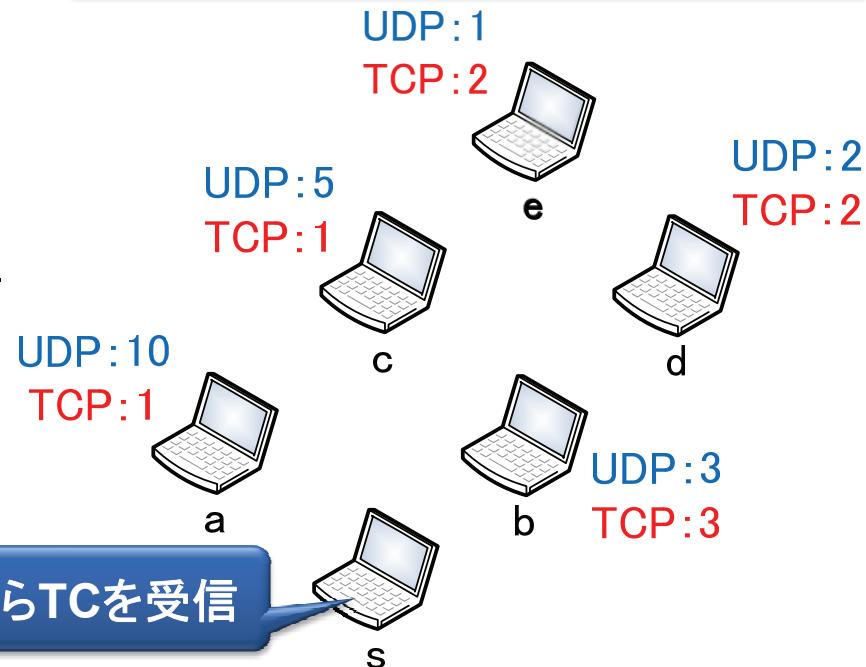
ノードsからノードeへの経路生成

- sがHELLO, TCを受信すると、TCP Session と UDP Traffic の情報を持続する

RCT (Route Calculation Table)  
を生成

ノードsのRCT

宛先	中継ノード	ホップ数	UDP Traffic	TCP Session
a	a	1	10	1
b	b	1	3	3
c	a	2	10	1
c	b	2	3	3
d	b	2	3	3
e	c, a	3	10	1
e	c, b	3	3	3
e	d, b	3	3	3



eからTCを受信

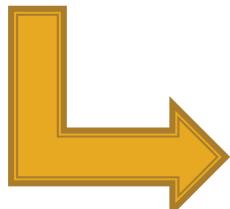
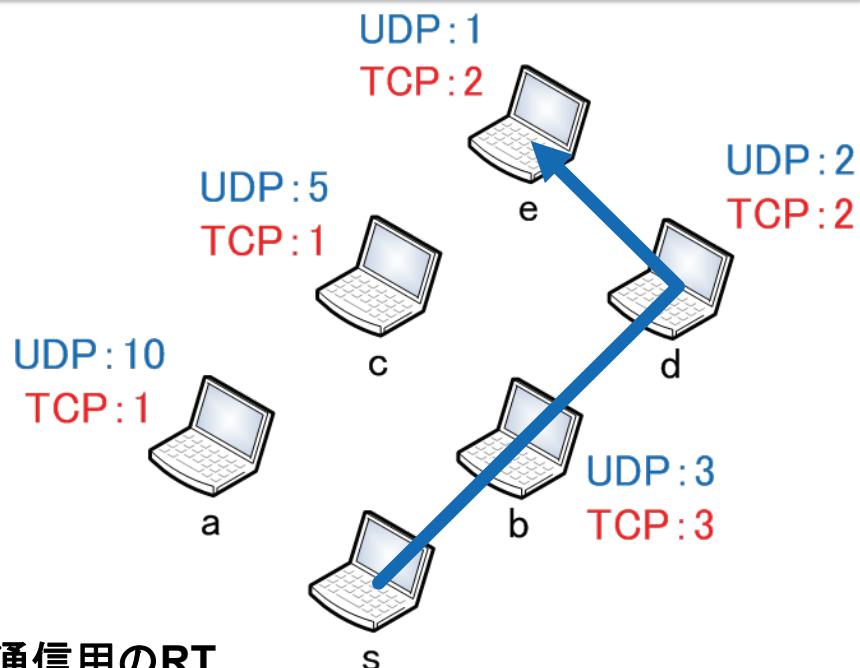
New

# PD-OLSRの経路生成 ③

ノードsのRCT

宛先	中継ノード	ホップ数	UDP Traffic	TCP Session
a	a	1	10	1
b	b	1	3	3
c	a	2	10	1
c	b	2	3	3
d	b	2	3	3
e	c, a	3	10	1
e	c, b	3	3	3
e	d, b	3	3	3

ノードsからノードeへのUDP信用経路

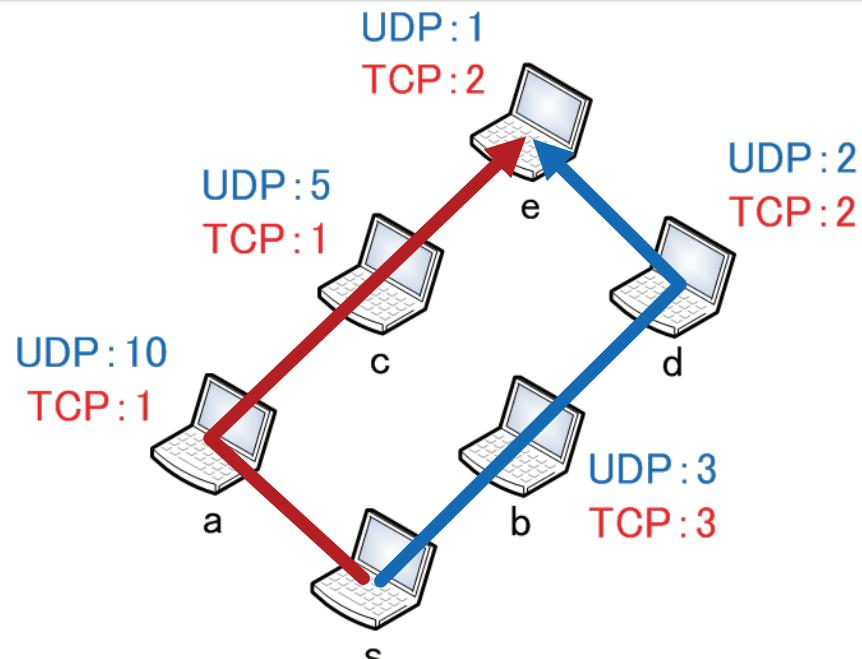


# PD-OLSRの経路生成 ④

ノードsのRCT

宛先	中継ノード	ホップ数	UDP Traffic	TCP Session
a	a	1	10	1
b	b	1	3	3
c	a	2	10	1
c	b	2	3	3
d	b	2	3	3
e	c, a	3	10	1
e	c, b	3	3	3
e	d, b	3	3	3

ノードsからノードeへのTCP通信用経路



→

宛先	次ホップ	NSB
a	a	22.0
b	b	12.8
c	a	22.0
c	b	12.8
d	b	12.8
e	a	22.0
e	b	12.8

$$NSB = \frac{X - UDP\text{Traffic}}{TCP\text{Session} + 1}$$

※X=54Mbps

ノードsのTCP通信用の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

# まとめ

## ■ 本発表

- 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するため、負荷の高い経路を選択してしまい、スループットの低下が起きる可能性がある
- TCPとUDPの混在環境において、異なる2種類のトラヒックの特性に着目して、UDPとTCP通信用に別々のルーティングテーブルを生成するPD-OLSRを提案

## ■ 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

ご清聴ありがとうございました

# 補足

# TCP・UDPの特性 詳細

## ■ UDP

- 端末側が意図した流量のトラヒックがそのままネットワークへ送出

## ■ TCP

- 輻輳制御によって順調にACKが返ってこればウィンドウサイズを拡大し帯域を有効に使おうとする
- パケットロスが発生するとウィンドウサイズが適切な大きさに調整され、ネットワークが更に輻輳することを防ぐ

## ■ つまり

- TCPとUDP通信が混在するネットワークのトラヒックは
  - ① 送出されるUDPパケットの合計よりUDPが占めるトラヒック量が定まり、残りの余裕のある帯域分を複数のTCPセッションが分け合う
  - ② UDPのパケットロスはそのまま消滅するが、TCPは再送制御を行いながらスループットが最大になるように輻輳制御が働く

TCP効率は、TCPの輻輳制御がうまく機能するかどうかによって決まる

# 提案方式 - 着目点 -

## ■ 着目点

- 現在、インターネット上を流れるトラヒックの9割以上はTCPによるものである[1]
- 今後、UDPによるVoIPや動画のストリーミング配信などの需要がさらに高まる

MANETにおいても同様な状況が起こりうると想定

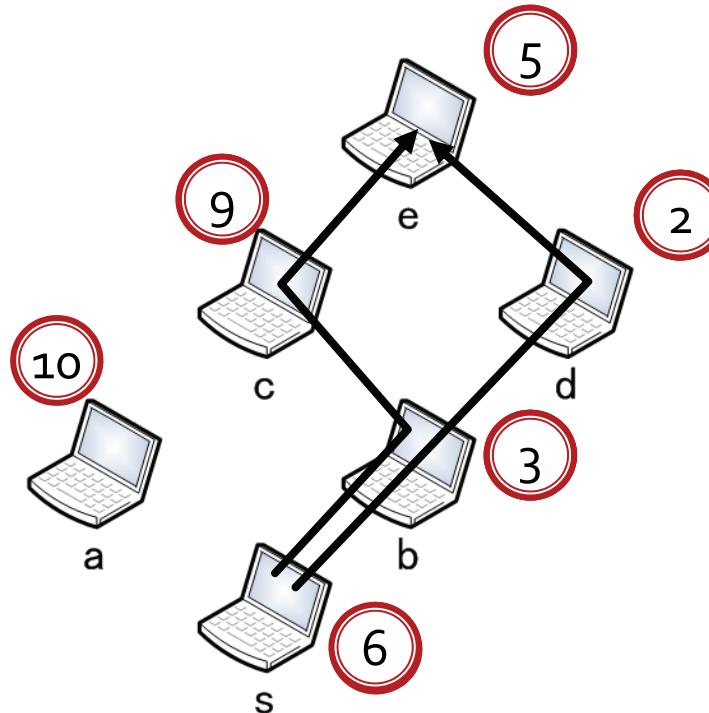


TCPとUDPの混在環境において、特徴が  
全く異なる2種類のトラヒックに着目

[1] Thompson, K., Miller, G. and Wilder, R.:Wide-Area Internet Traffic Patterns and Characteristics, IEEE/ACM Trans. Networking, Vol.11, No.6, pp.10–23 (1997)

# 経路全体のトラヒックの場合

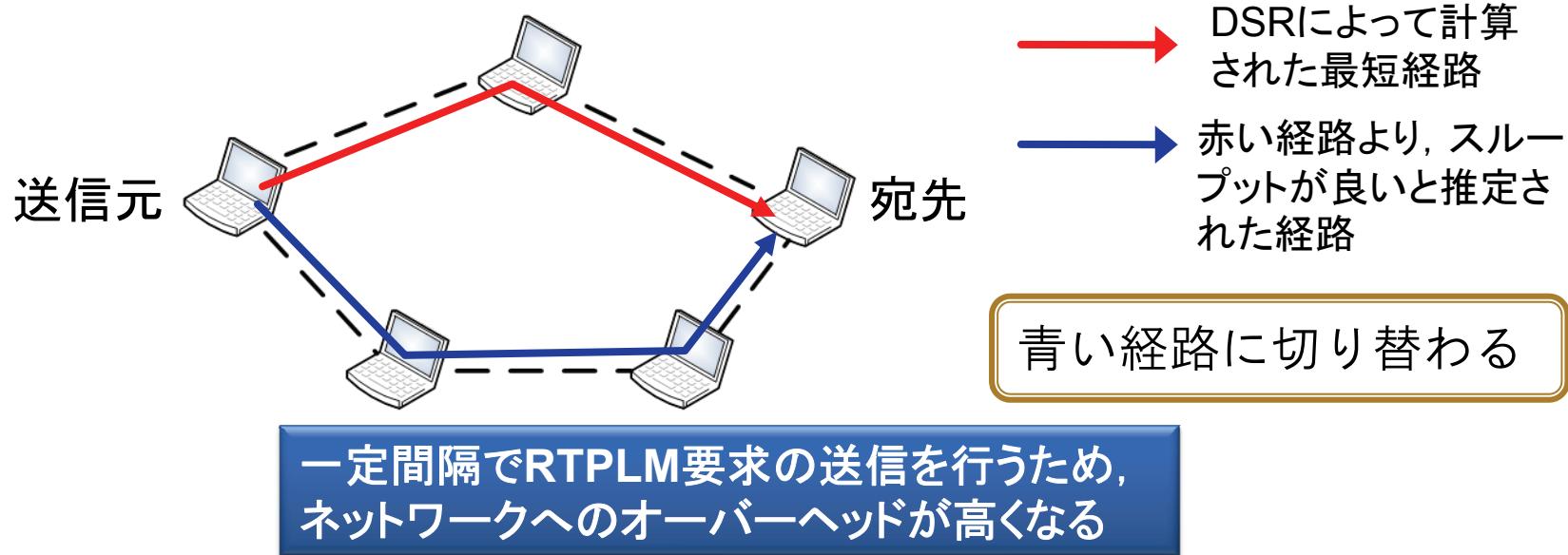
- 経路全体のトラヒックを経路選択の指標にすると
  - 通信によって経路のトラヒックが増え、更新時に経路が頻繁に切り替わってしまう
  - トラヒックの高いノードが含まれる経路ほど振動しやすくなる



- 次ホップノードのトラヒックで考えると経路の振動が起こりにくくなる

# 関連研究

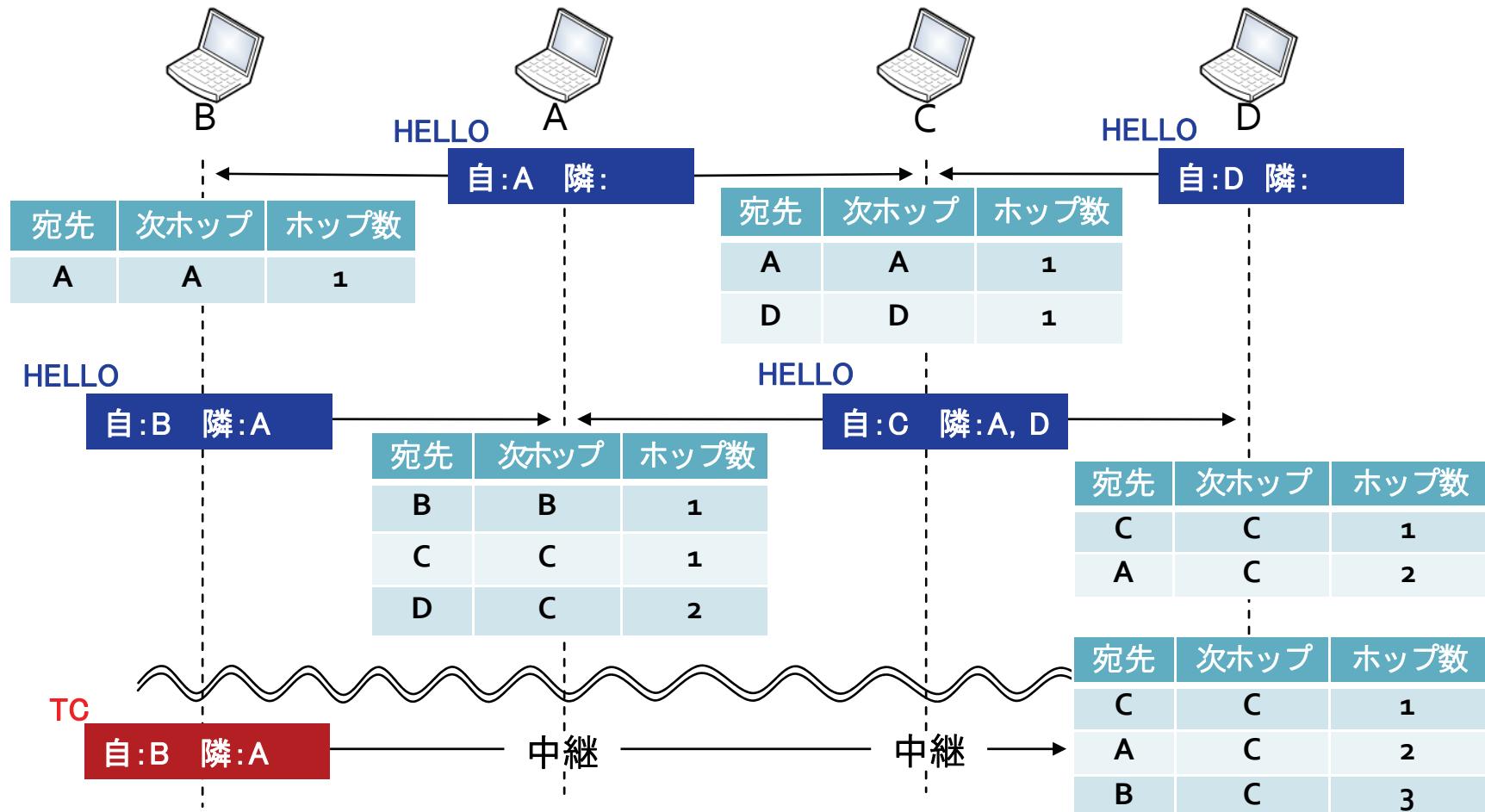
- ETR (Estimated-TCP-Throughput Maximization based Routing)
  - リアクティブ型のDSR (Dynamic Source Routing) を拡張
  - DSRによって計算された最短経路から、スループットが良いと推定される経路に切り替える
  - スループットの算出に必要な遅延と往復パケット損失率は、データ送信開始時から一定間隔で送信されるRTPLM (Round-Trip Packet Loss ratio Measurement) 要求とその応答によって収集



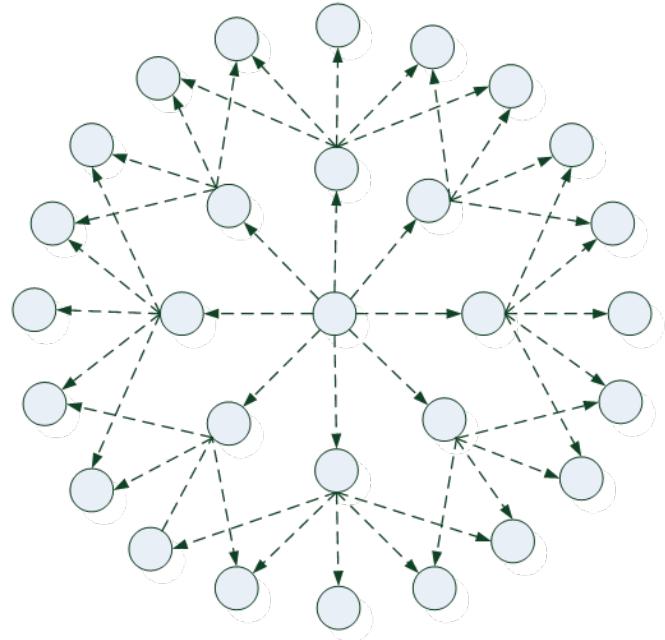
# 制御メッセージの送受信

## ■ HELLO, TCメッセージの送受信

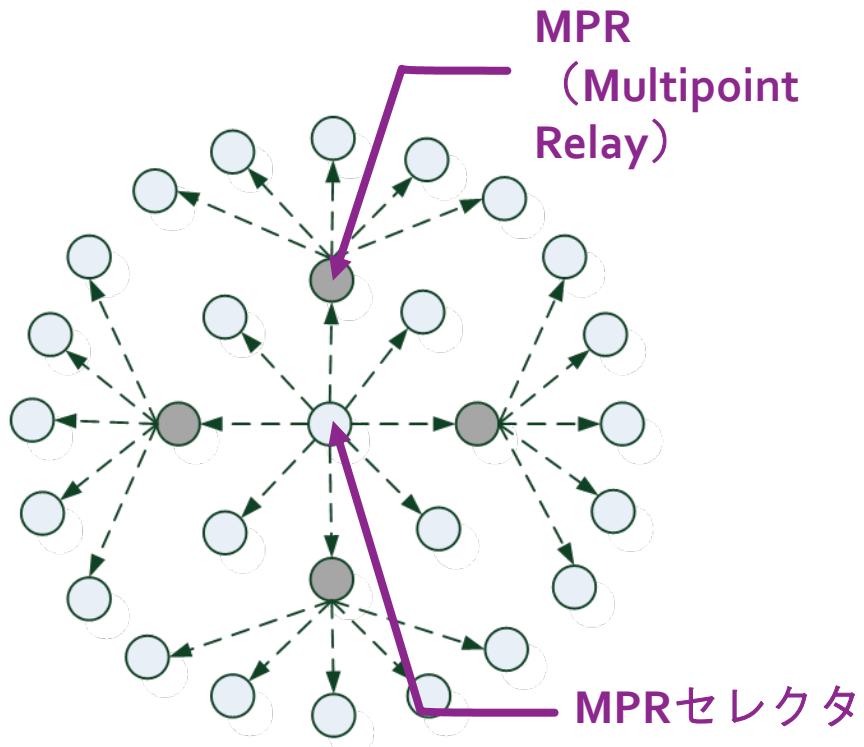
- HELLO, TCとともに自身のアドレスと隣接ノードのアドレスから構成



# OLSRのフラッディング

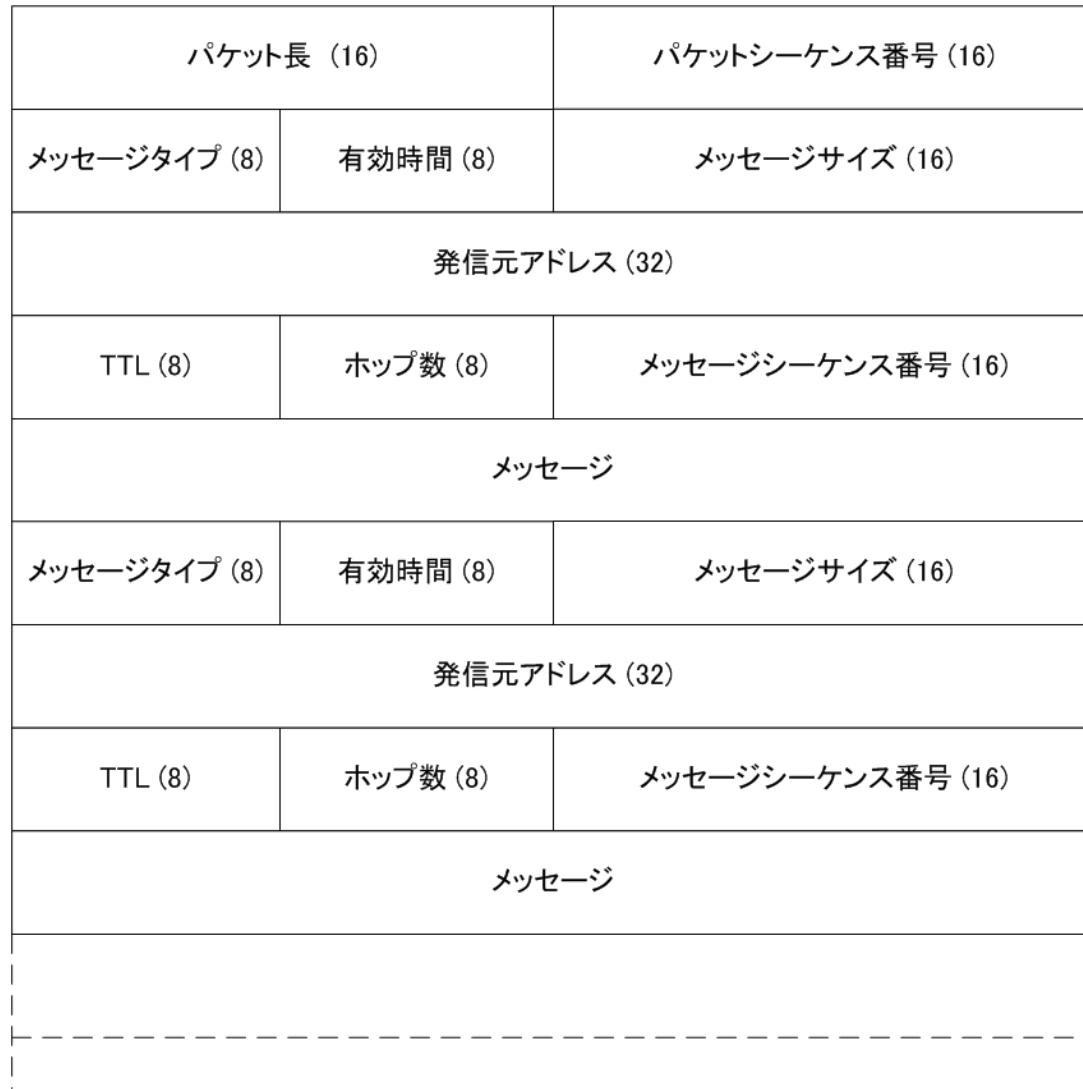


通常のフラッディング



OLSRのフラッディング

# OLSRのパケットフォーマット

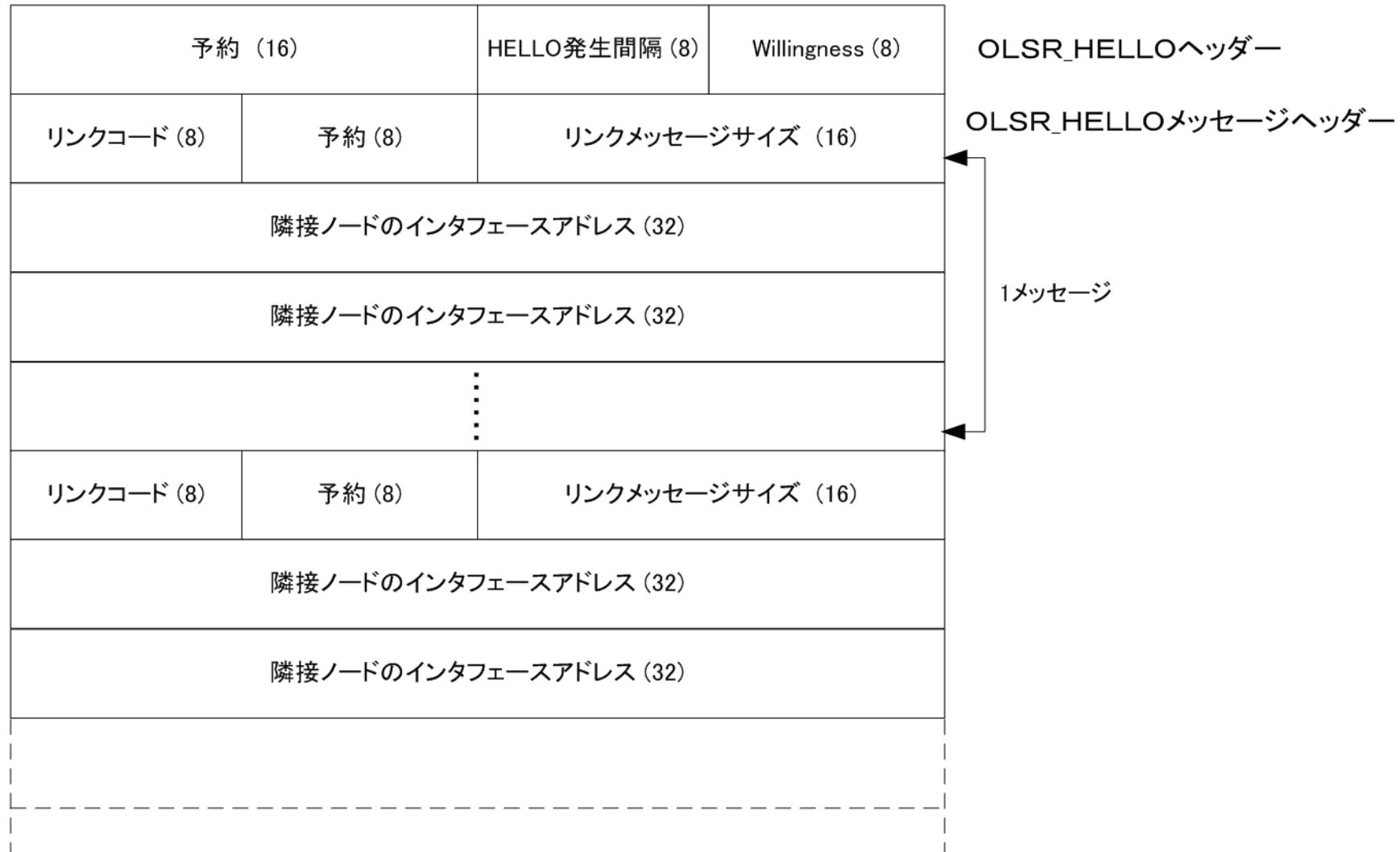


OLSRパケットヘッダー  
(4バイト)

OLSRメッセージヘッダー  
(12バイト)

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

# HELLOメッセージフォーマット



# TCメッセージフォーマット

