

アドホックネットワークにおける通信状態を考慮した ルーティング手法の提案

三嶋 勇太[†] 旭 健作[†] 鈴木 秀和[†] 渡邊 晃[†]

[†]名城大学大学院理工学研究科 〒468-8502 愛知県名古屋市天白区塩釜口 1-501

E-mail: †yuta.mikamo@wata-lab.meijo-u.ac.jp, †{asahi,hsuzuki,wtnbakr}@meijo-u.ac.jp

あらまし 有線で接続されていたアクセスポイント間を、アドホックネットワークによって接続する無線メッシュネットワークの研究が注目されている。無線メッシュネットワークのルーティングプロトコルはアドホックルーティングプロトコルや、それを改造したものが使用される。多くのアドホックルーティングプロトコルは、最短経路が複数存在する場合にどの経路を選択するかは実装に依存したものとなっている。本稿では、OLSR (Optimized Link State Routing) を拡張することにより、TCP, UDP それぞれの特性を生かせる経路選択が可能なアドホックネットワーク向けのルーティングプロトコル PD-OLSR (Protocol Dependent-OLSR) を提案する。

キーワード アドホックネットワーク, ルーティングプロトコル

Proposal of Routing in Ad-hoc Networks Considering the Communication Status

Yuta MIKAMO[†], Kensaku ASAHI[†], Hidekazu SUZUKI[†], and Akira WATANABE[†]

[†] Graduate School of Science and Technology, Meijo University Shiogamaguchi 1-501, Tempaku-ku,
Nagoya, 468-8502 Japan

E-mail: †yuta.mikamo@wata-lab.meijo-u.ac.jp, †{asahi,hsuzuki,wtnbakr}@meijo-u.ac.jp

Abstract Studies on the wireless mesh network that connects access points (AP) by an ad-hoc network have been drawing much attention. As the routing protocol for the wireless mesh network, ad-hoc routing protocol or its improved version is used. In many of such ad-hoc routing protocols, it depends on the implementation which route to be chosen when there exist a number of shortest routes. In this paper, we propose a routing in ad-hoc network that can make route selection that takes advantage of features of both TCP and UDP, by extending the Optimized Link State Routing(OLSR).

Key words Ad-hoc Network, Routing Protocol

1. ま え が き

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。中でも、インフラストラクチャモードにおいて有線接続されていたアクセスポイント（以下 AP と記述）間を、アドホックネットワークで接続する無線メッシュネットワークの研究に注目が集まっている。

アドホックネットワークの経路を生成するには、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) において、現在まで多くの方式が標準化されている [1-7]。しかし、これらの

方式は、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）を探索することを目的としており、最短経路が複数存在する場合にどの経路を選択するかは実装に任されている場合が多い。そのため、トラフィ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、スループットが低下するという課題がある [8]。

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) [9] の経路選択では、リンク切断が長時間起らない、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できるため、安定した経路



アドホックネットワークにおける 通信状態を考慮したルーティング手法の提案

名城大学大学院 理工学研究科
三鴨勇太 旭 健作 鈴木秀和 渡邊 晃

背景

- ▶ スマートフォンやタブレット端末など無線LANを利用可能な端末の普及に伴い、アドホックネットワークに関する研究が注目されている
- ▶ アドホックルーティングプロトコル
 - アドホックネットワークに特化したルーティングプロトコル
 - 周辺ノードとやりとりしRT(Routing Table)を生成
- ▶ 利用形態
 - インフラを利用できない環境
 - 災害時, イベント会場など一時的な通信
 - 無線メッシュネットワーク
 - アクセスポイント同士がアドホックモードで接続

既存方式:OLSR

OLSRの概要

(Optimized Link State Routing)

- ▶ プロアクティブ型
 - 通信要求発生前からRTを生成しておく
- ▶ 周辺ノードの情報を収集することによってRTを生成
 - 各ノードは定期的に制御メッセージを送受信
- ▶ 制御メッセージ
 - HELLOメッセージ
 - 各ノードが持つ情報を通知
 - 2秒毎に隣接ノードへブロードキャスト
 - TCメッセージ
 - ネットワークトポロジーを通知
 - 5秒毎にネットワーク全体にフラッディング

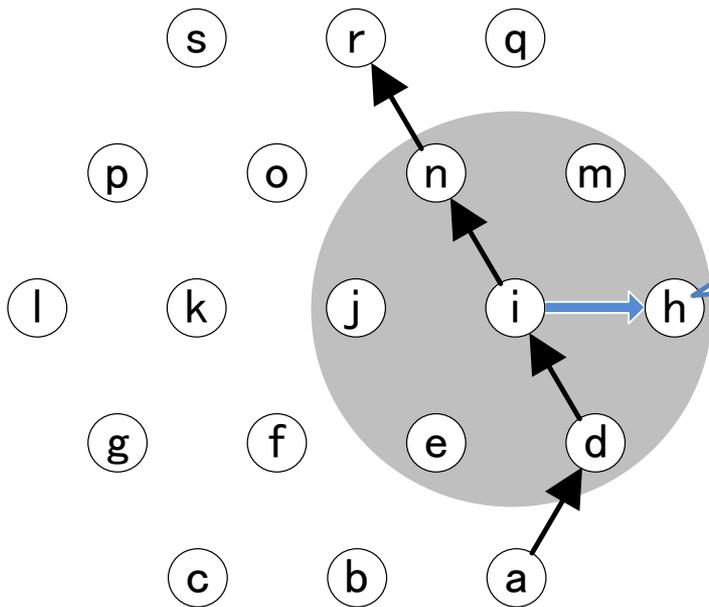
制御メッセージには
リンク情報のみ

OLSRのRT

Dest: 宛先ノード
 Next: 次ホップノード
 hop: 宛先ノードまでのホップ数

▶ 制御メッセージのやりとりによってRTが生成されていく

Dest	Next	hop
b	b	1
d	d	1



ノード数 : 19台
 電波到達範囲: 隣接ノードまで
 既に行われている通信: i→h

OLSRで生成される経路例

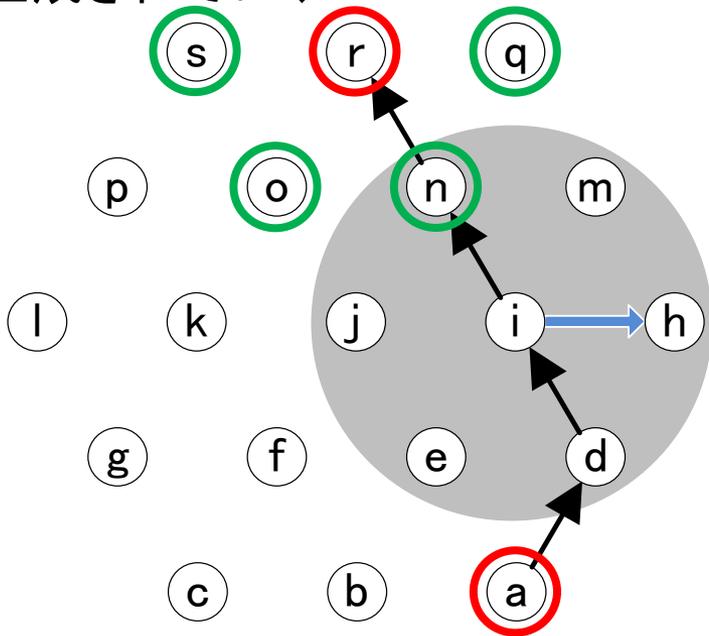
● 高トラフィックゾーン
 ○ ノード

j	e	2
g	b	3
k	e	3
m	d	3
n	d	3
o	e	3
l	e	4
p	e	4
q	d	4

OLSRのRT

Dest: 宛先ノード
 Next: 次ホップノード
 hop: 宛先ノードまでのホップ数

▶ 制御メッセージのやりとりによってRTが生成されていく



OLSRで生成される経路例

● 高トラフィックゾーン
 ○ ノード

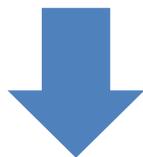
Dest	Next	hop	Dest	Next	hop
b	b	1	b	b	1
d	d	1	d	d	1
e	e	1	e	e	1
c	b	2	c	b	2
f	e	2	f	e	2
h	d	2	h	d	2
i	d	2	i	d	2
j	e	2	j	e	2
g	b	3	g	b	3
k	e	3	k	e	3
m	d	3	m	d	3
n	d	3	n	d	3
o	e	3	o	e	3
l	e	4	l	e	4
p	e	4	p	e	4
q	d	4	q	d	4
r	d	4	r	d	4
s	e	4	s	e	4

New

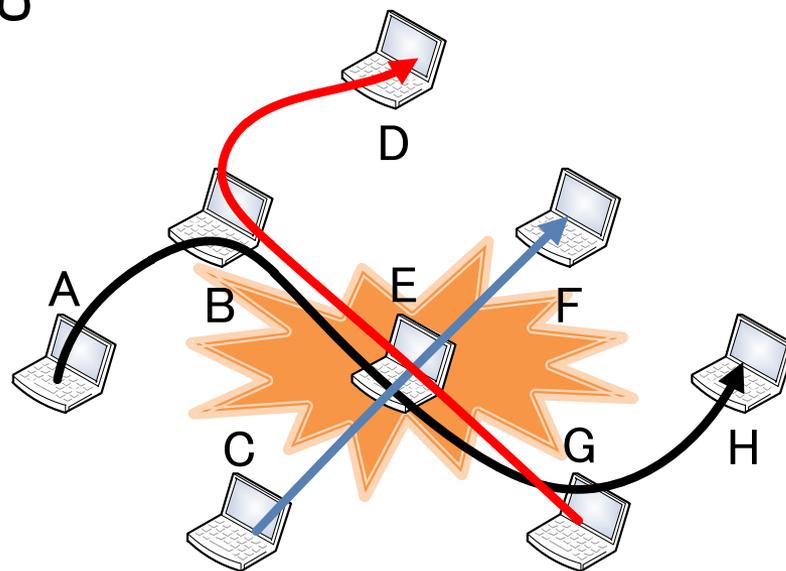
課題1

- ▶ 多くのアドホックルーティングプロトコルは、経路の中継ホップ数が最小となる経路を選択
- ▶ 最短ホップ数の複数の経路の中からどの経路が選ばれるかは定義されていない
- ▶ 複数の通信で同一のノードを経由する経路が選択され、トラフィックが集中する可能性も

パケットロスが多発



スループットが低下



UDPとTCPの特性

▶ UDP通信

- 端末側が意図した流量のトラフィックがそのままネットワークへ送出

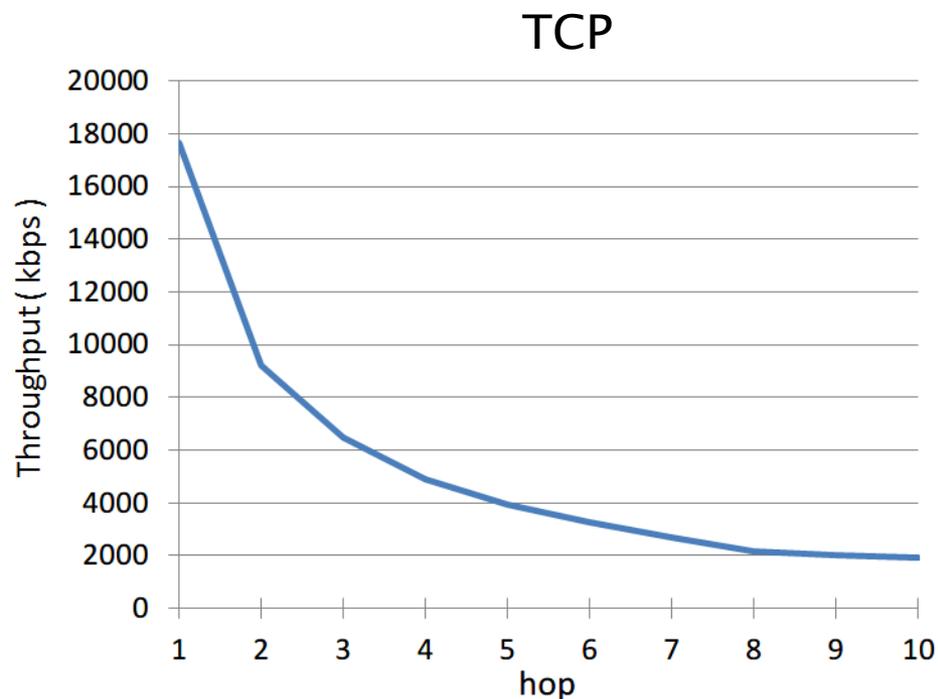
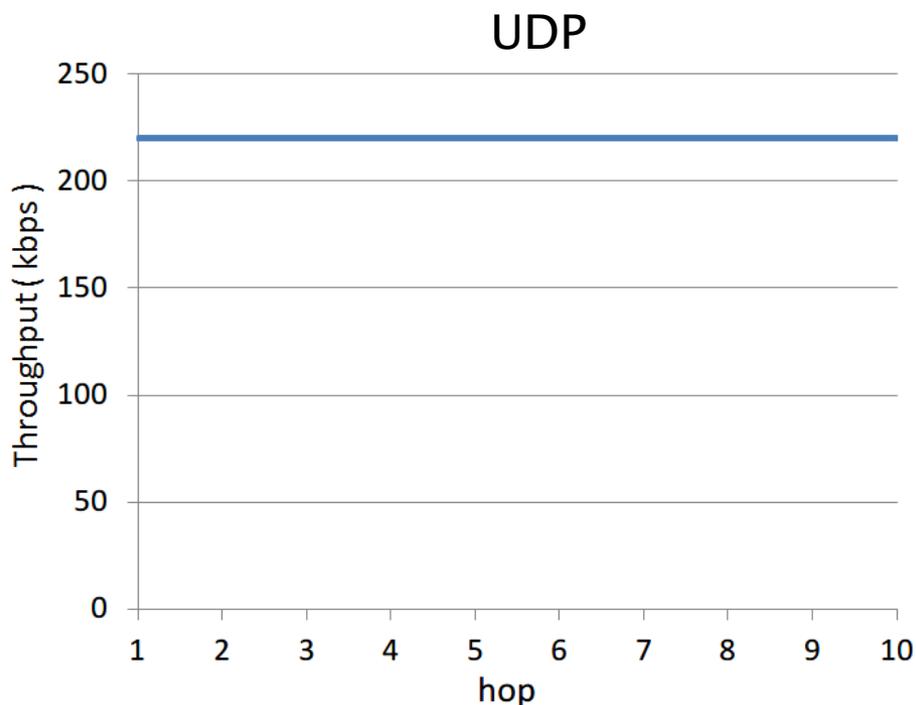
▶ TCP通信

- 輻輳制御によって順調にACKが返ってきた場合はウィンドウサイズを拡大し帯域を使いきろうとする

UDP通信とTCP通信が混在するネットワークのトラフィックは、送出されるUDPパケットの合計からUDPが占めるトラフィック量が定まり、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

UDPとTCPのマルチホップ通信

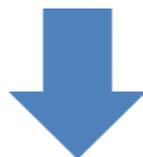
- ▶ 1～10ホップのスループットをシミュレーションで測定
 - ノードを一直線上に配置



TCPのスループットはホップ数に反比例して低下

課題2

- ▶ UDPとTCPでは通信性質が異なる
 - トラフィック発生量
 - ホップ数によるスループット変化
- ▶ 既存のルーティングプロトコルでは2種類の通信は同一RTを使用



同一経路を用いることによって
TCP通信のスループットが低下する可能性

提案方式:PD-OLSR

提案方式: PD-OLSR (Protocol Dependent-OLSR)

- ▶ OLSRを改造
- ▶ 目的
 - UDP用とTCP用別々にRTを生成
 - トラフィックの高いノードを避けた経路選択

PD-OLSR 通信状態指標

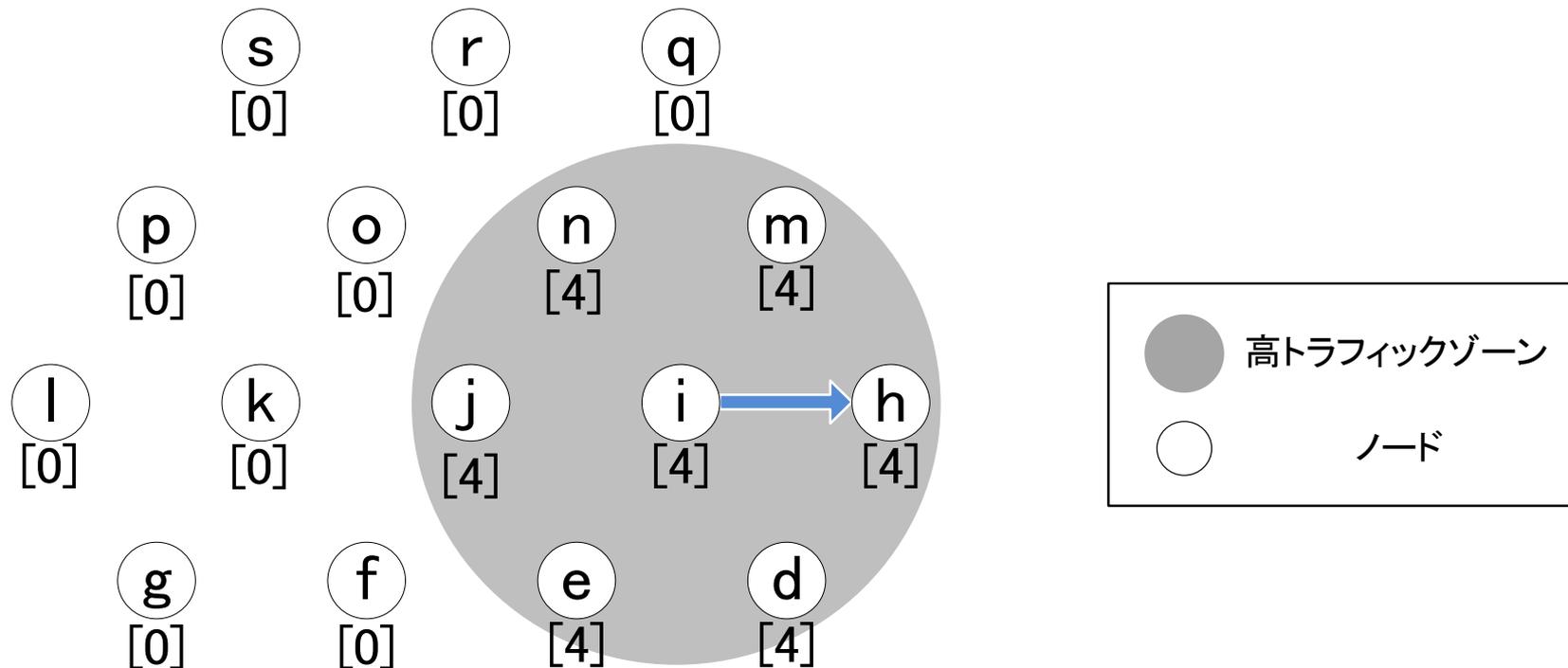
- ▶ UDP通信: UDP Traffic
 - 自身が検出するネットワーク上のキャリアの総量
- ▶ TCP通信: TCP Session
 - 自身が検出するTCPセッション数の合計
 - 各セッションがウィンドウサイズを増減
 - セッション間の公平を保つため
- ▶ 各ノードが計測

制御メッセージ

- ▶ OLSRの制御メッセージの仕組みはそのまま用いる
- ▶ トラフィックの情報を制御メッセージ(HELLO,TC)に追加,
ネットワーク全体に広告
- ▶ ネットワーク内のノードのトラフィック情報を共有

動作

環境



- ▶ []内の数字は各ノードのトラフィック
- ▶ i→hの通信のトラフィック量:4

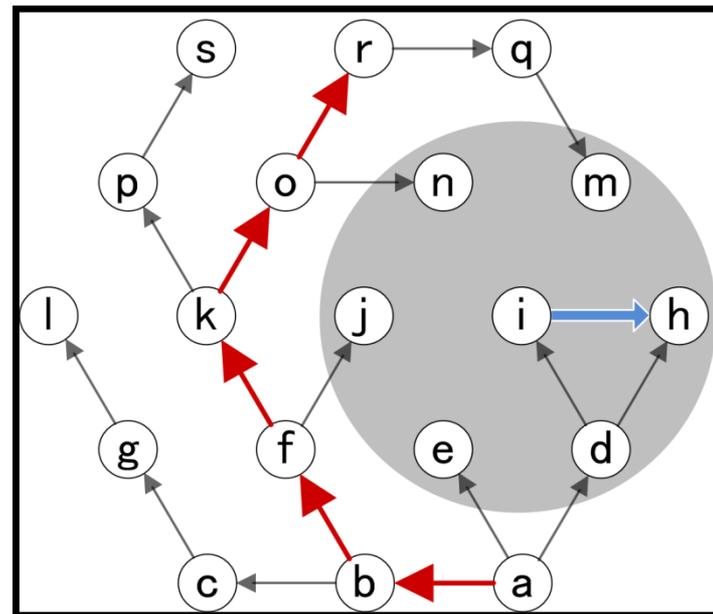
経路探索

- ▶ 制御メッセージによって共有した情報をもとに行う
- ▶ アルゴリズム:ダイクストラ法
 - グラフ理論における最短経路問題解決アルゴリズム
- ▶ UDPではホップ数によってスループットが低下しないためホップ数を増やした経路も許容できる
 - すべての経路から最適なものを選択する
- ▶ 経路の合計コストが最小のものを選択
 - コスト:トラフィック量

経路探索結果

冗長経路を含めたすべての経路を探索し
合計コストが一番小さいものを選択

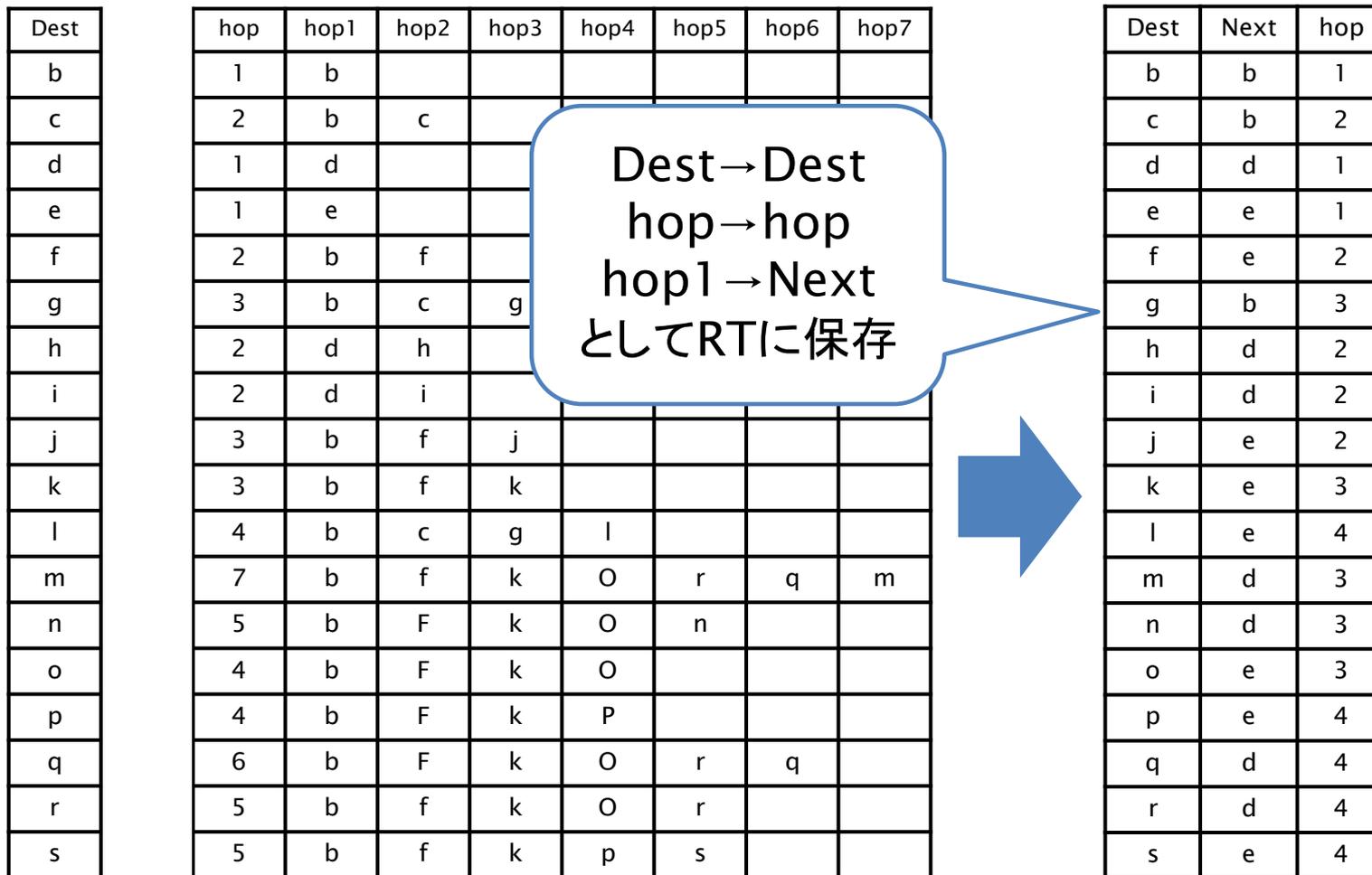
Dest	Cost	hop	hop1	hop2	hop3	hop4	hop5	hop6	hop7
b	0	1	b						
c	0	2	b	c					
d	4	1	d						
e	4	1	e						
f	0	2	b	f					
g	0	3	b	c	g				
h	8	2	d	h					
i	8	2	d	i					
j	4	3	b	f	j				
k	0	3	b	f	k				
l	0	4	b	c	g	l			
m	4	7	b	f	k	o	r	q	m
n	4	5	b	f	k	o	n		
o	0	4	b	f	k	o			
p	0	4	b	f	k	p			
q	0	6	b	f	k	o	r	q	
r	0	5	b	f	k	o	r		
s	0	5	b	f	k	p	s		



ノードaから各ノードへの経路

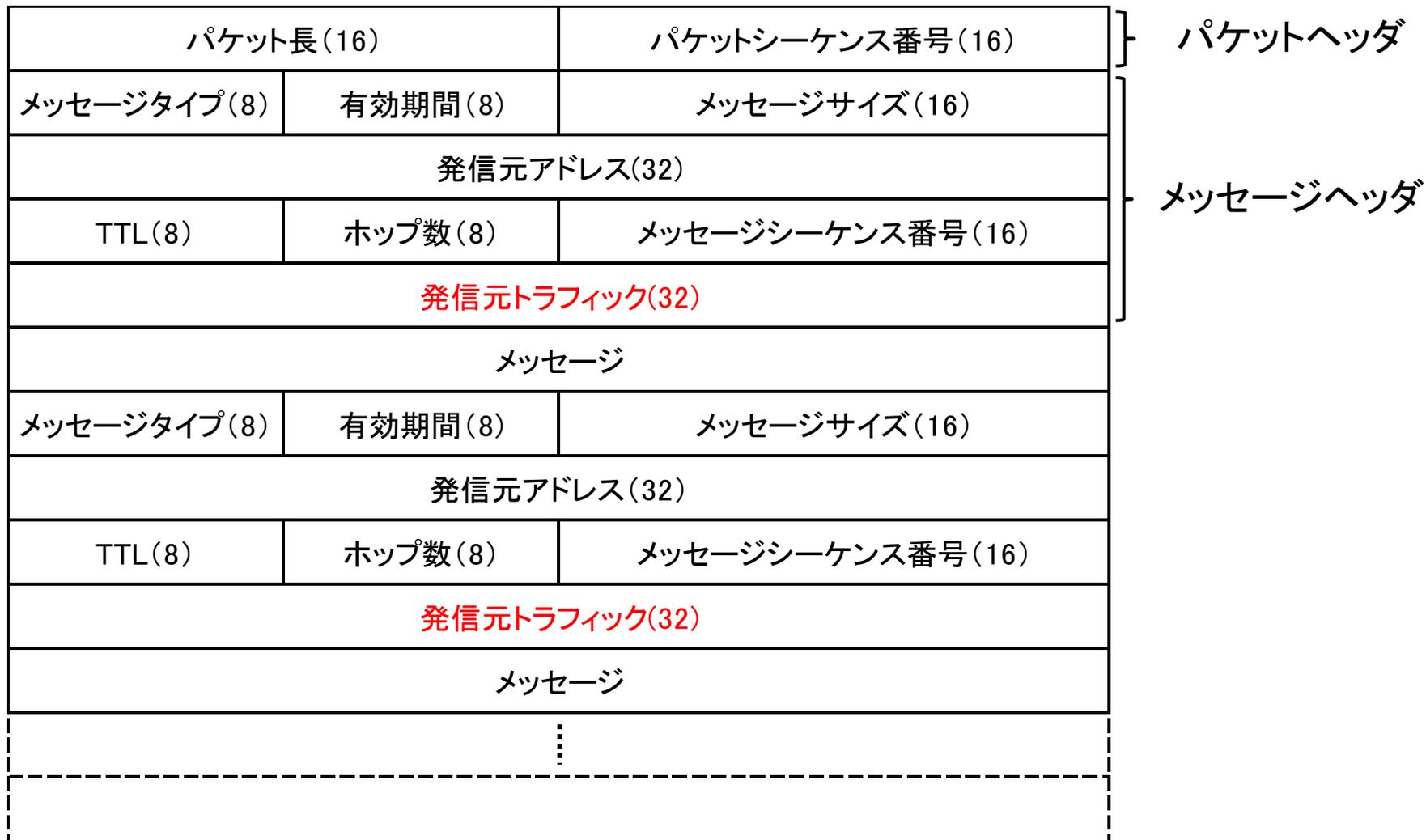
ノードaによる経路探索結果

RT生成

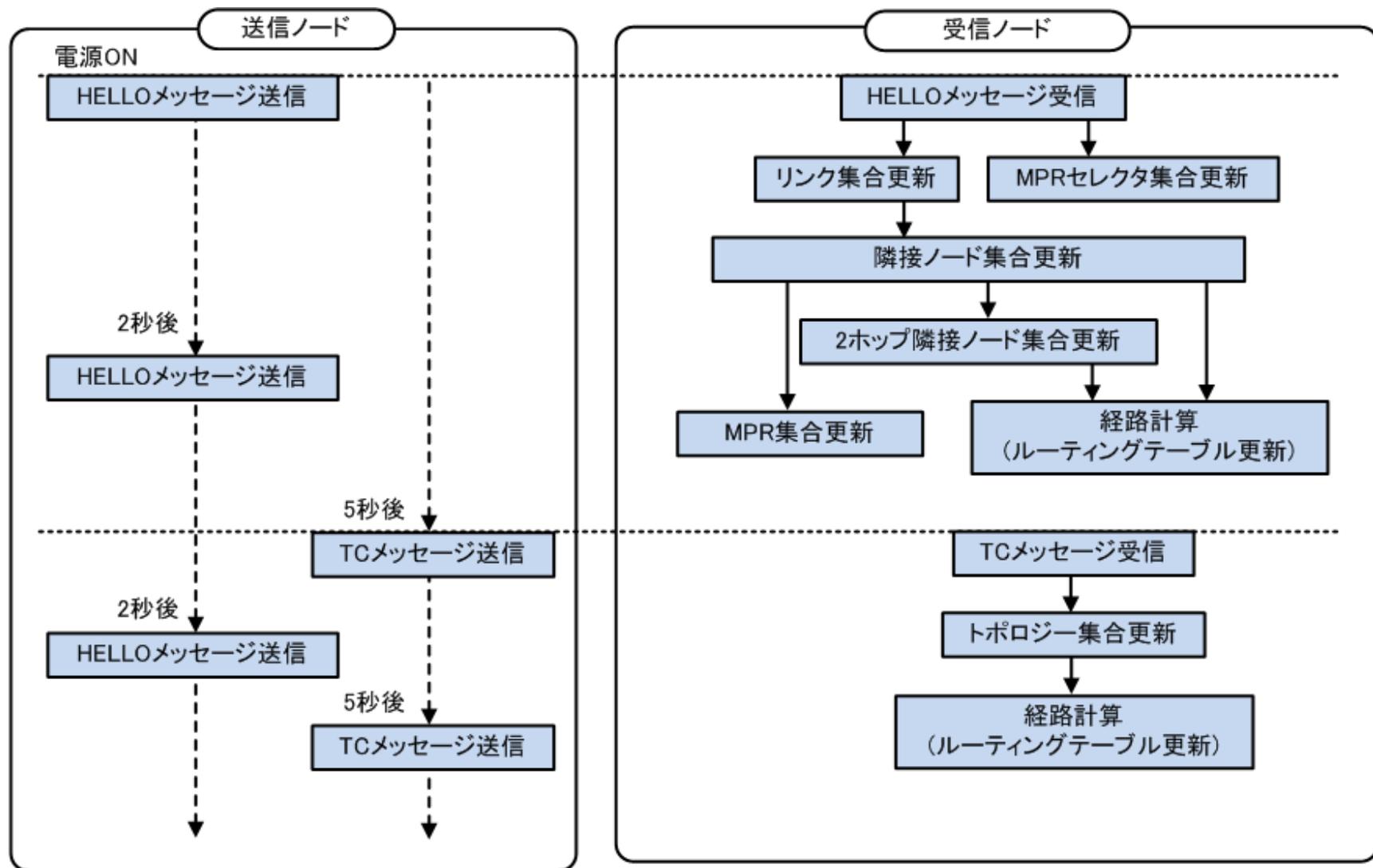


実装

PD-OLSRパケットフォーマット

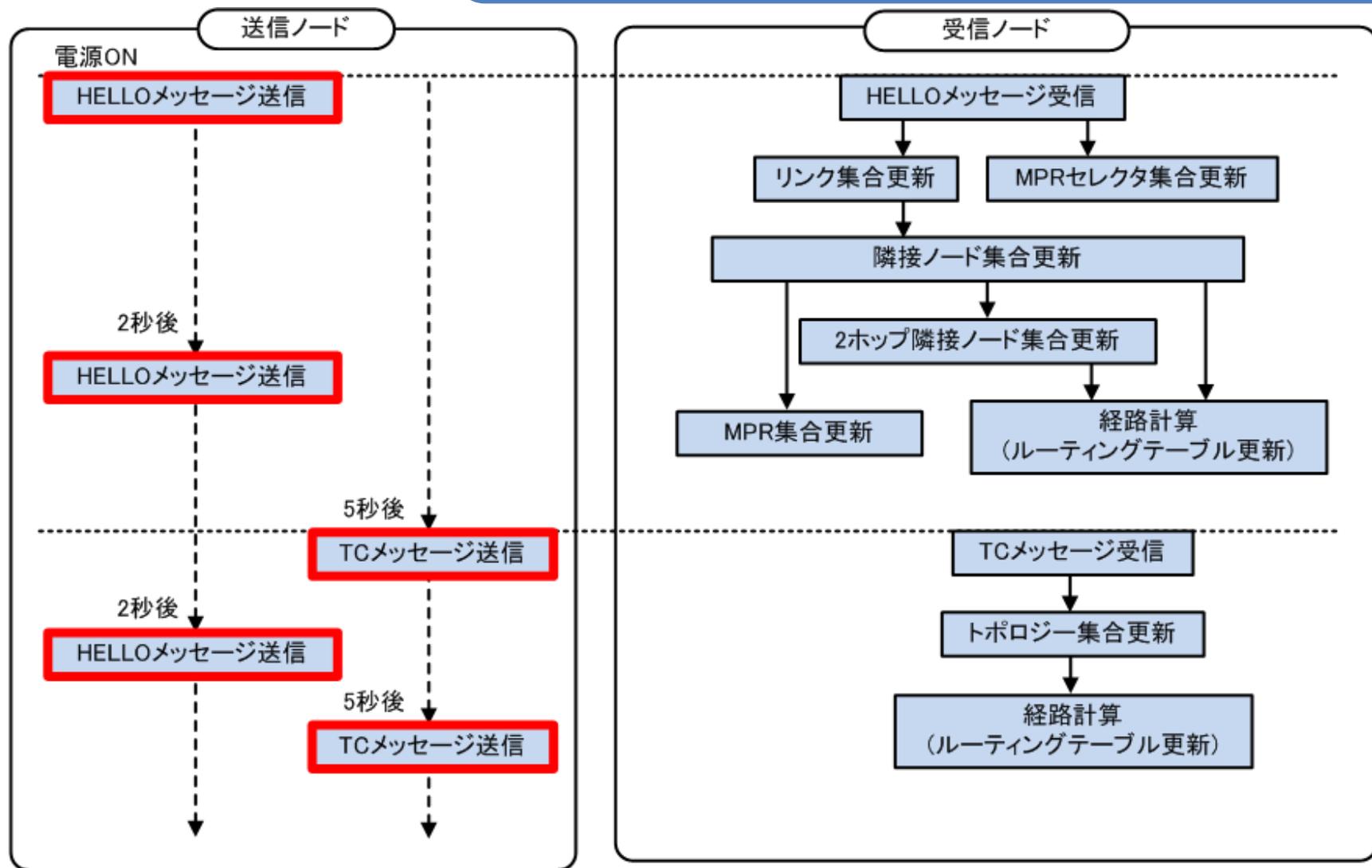


拡張箇所



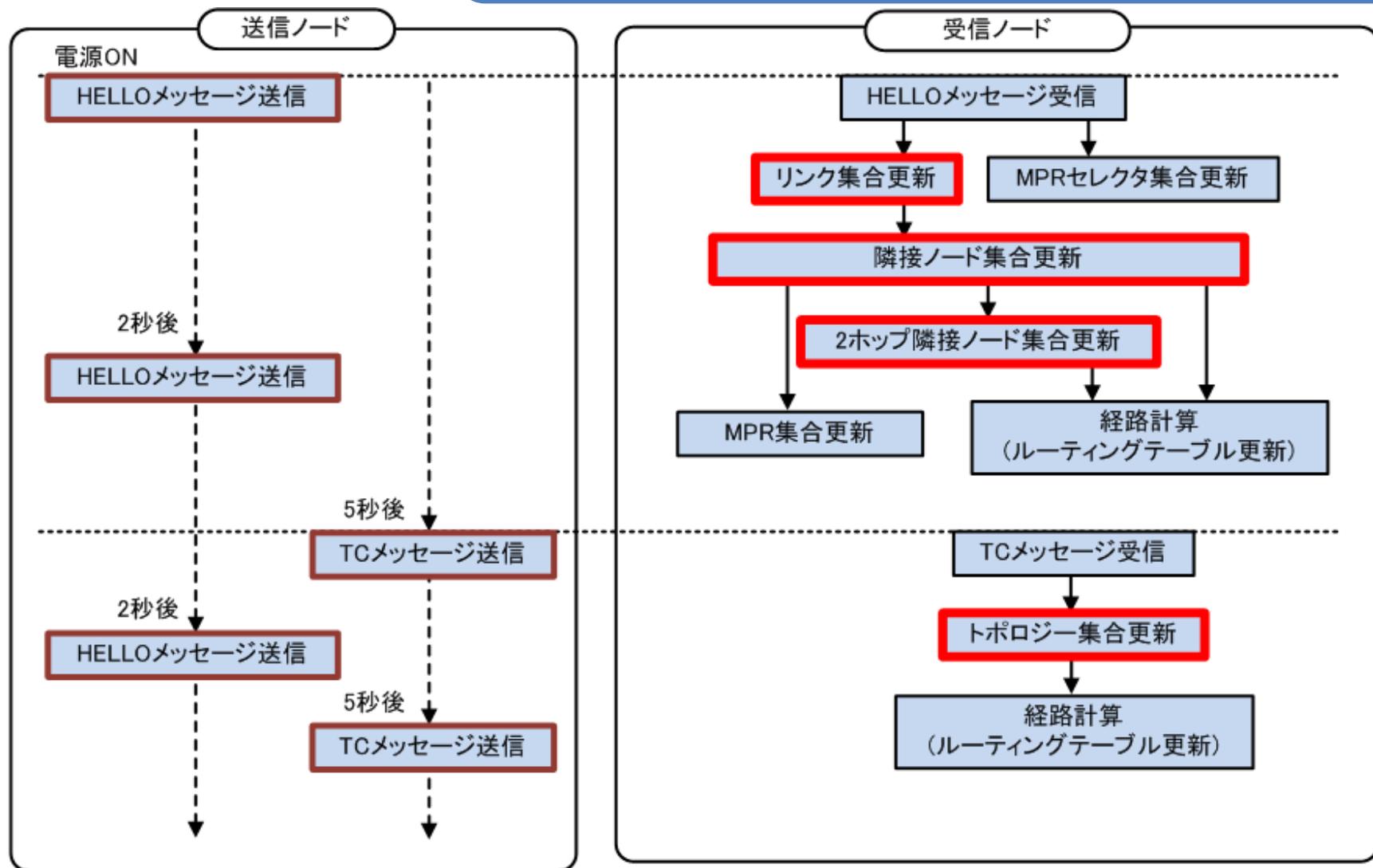
(1) HELLO, TCメッセージにトラフィック情報を追加

拡張箇所



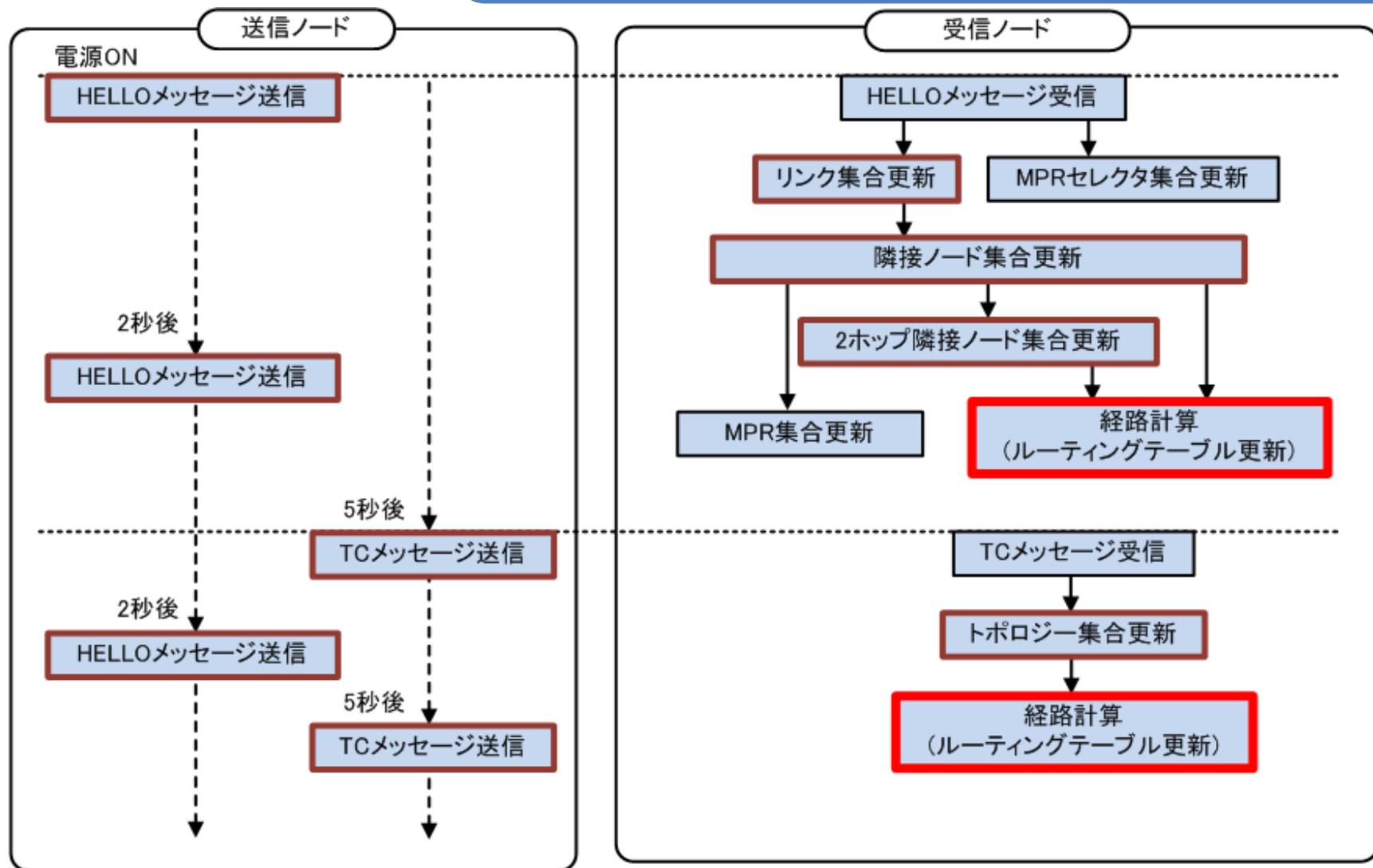
拡張箇所

- (1) HELLO, TCメッセージにトラフィック情報を追加
- (2) 各リポジトリ集合にトラフィック情報を追加



拡張箇所

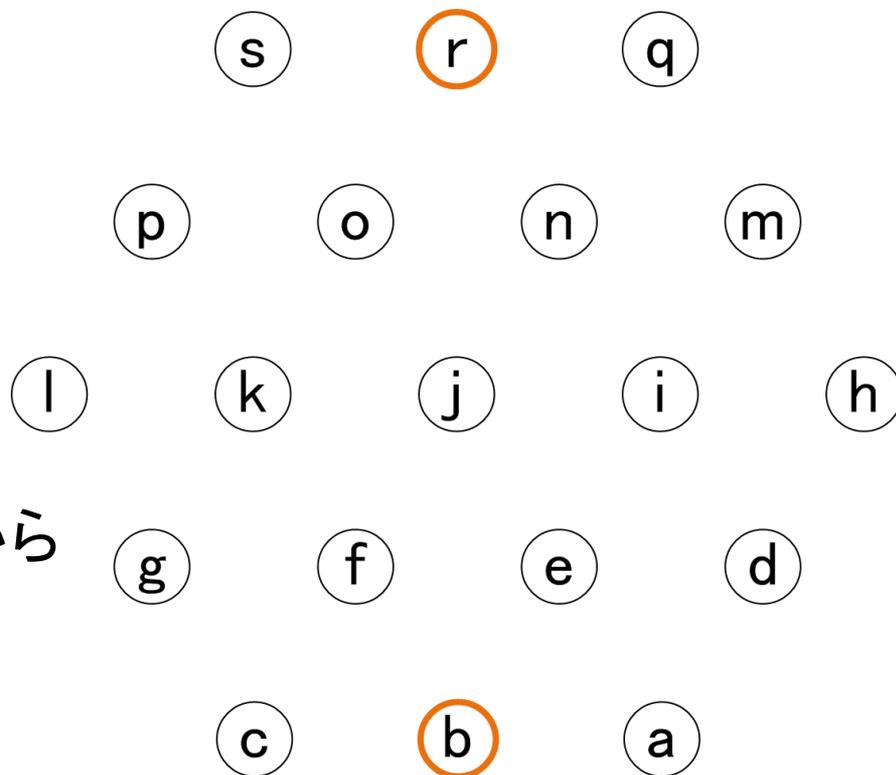
- (1) HELLO, TCメッセージにトラフィック情報を追加
- (2) 各リポジトリ集合にトラフィック情報を追加
- (3) 経路の合計コストを基準とするRT生成モジュール



動作検証

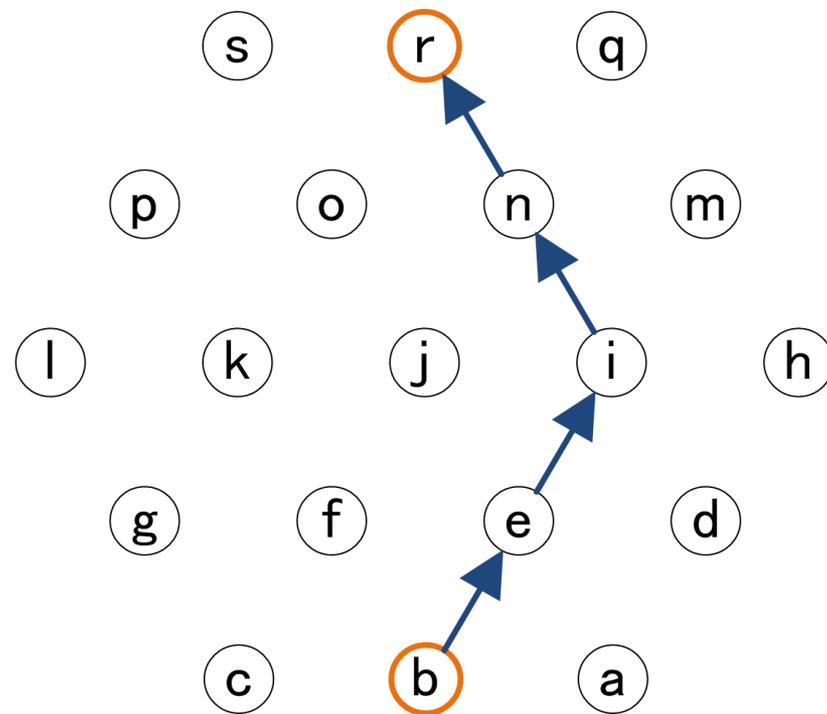
環境

- ▶ シミュレータ: NS-2
- ▶ ノード数: 19台
- ▶ 電波到達範囲: 隣接ノード
- ▶ IEEE802.11g
- ▶ シミュレーション開始30秒後から20秒間UDP通信
- ▶ ノードb → r



動作検証 (1 / 3)

最短経路で通信開始



動作検証 (2/3)

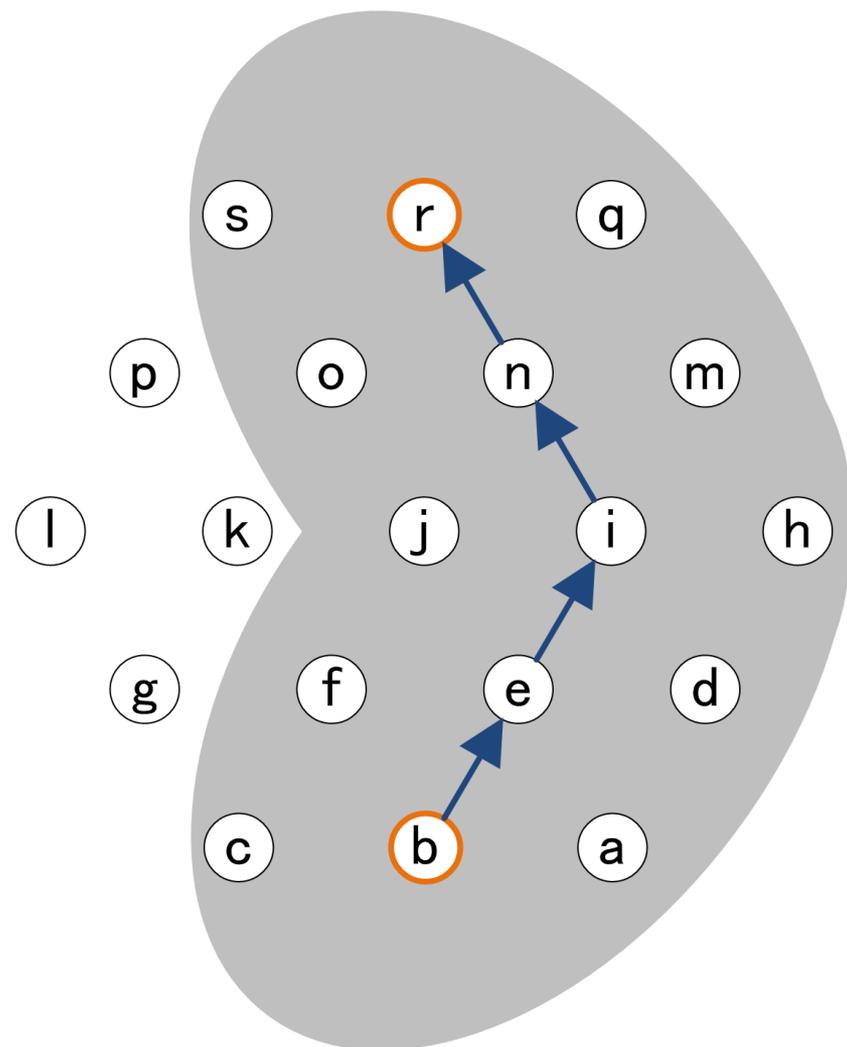
最短経路で通信開始



トラフィック検出



制御メッセージ送受信



動作検証 (3 / 3)

最短経路で通信開始



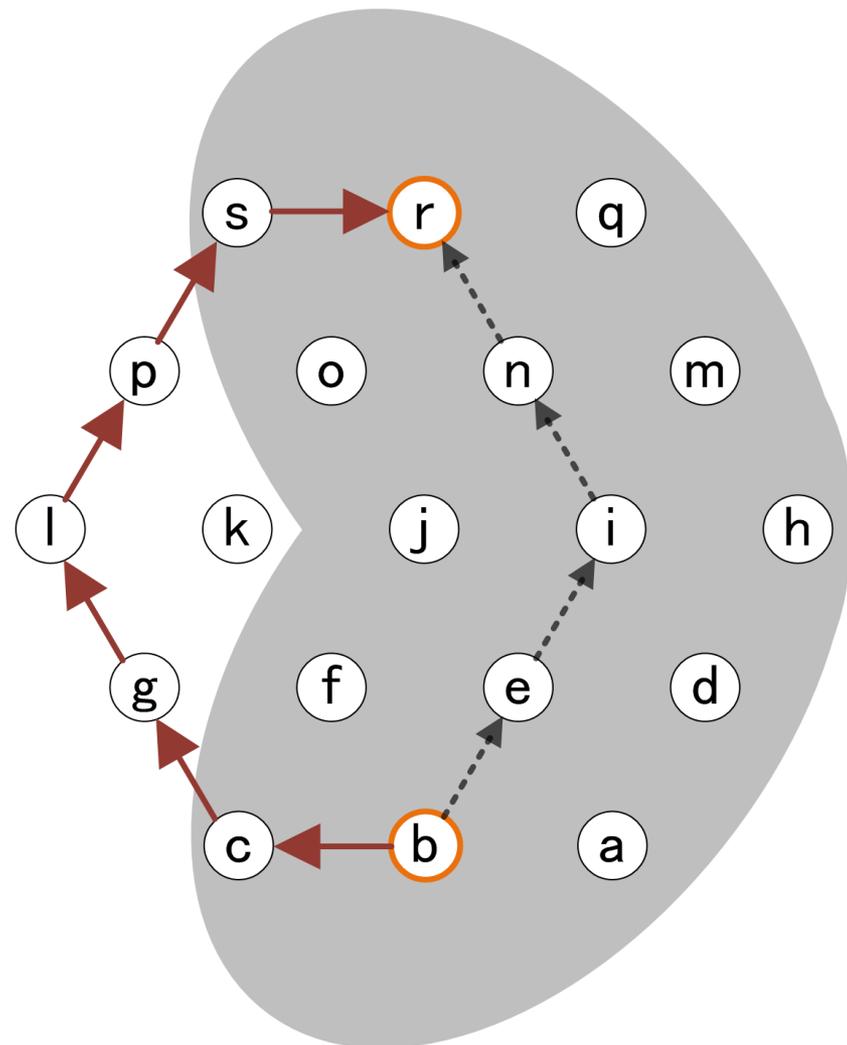
トラフィック検出



制御メッセージ送受信



経路切替



動作検証 (3 / 3)

最短経路で通信開始



トラフィック検出



制御メッセージ送受信

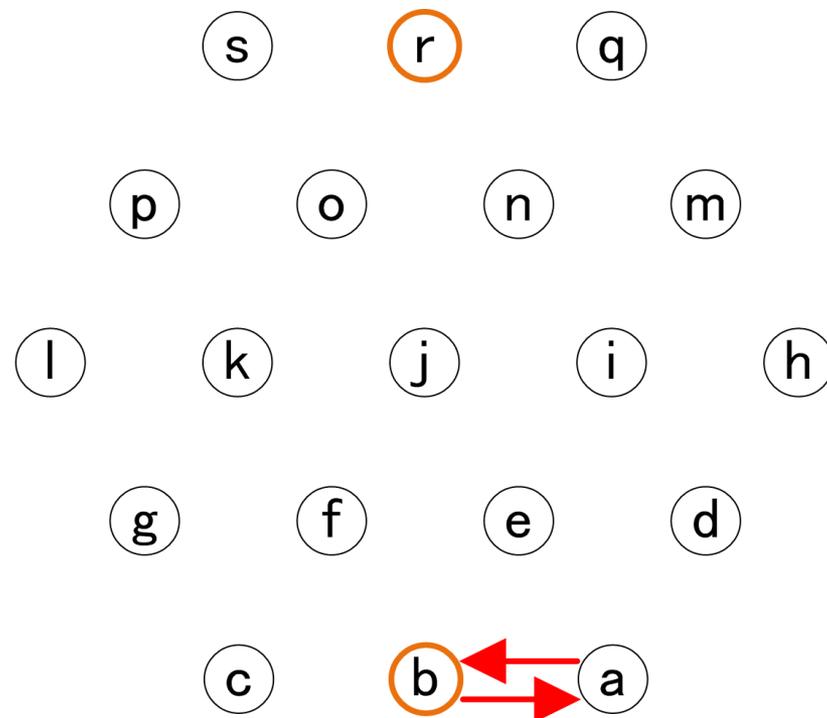


経路切替



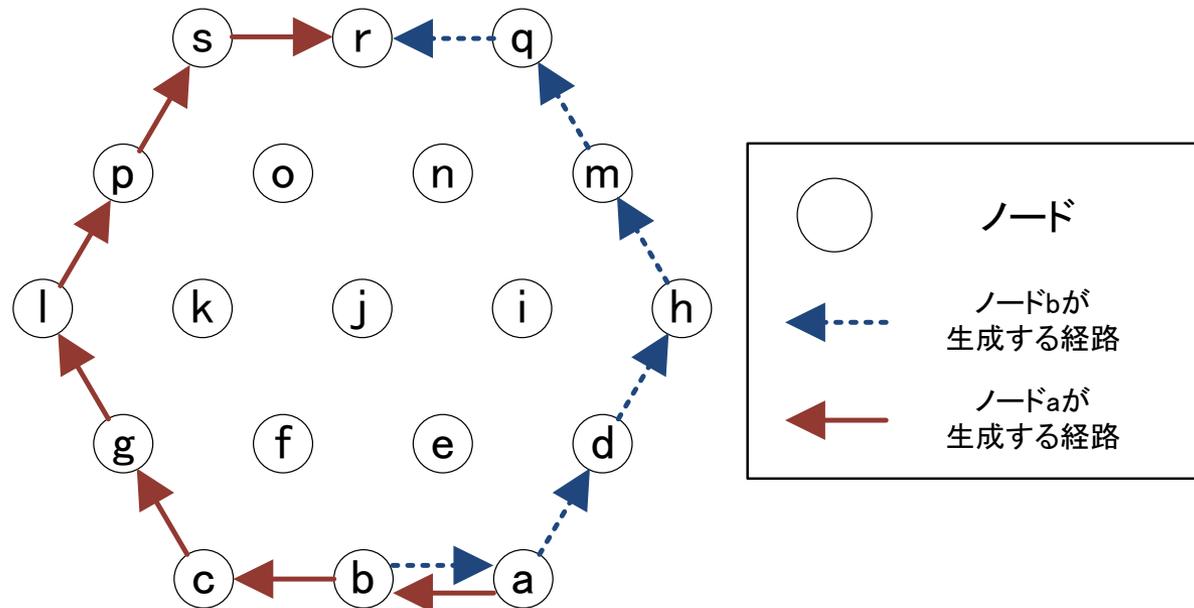
さらに経路が切り替わると...

経路ループ発生



経路ループ

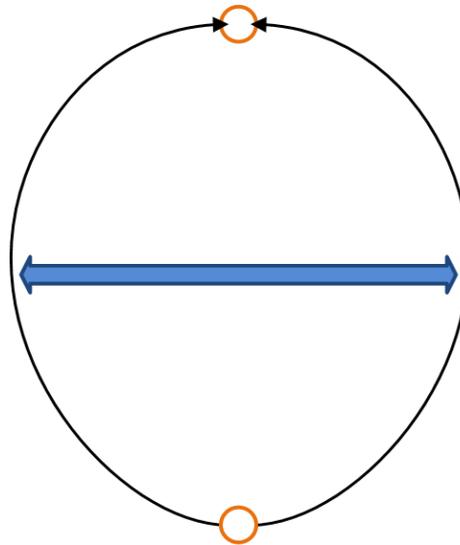
- ▶ ノードa, b間で確認
- ▶ 発生時の経路探索結果



隣接ノードで逆方向の経路を探索している

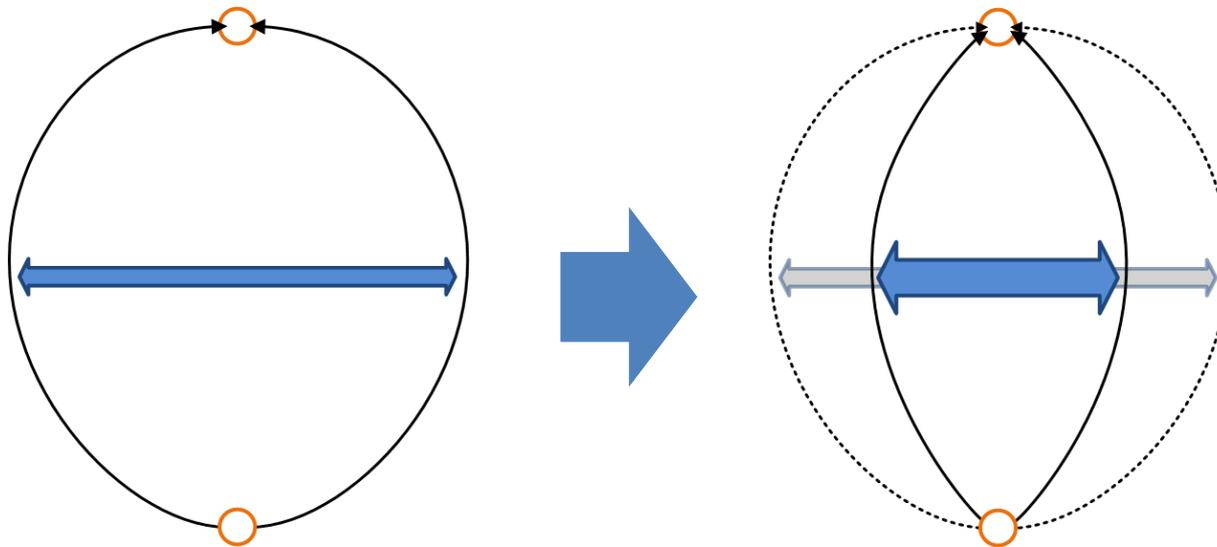
経路ループ対策 (検討)

- ▶ 経路の選択範囲を制限する
- ▶ 最短経路に対してホップ数を伸ばす量を制限
→ 隣接ノードで逆方向の経路を選択しない



経路ループ対策 (検討)

- ▶ 経路の選択範囲を制限する
- ▶ 最短経路に対してホップ数を伸ばす量を制限
→ 隣接ノードで逆方向の経路を選択しない



ある程度制限することで経路ループが発生しなくなる

むすび

▶ 本発表

- OLSRを拡張することによって, UDP用とTCP用それぞれのRTを別々に生成し, 経路上の通信状態を考慮して経路生成ができるプロトコルPD-OLSRを提案した

▶ 今後

- ホップ数の制限方法の検討
- 提案方式のシミュレータへの実装を完了し, RTをUDPとTCPで分けたことによる効果を検証する

補足資料

今までの実装評価

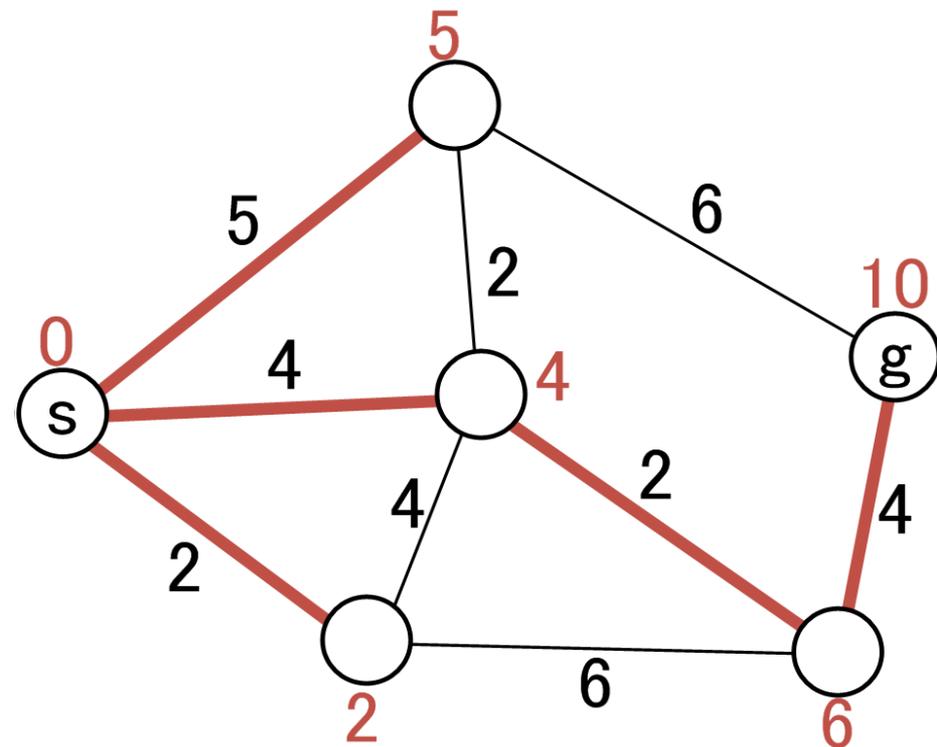
- ▶ 複数ある最短経路の中から最適なものを選択する方式
- ▶ 環境

アドホック ネットワーク	ノード数 電波到達範囲 ノード間距離 ルーティングプロトコル 無線規格	37 [台] 100 [m] (1hop) 95 [m] OLSR, PD-OLSR 802.11g
VoIPを想定した UDP通信	台数 選び方 通信タイプ トランスポートプロトコル パケットサイズ データ転送量	2台1ペア ランダム CBR UDP 200 [Byte] 64 [Kbps]

- ▶ 結果: 10回のシミュレーションでパケット到達率が平均6%改善

ダイクストラ法(Dijkstra's Algorithm)

- ▶ 最短経路問題を効率的に解く
グラフ理論におけるアルゴリズム
- ▶ スタートノードからゴールノード
までの最短経路とその距離を
求めることができる



最短経路に限らずホップ数を増やした経路を選択可能に

数字(黒字): 各地点間のメトリック

数字(赤字): その地点までのコスト

PD-OLSR

▶ 経路選択指標

- UDP:UDP Traffic
 - 自身が検出するネットワーク上のキャリアの総量
- TCP:TCP Session
 - キャリアとして検出するTCPセッション数と実際に自身が処理しているTCPセッション数の合計

▶ UDP通信用の経路

- 単純にUDP Trafficが最小の経路を選ぶ

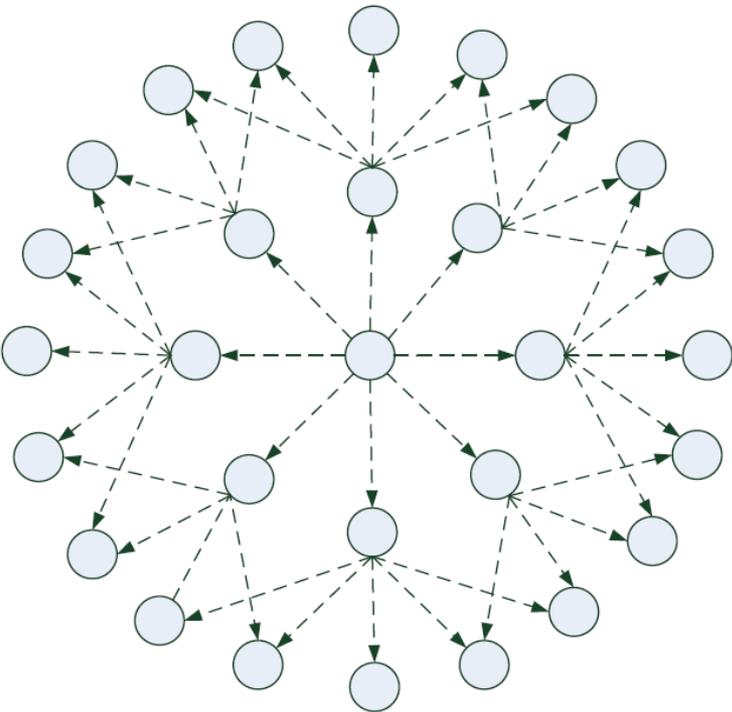
▶ TCP通信用の経路

- TCP の特性を活かして TCP スループットの公平性がとれる経路選ぶため, TCP Sessionが最小の経路を選ぶ

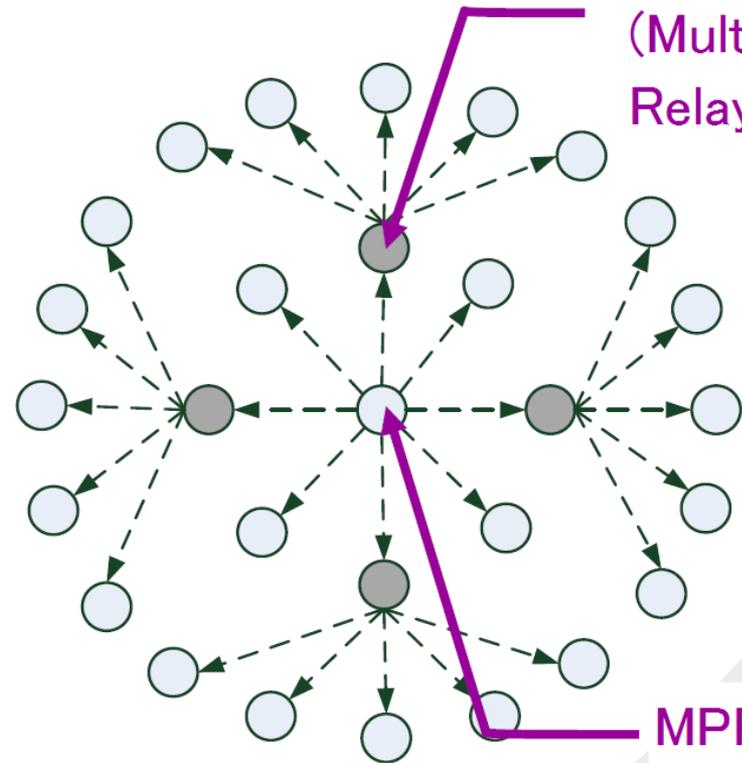
OLSRのフラッディング

MPR
(Multipoint
Relay)

MPRセレクタ



通常のフラッディング



OLSRのフラッディング

アドホックルーティングプロトコル

▶ プロアクティブ型

- 通信要求が発生する前からRTを生成
- ノードの移動が少なく、通信頻度の高いネットワークに適する
- 例: OLSR(Optimized Link State Routing)

▶ リアクティブ型

- 通信要求が発生した際にネットワーク内で経路を探索する
- ノードの移動が頻繁なネットワークに適する
- 例: AODV(Ad hoc On-Demand Distance Vector)

が生成できる。しかし、ノードの移動が少ない環境では、ピ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) [10] は DSR (Dynamic Source Routing Protocol) [4] 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。ETR は TCP スループットだけに着目しおり、UDP スループットは考慮していない。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。

IP ネットワークでは、フロー制御やウィンドウサイズを変化させる輻輳制御を行い、ネットワーク帯域を有効に使用しようとする TCP と、輻輳制御を行わず、端末が意図した通信がそのまま送出される UDP という異なる性質の通信が存在する。しかし、これらの性質の違いを考慮した経路選択方式は存在しない。そこで、本稿では、アドホックルーティングプロトコルの中でプロアクティブ型の代表的プロトコル OLSR (Optimized Link State Routing) を拡張することにより、ルーティングテーブル (以下 RT と記述) を TCP 用と UDP 用で別々に生成し、TCP と UDP の通信特性を生かした最適な経路選択を可能とするアドホックルーティングプロトコル PD-OLSR (Protocol Dependent-OLSR) を提案する。

以下、2章で OLSR を例にして既存技術の問題点を示し、3章では PD-OLSR の概要、4章でシミュレータ上での実装について示す。5章で動作検証を行い、6章でまとめを行う。

2. 既存技術

既存のルーティングプロトコルの動作を OLSR を例にして以下に示す。

2.1 OLSR 概要

OLSR は常時経路を生成しておく Proactive 型のルーティングプロトコルである。OLSR では各ノードが隣接ノードへ定期的にブロードキャストする HELLO と、ネットワーク全体へ定期的にフラッディングする TC という制御メッセージを送受信することにより、自身の存在をネットワークの全ノードに把握させる。HELLO と TC で送信される情報は、各メッセージの送信元ノードのアドレス、送信元ノードが把握している自身の隣接ノードのアドレス、情報の新しさを識別するシーケンス番号などである。これらの情報は RT を生成するために必要な情報であり、各ノード内の情報リポジトリに登録される。OLSR の RT 生成プロセスは HELLO と TC の受信により、情報リポジトリが更新されていくことによって進行する。

2.2 RT の生成プロセス

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

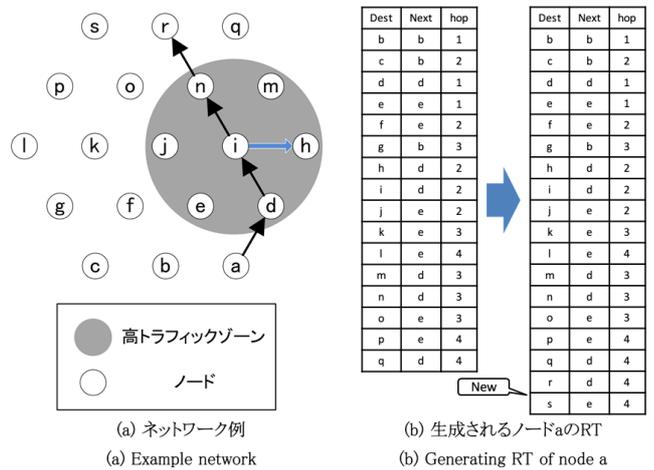


図1 OLSR の RT 生成
Fig. 1 Generating RT of OLSR

Dest に対して1つの経路を保持する。図1に、OLSR の RT 生成プロセスを示す。簡単化のためノードは規則的に配置されており、電波到達範囲は隣接ノードまでとする。図1(b)に示す RT は、ノード a を対象としてノード r および r への経路が新たに生成される様子を示している。制御メッセージの送受信によってノード r および s へのそれぞれ1つの最短経路が生成される。最短経路が複数存在する場合にどの経路が選択されるかは規定されていない。図1(a)ではノード a からノード s へ最短経路 [a → d → i → n → r] が生成された様子を示している。

2.3 UDP と TCP の特性の違い

TCP/IP ネットワークでは UDP と TCP という特性の異なる通信が存在する。UDP では端末が意図した流量のトラフィックがそのままネットワークへ送出され、ネットワーク内のパケットロスの影響が考慮されることはない。これに対し TCP では輻輳制御の機能により ACK が順調に返ってくるとウィンドウサイズを拡大し、帯域を有効に使おうとする。パケットロスを検出するとネットワークの輻輳が発生したものと判断し、ウィンドウサイズを縮小する。このようにウィンドウサイズが適切に調整され、ネットワークの更なる輻輳を防止する。TCP と UDP が混在したネットワークにおいては、ネットワーク上のトラフィックはまず送出された UDP パケットの合計により UDP が占めるトラフィックが定まり、残りの帯域を複数の TCP セッションが分け合う形になる。

図2に UDP について、図3に TCP についてのシミュレーションで求めたマルチホップ通信におけるホップ数とスループットの関係を示す。ノードを一直線上に並べ1~10ホップの通信でのスループットを測定した。UDP では、一般にネットワークに余裕がある限りホップ数増加によるスループット低下は見られない。これに対し、TCP では輻輳制御によってネットワーク帯域を使い切ろうとするため、ホップ数の増加とともにスループットが大きく低下する。

このように、UDP と TCP ではノードから送出されるトラフィック量およびマルチホップ通信時のホップ数によるスループット

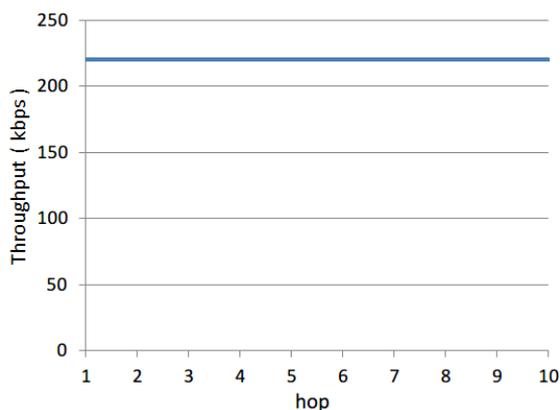


図2 マルチホップ通信におけるUDPスループット
Fig.2 UDP throughput in multi-hop communication

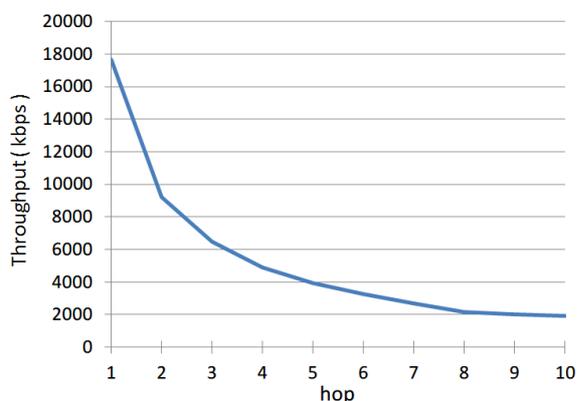


図3 マルチホップ通信におけるTCPスループット
Fig.3 TCP throughput in multi-hop communication

プットの変化が大きく異なる。

2.4 既存技術の課題

既存のルーティングプロトコルでは前述したOLSRの動作のように、ホップ数のみを基準に経路を生成するものが多い。OLSRでは、複数の最短経路が存在する場合、どの経路を選択するかという手順は定義されていないため、実際に生成される経路は、実装に任されており、多くの場合、最初に発見された最短経路が選択される。ここで、ノード*i*から*h*への通信がすでに行われていたものとする。ノード*i*から送信されるキャリアは隣接ノード*e*, *d*, *h*, *j*, *m*, *n*にも届く。そのため仮に図1(a)のような経路が生成されると、ノード*i*の周辺はトラフィックが増加し、スループットが低下する可能性がある。このようにOLSRではネットワークのトラフィックに偏りがあつた場合、最適な経路を生成することができない。

さらに、既存技術では特性の異なるUDPとTCPの通信を同一のRTを用いて制御を行っており、通信特性の違いを経路生成に反映することができない。

3. PD-OLSR

3.1 概要

提案方式であるPD-OLSRは、既存技術の課題として挙げたホップ数のみを基準にした経路選択、UDP/TCPで共通の

RTを用いている点に注目する。UDP/TCPのそれぞれで、通信状態を計測し、その情報をもとに経路を生成する。さらに、経路生成のアルゴリズムとしてダイクストラ法を用いることで、最短経路に限らず最適な経路を生成する。以下に、UDP/TCPそれぞれの経路選択指標、ダイクストラ法の適用、および経路生成手法について示す。

3.2 PD-OLSRの経路選択指標

PD-OLSRではOLSRの基本部分はそのま利用し、制御メッセージに各ノードが測定した自身の通信状態を表す情報をOLSRの制御メッセージに追加して送受信する。受信したノードはその情報を元にUDPとTCPそれぞれ専用のRTを生成する。そのため、UDPとTCPの経路選択に用いる指標を別々に考える。UDPの経路選択指標はUDPトラフィック(UDP Traffic)、TCPの経路選択指標はTCPセッション数(TCP Session)とする。UDPトラフィックとは各ノードが検出するネットワーク上の受信可能な電波(キャリア)の総量で、TCPセッション数は各ノードが検出するTCPセッション数の合計である。また、キャリアには、UDPおよびTCPの両方が含まれる。

3.3 ダイクストラ法の適用

PD-OLSRでは経路探索を行う際に、ダイクストラ法[11]を用いる。ダイクストラ法は、グラフ上の2頂点間の最短経路を効率的に求めるアルゴリズムである。経路の合計コストを基準に経路を求めることができ、カーナビゲーションシステムの経路探索や、鉄道の経路案内においても用いられている。

既存のOLSRでは、ネットワークトポロジの情報からホップ数を基準に最短経路を得ている。それに対し、ダイクストラ法を用い、経路コストとして各ノードが測定した通信状態を使用することにより、例えばUDPにおける総トラフィック量を基準に経路選択が可能となる。この方式によると、ホップ数が多いが混雑している部分を迂回した経路を選択することなどが可能である。

3.4 PD-OLSRの経路生成手法

3.4.1 生成される経路例

図4にPD-OLSRで生成される経路例、図5にRT生成の様子を示す。これらの図を用いてUDP用のRT生成を例にして、PD-OLSRの経路生成手順を示す。TCP用のRT生成についても、コスト計算方法は異なるが、UDP用RTの生成手順と同様の方法で実現できる。トラフィックの条件は図1の場合と同じである。図4(a)のテーブルは、各ノードが計測したUDP Trafficの情報(個数)である。ノード*i*からノード*h*への通信が行われているため、隣接ノードであるノード*d*, *e*, *j*, *m*, *n*ではUDP Trafficが検出されている。ここで検出されるトラフィックを仮に4として記載している。

3.4.2 経路生成時の動作

PD-OLSRでは、各ノードが計算した自身の通信状態を表すUDP Traffic情報をHELLOメッセージとTCメッセージに載せて隣接ノードへ広告する。各ノードは制御メッセージによって共有したネットワーク内のノードの隣接ノード情報とUDP Traffic情報からリンクメトリックに変換し、その情報を元にダ

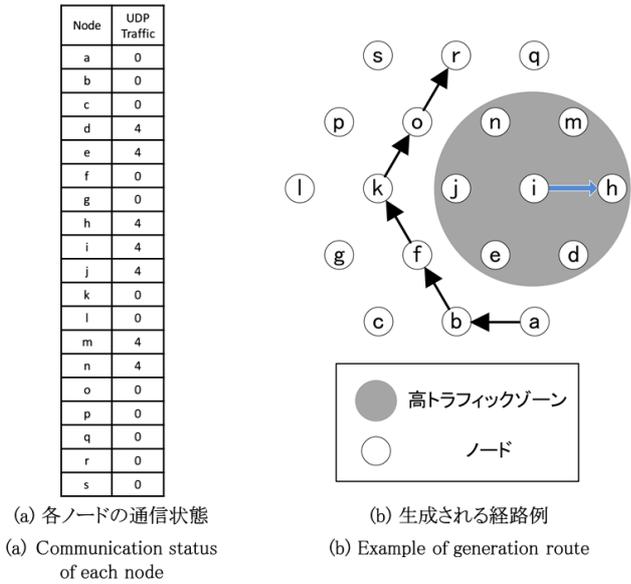


図 4 PD-OLSR で生成される経路例
Fig. 4 Example route generated in PD-OLSR

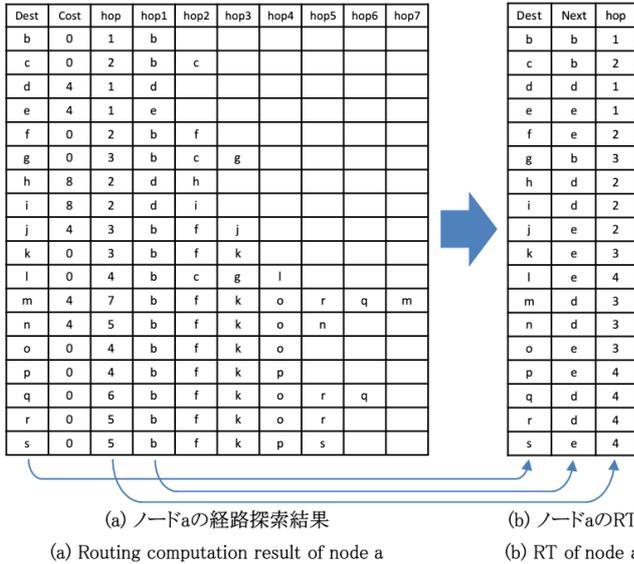
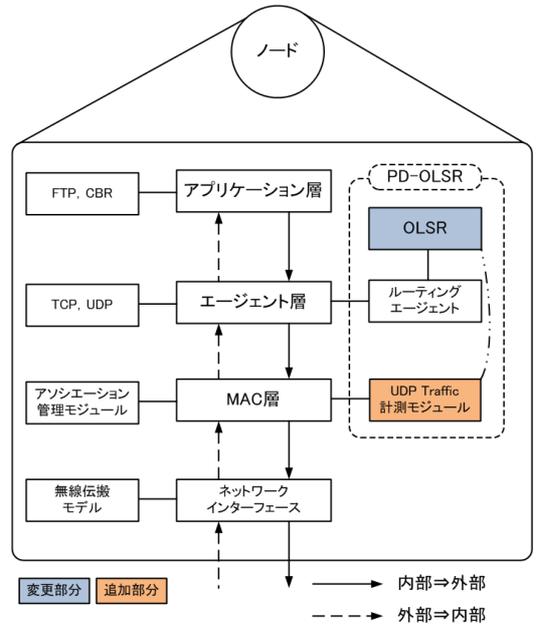
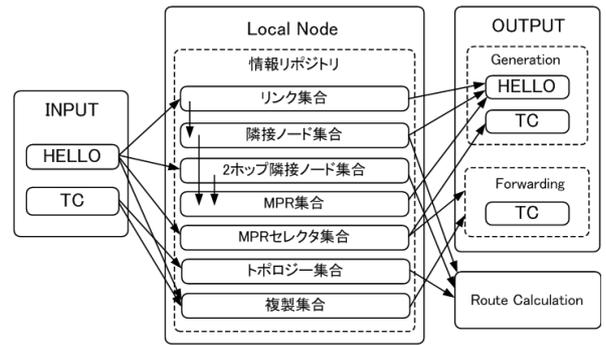


図 5 PD-OLSR の RT 生成
Fig. 5 Generation RT of PD-OLSR

イクストラ法によって各宛先に対して、複数存在する経路の中から最適な経路の探索を行う。経路探索結果は図 5(a) のようになり宛先ノード (Dest), 探索された経路のコスト (Cost), ホップ数 (hop) および経路中の中継ノード (hop1, hop2, ...) から構成される。ここで、リンクメトリックへの変換は、両端ノードの UDP Traffic の和をリンクメトリックとすることで行う。

PD-OLSR の RT は各宛先について探索された最適経路の次ホップおよびホップ数を、RT に追加していくことで生成する。

生成される経路は、各 Dest に対して経路の合計コストが最小のものである。例えば、図 4 でノード a からノード r へ UDP で通信が行われると、高トラフィックゾーンを避けた経路 [a → b → f → k → o → r] が生成される。



TCP 用の RT 生成については、上記の説明において、UDP Traffic を TCP Session 数をコストに置き換えたもので計算することで生成できる。TCP 用の RT 生成の場合、各 Dest に対して最小 TCP Session となる経路が選択される。もし、TCP Session が同じであった場合は、UDP Traffic の少ない経路が選択される。

4. シミュレータへの実装

PD-OLSR をネットワークシミュレータ ns-2[12] に実装した。以下に UDP 用の RT 生成機能を例にして、その概要を示す。

4.1 ns-2 の変更部分

図 6 に、ns-2 の内部構造と変更部分を示す。MAC 層に PD-OLSR の UDP Traffic を計測するモジュールを追加した。また、UDP Traffic 計測モジュールで計測した UDP Traffic をルーティングエージェントで呼び出せるようにし、ルーティングエージェントの OLSR を PD-OLSR の経路生成動作が行えるよう拡張した。

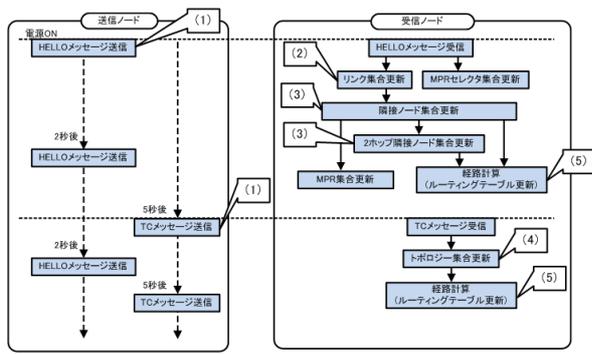


図 8 OLSR の改造箇所
Fig. 8 Alterations of OLSR

4.2 OLSR の拡張

OLSR において、制御メッセージと情報リポジトリには図 7 で示すような関係がある。HELLO メッセージを受信したノードはリンク集合、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元に、新しい HELLO メッセージ及び、TC メッセージを生成する。さらに、隣接ノード集合、2 ホップ隣接ノード集合、トポロジ集合の情報を元に RT を生成する。

OLSR の改造において、図 7 の情報リポジトリ内のリンク集合、隣接ノード集合、ホップ隣接ノード集合、トポロジ集合に通信状態情報である UDP Traffic の情報を追加した。OLSR の送信ノードと受信ノードにおける制御メッセージの処理の流れを図 8 に示す。以下に拡張したそれぞれの処理を示す。

- (1) 制御メッセージの送信
 - HELLO メッセージと TC メッセージに送信元ノード自身の UDP Traffic を付加
- (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通信状態情報を記録
 - 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生成
- (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通信状態情報を記録
- (4) トポロジ集合の更新
 - 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通信状態情報を記録
 - 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生成
- (5) 経路計算
 - 5 章に示す方法で RT を生成

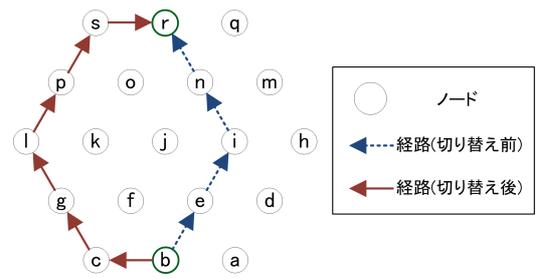


図 9 PD-OLSR での経路切り替え
Fig. 9 Switching route in PD-OLSR

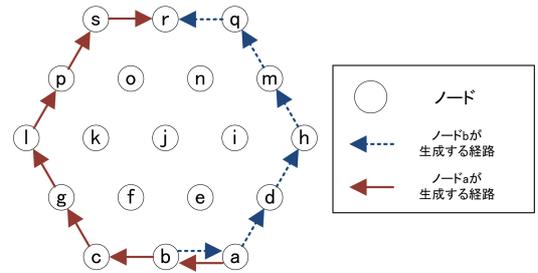


図 10 経路ループ発生時のノード a および b の経路
Fig. 10 Route at the time of occurrence a routing loop in node a and b

5. 動作検証

実装を行った UDP 通信用 RT の生成機能の動作検証を行った。以下に内容を示す。

5.1 検証内容

ノード 19 台を図 9 のように定期的に配置し、電波到達範囲を隣接ノードまでとしてアドホックネットワークを構築した。シミュレーション開始 30 秒後から 20 秒間ノード b からノード r へ UDP で通信を行い、制御メッセージによってノードが持つトポロジ情報が更新された際に通信経路がどのように切り替わるかを検証した。

5.2 検証結果

検証を行った結果、通信開始時には図 9 内の破線のような最短経路のうちの 1 つが選択され通信が行われた。この通信を行っている際、経路中のノードおよびその隣接ノードでは通信によるトラフィックが検出される。検出されたトラフィックが制御メッセージによって広告、共有されることでトラフィックが検出されるノードを避けた図 9 中の実線経路に切り替わる様子が確認できた。

しかし、さらにトポロジ情報が更新され経路が切り替わっていくと、経路ループの発生が確認できた。発生した経路ループは、制御メッセージによってトラフィックの情報が更新されるまで解消されず、発生時には通信が行えないためスループットが低下する。

経路ループ発生時の、経路ループが発生したノード a、b それぞれにおいてノード r への経路探索結果を図 10 に示す。ノード a では時計回り、ノード b では反時計回りに迂回するような経路が探索され、それらの経路の次ホップノードとしてノード a が b を、ノード b が a を登録したため経路ループが発生した

ことがわかる。

このように、探索される経路に差が出るのは経路探索時に保持しているネットワークトポロジの情報がノードごとに異なるためである。ネットワークトポロジ情報は受信したノードが制御メッセージによって更新される。しかし、制御メッセージは隣接ノードであっても同じものを受信するとは限らないため結果としてノードが保持する情報に差が生じる。

このような経路ループは各ノードが探索する経路が迂回する度合い、つまり最短経路に対してあまりホップ数を伸ばさないような経路を選択することで隣接ノードで正反対の方向に迂回するような経路探索を行わないようになり隣接ノード間における経路ループの発生を防ぐことができると考えている。

6. む す び

OLSR を拡張することにより、TCP/UDP において別々の RT を生成し、ノードの通信状況を考慮した経路選択が可能となるプロトコル PD-OLSR を提案した。RT を分けることでそれぞれの通信特性に合わせた経路選択を行う。通信状態指標をコストとし、ダイクストラ法によって経路探索を行うことで、最短経路に限らず最適な経路選択を行うことができる。今後は、PD-OLSR の実装を完了し、UDP と TCP が混在する環境や様々な環境におけるシミュレーションを行い PD-OLSR の効果を確認する。その中で、経路コストへのホップ数の反映方法を検討していく。

文 献

- [1] T. Clausen, Ed. : Optimized Link State Routing Protocol (OLSR), RFC 3626 (2003)
- [2] D. Johnson : The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4, RFC 4728 (2007)
- [3] C. Perkins : Ad hoc On-Demand Distance Vector (AODV) Routing, RFC 3561 (2003)
- [4] R. Ogier: Topology Dissemination Based on Reverse-Path Forwarding (TBRPF), RFC3684, IETF (2004).
- [5] Zygmunt J. Haas, Marc R. Pearlman, Prince Samar: The Zone Routing Protocol (ZRP) for Ad Hoc Networks, Internet draft, IETF MANET Working Group (2002). Expiration: January, 2003.
- [6] Charles E. Perkins, Pravin Bhagwat: Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers, ACM SIGCOMM , Vol. 24, No. 4 (1994).
- [7] V.Park, S.Corson: Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specication, Internet draft, IETF MANET Working Group (2001).
- [8] Douglas S. J. De Couto, Daniel Aguayo, Benjamin A. Chambers, Robert Morris: Performance of Multihop Wireless Networks: Shortest Path is Not Enough, ACM SIGCOMM
- [9] Toh, C.-K.: Associativity-Based Routing for Ad-Hoc Mobile Networks, Wireless Personal Communications , Vol. 4, No. 2, pp. 103-139 (1997).
- [10] 高橋ひとみ, 齊藤匡人, 間 博人, 戸辺義人, 徳田英幸: MANET における TCP スルー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, pp. 2857-2870 (2005).
- [11] Dijkstra, E.W. (1959). A note on two problems in connexion with graphs. In Numerische Mathematik, 1 (1959), S. 269 ~ 271.
- [12] The Network Simulator - ns-2: <http://www.isi.edu/nsnam/ns/>.