

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

高橋 ひとみ[†] 斉藤 匡人[†] 間 博人[†]
戸辺 義人^{††} 徳田 英幸^{†††}

無線端末の急速な普及にともない、動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) が注目を集めている。MANET を構築する経路制御プロトコルは多く提案されているが、ほとんどの経路制御は単純にホップ数を指標として最短経路を選択する。しかし実際の無線環境では使用される経路の電波品質によりスループットが変動するため、最小ホップ数の最短経路が最適な経路であるとは限らない。現在のインターネットにおいて TCP が多く使用されていることに着目し、MANET で効率の良いネットワーク運用が行われるためには、TCP スループットの向上する経路指標を用いることが必要である。本稿ではソースルーティングを行う MANET 上のプロトコルで動作する、パケット喪失率と遅延を指標とした経路制御機構である ETR (Estimated-TCP-Throughput Maximization based Routing) を提案する。本研究は既存研究では研究数が少ない実環境による評価を目的とし、ETR を MANET 経路制御プロトコルの 1 つである DSR (Dynamic Source Routing) に組み込み、実際に MANET 環境を構築し評価を行った。ETR を用いることで DSR 層におけるパケット喪失率が 0.5 の場合、本評価システムにおいて通常の DSR に比べ TCP スループットが最大で 1.9 倍向上した。本システムを組み込むことにより同じ宛先に対し複数経路が存在した場合、TCP スループットが良い経路の動的な切替えが可能となる。

Real Environment Evaluations of a Routing Scheme Based on Estimated TCP Throughput for MANET

HITOMI TAKAHASHI,[†] MASATO SAITO,[†] HIROTO AIDA,[†]
YOSHITO TOBE^{††} and HIDEYUKI TOKUDA^{†††}

We present a novel routing scheme, Estimated-TCP-Throughput Maximization based Routing (ETR) for mobile ad hoc networks (MANET). Many routing protocols for MANET have been proposed. Most of these protocols use the number of hops as a metric to select route. In MANET, nodes use wireless media to send a data. A performance of a route nodes use is influenced by radio link quality. Thus the shortest path does not always have the best performance under the bad link quality. To solve this problem, a source node should select the route with the maximum end-to-end TCP throughput because most of applications use TCP. ETR uses a first approximation maximum TCP throughput of a route as a new metric. This paper shows the design and implementation of ETR. Our goal is real environment evaluations for ETR. We add ETR to Dynamic Source Routing (DSR) protocol and evaluate ETR. In the case of 0.5 packet loss at DSR layer, ETR improves TCP throughput by 0.93 than unmodified DSR. ETR can select the path whose TCP throughput is maximum among the multiple paths to a destination.

1. はじめに

近年、無線通信技術の発達や無線端末の急速な普及にともない、無線端末どうしが瞬時に動的なネットワークを構築する MANET (モバイルアドホックネットワーク) に注目が集まっている。MANET は無線基地局を介した従来の無線ネットワークとは異なり、既存の施設や設備を必要とせずにネットワークを構築可能である。そのため災害地や僻地、戦場などの大規

[†] 慶應義塾大学大学院政策・メディア研究科
Graduate School of Media and Governance, Keio University

^{††} 東京電機大学工学部情報メディア学科
Department of Information Systems and Multimedia,
School of Engineering, Tokyo Denki University

^{†††} 慶應義塾大学環境情報学部
Faculty of Environmental Information, Keio University

模ネットワークへの利用から、屋内での小規模ネットワークへの適応も考えられており、MANET を構築する経路制御プロトコルも多く提案されている。

MANET における文献 1)~3) などの経路制御プロトコルはホップ数に基いて経路制御が行われ、複数経路が存在した場合は最短経路が選択される。また経路状態が調べられるのは経路探索時のみである。しかし、このような単純な経路制御では、MANET 環境において最適なネットワーク運用がなされるとは限らない。無線環境では有線環境と異なり、使用する経路の電波品質により通信性能が大幅に変動し、電波品質が悪ければ TCP スループットは低下する。すなわち最短経路であっても電波品質が悪いと、必ずしも最短経路が最適な経路とは限らない。また使用中の経路における電波品質の状態は端末の移動により大きく変動する。そのため、経路探索時に最善であった経路の電波品質が劣化し、他に良好な経路が現れたとしても、既存の経路制御では電波品質の劣化した最短経路が使われ続ける。

通常の有線ネットワークにおいて上記の問題を解決する手法として文献 4), 5) があげられる。これらの手法は各ホストが経路状態を測定し、それらの情報を代表ルータが収集および相互に交換することで柔軟な経路選択を実現している。しかし有線ネットワークと異なり、MANET ではすべての端末が分散しており、測定を行う固定端末を設けることは非効率的である。そのため上記の手法は MANET へそのまま適応できない。

無線ネットワークでは有線ネットワークとは異なり、電波品質が TCP スループットに大きく影響を与える。また電波品質は端末の移動により大きく変動し、それまでの電波品質が良好であっても、急激に電波品質が悪化することもありうる。これらより MANET の経路制御に必要な機能要件は以下となる。

- (1) 終端端末が経路情報を収集し、経路制御を行う。
- (2) リンクの状態に基づいて使用する経路が切り替えられる。
- (3) 使用する経路は他の経路と比較し最適な TCP スループットとなる。

これらの機能要件を満たす経路制御プロトコルとして、本稿では ETR (Estimated-TCP-Throughput Maximization based Routing) を提案する。ETR は新しい経路指標として TCP スループットの推定値を用い、分散的な端末で構成される MANET 上で効率の良い通信を実現する。各端末は遅延とパケット喪失率を測定、そして TCP スループットの推定値を算出

し、最適な TCP スループットを持つ経路への切替えを動的に行う。

また既存の MANET に関する論文は、シミュレータによる評価が大多数を占める。無線ネットワークの性能は、電波品質による影響を多大に受け、その電波品質は無線どうしの干渉、障害物による減衰などの多くの要素によって変動する。しかし、シミュレータは実環境に比べ電波品質に影響を与える要素が非常に簡便化されており、電波品質の変動も小さい。そのため、無線品質による影響を大きく受ける本機構の評価は、実用性を考慮すると現実的に即した実環境による評価を行うべきである。そこで本稿はシミュレータでなく、実際の MANET 環境を構築したテストベッド上で評価を行う。

本稿は、2 章で関連研究について述べ、3 章で既存の MANET を構築する経路制御プロトコルの問題を示す。その後、4 章で ETR の設計、5 章で実装について述べ、6 章で ETR の評価を行う。7 章において ETR の考察を行い、最後に 8 章でまとめと今後の課題について述べる。

2. 関連研究

本研究で提案する ETR と同じようにソースルーティングを行う MANET 上の経路制御として、Lin が提案する経路制御手法⁶⁾がある。この経路制御は経路選択の指標として TDMA 方式のフリースロット数を用いる。送信元は宛先までの経路を要求する際に、必要なフリースロット数とともに経路要求のパケットを送信する。もし要求したフリースロット数以上を確保可能な経路が存在した場合にのみ、送信元がフリースロットを予約し、データを送信する。しかし指標はフリースロット数のみに限定しているため、要求したスロット数以上の確保が可能であっても、電波品質が悪い場合、スループットは低下する。またフリースロット数が発見され、確保されるのはデータ送信時の最初のみであり、この方法では頻りに経路の電波品質が変化する MANET 環境において不十分である。なぜならデータ送信の初期段階でスループットが良好であっても、端末の移動により電波品質が悪化した場合、スループットは低下してしまうからである。

Toh が提案した ABR (Associativity-Based Long-lived Routing)⁷⁾では、経路選択の指標としてホップ数でなく「リンクの寿命」という概念を用いる。ABR において、各端末は一定間隔ごとに隣接端末へビーコンを送信する。もし隣接端末の移動が低速、もしくはない場合やビーコンを送信している端末と移動方向

が同一な場合、隣接端末が受信するビーコン数は多くなる。送信元が経路探索を行い経路探索の応答を中間ノードが受信した際、受信ビーコン数を付け加えて送信元へ転送する。送信元はそのビーコン数に基づき、持続性が高いと期待できる経路を選択できる。しかし、ノードの移動が少ない MANET 環境においては、選択経路が最短経路となり使用経路のスループットが必ずしも最善にならない。

一方、有線ネットワークにおいて、SPAND⁸⁾ は経路の状態を指標にし経路選択を行うルーティング機構である。各端末は受動的な計測を行い、経路の状態を示すパケット喪失率や帯域の情報を収集する。計測を行った端末は経路状態の情報をサーバへ送信し、送信元がデータ送信を行う際サーバに経路情報を問い合わせる。しかし自律分散的な MANET に SPAND の機構をそのまま適用することは難しい。

Anjali らは、有線ネットワークにおいて経路の利用可能帯域を予測し経路選択を行う経路制御手法⁹⁾ を提案している。この経路制御手法は SNMP と MRTG を用い過去の利用可能帯域からスループットの予測を行う。しかし MANET では端末の移動による経路変化が発生し、1 つの経路における過去の経路情報を収集、累積して使用することは非効率である。

Holland らが提案した Explicit Link Failure Notification (ELFN)¹⁰⁾ は本研究の目的と同じように、MANET において TCP スループットの向上を目的とした機構である。ELFN は TCP データの送信端末へリンクもしくは経路が途切れパケットロスが発生した情報を送信元へ伝達し、輻輳制御による輻輳ウィンドウの減少を発生させないよう動作する。ELFN は TCP スループットの向上という目的が ETR と同一であるが、目的達成の手法や TCP スループットが減衰する問題となるべき事象がまったく異なる。そのため同一ホスト上で、ETR と ELFN を同時に使用でき、単独でそれらの機構を使用するよりも TCP スループットが増加すると考えられる。

3. MANET における経路制御の問題点

前章で述べたように MANET を構築する経路制御の多くは同じ宛先に対し複数経路があった場合、ホップ数を指標にして最短経路が選択される。この手法はパケットの喪失が起きないと仮定した場合、ホップ数の少ない経路、つまり遅延の小さい経路が選択されることで TCP スループットにおいて最善な経路が選択可能であり、また実装も容易である。しかし TCP スループットは経路の遅延のみではなくパケット喪失率

によっても大きく変動する。MANET 環境では経路のパケット喪失率・遅延が、電波品質や端末の位置により変動することが予想される。そのため最短経路が TCP スループットにおいて最善な経路とは限らない¹¹⁾。また MANET の既存経路制御プロトコルでは、シミュレータによる評価が大部分を占めている。シミュレータ上では電波品質の大きい変動がないため、ホップ数を指標とした経路選択の手法に関してあまり問題視されていない。しかし実際の無線ネットワークは様々な要因で電波品質が変動するため、シミュレーション環境に比べ、経路選択の手法を重要視するべきである。

最短経路が TCP スループットにおいて必ずしも最善な経路ではないことを示すため、MANET を構築する経路制御プロトコルである DSR (Dynamic Source Routing)¹⁾ を用いて実環境において実験を行った。本章ではまず実験に用いた DSR の概要を述べ、次に実験について述べる。

3.1 DSR の概要

DSR は CMU の Monarch プロジェクト¹²⁾ で提案された MANET を構築する経路制御プロトコルである。DSR では、送信パケットのヘッダに送信元から宛先までの経路が含まれられ、ソースルーティングによりデータ配送が行われる。各端末には宛先への経路を保持する経路キャッシュが存在する。もし送信元の経路キャッシュに宛先までの経路が存在しない場合、送信元は経路要求のパケットをブロードキャストによりすべての端末に対して送信する。目的の宛先または、宛先までの経路を経路キャッシュに保持している端末が経路要求の応答を返し、明示的に経路が消滅したと分かるまでは経路キャッシュからその経路を削除しない。もし同じ宛先まで複数経路が存在した場合は、ホップ数の短い経路が選択される。

DSR のプロトコルスタック (DSR 層) は IP 層の一部として存在している。MAC 層で受信したパケットは IP 層に渡され、プロトコル番号により DSR 層に渡される。DSR 層に渡されたパケットはデータ転送のホップごとに処理が行われる。ある一定時間に DSR 層の ACK パケットによる応答がない、もしくは自分が送信したパケットが次のホップへ転送されていない場合にパケットの再送を行い、2 回の再送に失敗するとそのパケットを破棄する。また、データの再送間隔は各ノードが測定する隣接ノードへの RTT より推定 RTT が算出されており、推定 RTT を過ぎても転送先よりデータパケットが受信できない場合に再送が行

現在は Rice University へ拠点が移っている。

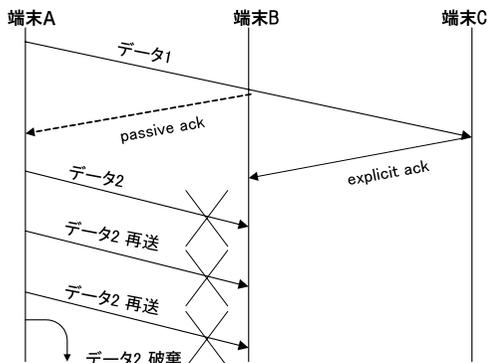


図 1 DSR のデータ送信手順
Fig. 1 Data sending process of DSR.

われる。

図 1 は宛先までの通信経路が決定した後のデータ送信手順を示したものである。端末 A は端末 B を経由し端末 C にデータを送信する。端末 A がデータ 1 を送信し、端末 B は端末 C へデータ 1 を転送する際、端末 A は端末 B が端末 C へ転送するデータを傍受しており、端末 B へ送信した同一の packets が転送されているかを判断する。それより、端末 A はデータ 1 が端末 B へ転送されていることが分かる (passive ack)。次に端末 C はデータ 1 を受信後、宛先である端末 B へ DSR 層における ACK (explicit ack) を送信する。これにより端末 B は端末 C へデータ 1 の転送が成功したと判断する。次に端末 A が端末 B へデータ 2 を送信する。このときデータ 2 は端末 B で受信できなかったとする。端末 A は端末 B からの passive ack が受信できないため、再送を 2 回行った後データ 2 を破棄する。

3.2 パケット喪失率と TCP スループットの関係

MANET 環境である無線ネットワークでは、通常の有線ネットワークと比較し非常にパケットロスが大きく、またロスの原因としてネットワークの輻輳によるものでなく、電波品質の劣化によるものが大部分を占める。有線ネットワークの状況と大きく異なる MANET 環境において最短経路が TCP スループットにおいて必ずしも最善な経路ではないことを示すため、DSR を用いて以下のような実験を行った。

MANET 環境は、FreeBSD3.3 Release+PAO および MANET を構築する経路制御プロトコルの一つである DSR を用いて構築した。実験内容として図 2 に示されるネットワークトポロジ上で、パケット喪失率と TCP スループットの関係、TCP シーケンス番号の推移を調べた。TCP スループットの測定には net-perf¹³⁾ を利用して 10 秒間の TCP データの送受信を

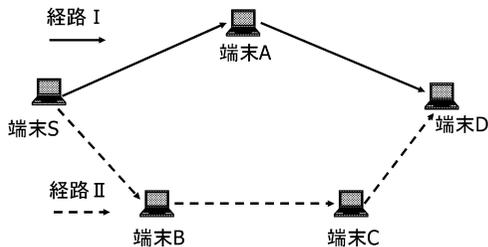


図 2 実験ネットワークのトポロジ構成図
Fig. 2 Network topology.

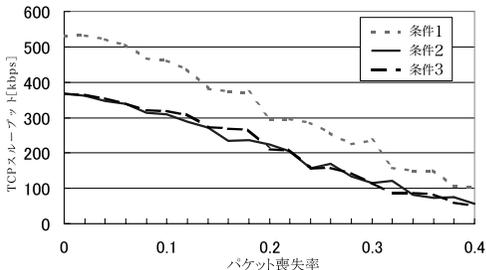


図 3 パケット喪失率と TCP スループット
Fig. 3 Packet loss ratio and TCP throughput.

10 回行い、TCP スループットの平均値を求めた。中間端末においてパケットをランダムに廃棄させるコードを加え、DSR 層でパケット喪失を発生させた。端末 S, A, D で構成される 2 ホップの経路 I, 端末 S, B, C, D で構成される 3 ホップの経路 II を構築し、以下の 3 通りの条件で TCP スループットを測定した。各中間端末の DSR 層におけるパケット喪失率を ρ とする。

- 条件 1 経路 I を用い、端末 A でパケットを廃棄する。ただし、 $0 \leq \rho \leq 0.4$ で変化させる。
- 条件 2 経路 II を用い、端末 B でパケットを廃棄する。ただし、 $0 \leq \rho \leq 0.4$ で変化させる。
- 条件 3 経路 II を用い、端末 C でパケットを廃棄する。ただし、 $0 \leq \rho \leq 0.4$ で変化させる。

この結果を図 3 に示す。ここで示されているパケット喪失率は終端端末間のパケット喪失率ではなく、パケットの廃棄を発生させた端末の DSR 層におけるパケット喪失率である。条件 1, 2, 3 より ρ が高くなるにつれ TCP スループットが低下することが分かる。また条件 1 と条件 2, 3 を比較すると ρ の値によっては、経路 I の TCP スループットよりも経路 II の TCP スループットの方が良い状況が発生することが分かる。たとえば端末 B, 端末 C において $\rho = 0$ 、端末 A において $\rho \geq 0.2$ であったならば、経路 I よりも経路 II を使用した方が TCP スループットは向上する。

次に経路 I と経路 II での TCP シーケンス番号の

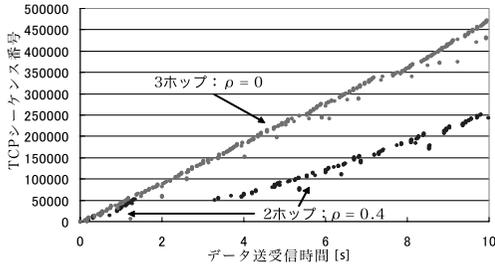


図 4 TCP シーケンス番号の推移
Fig. 4 Transfer of TCP sequence number.

推移を示す．先の実験と同様に netperf を用い TCP データの送受信を 10 秒間行った．経路 I の端末 A に $\rho = 0.4$ の，経路 II の端末 B において $\rho = 0$ のパケット喪失を発生させる．その結果を図 4 に示す．2 ホップの経路 I よりも 3 ホップの経路 II の方が傾きが大きい．このことから $\rho = 0.4$ である 2 ホップの経路よりも $\rho = 0$ である 3 ホップの経路の方が TCP スループットが良いことが分かる．これらの実験により，必ずしも最短経路が最善な経路ではないことが示された．

4. ETR の設計

ETR は同じ宛先に対し複数の経路が存在した場合，TCP スループットを予測し TCP スループットが良い経路に切り替える経路制御機構である．また ETR の想定環境として，MIT で行われている Roofnet¹⁴⁾ のように，ノードの移動が少ない環境であり，かつ屋内での小規模なネットワークを想定している．本章では ETR で用いる TCP スループットの指標，指標の算出方法，ETR のデータ送信手順について述べる．

4.1 TCP スループットの指標

同一の宛先に対し複数の経路が存在した場合，最も通信効率の良い経路が選択されるべきである．現在インターネットにおける IP トラフィックの 0.95 以上は TCP である¹⁵⁾．このことから MANET においても同様な状況が起こりうると想定し，本研究では TCP スループットに注目する．ETR は各経路の TCP スループットを推定し，通信効率が良いと期待できる経路を選択する．

前章の実験で，DSR 層におけるパケット喪失率と TCP スループットの関係を示した．ETR は TCP スループットの指標として具体的には TCP スループットのモデルを用い，最大スループットの推定値を計算し，TCP スループットが良い経路を選択する．一般的に知られている TCP スループットのモデルとして，M. Mathis らが提唱したモデル¹⁶⁾を基に，Floyd ら

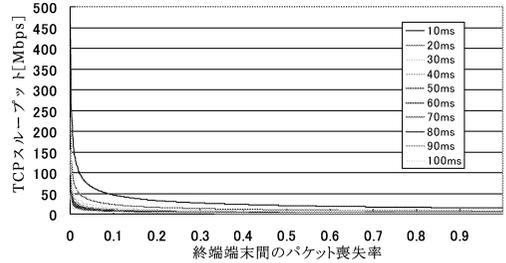


図 5 Floyd らのモデルによる TCP スループット
Fig. 5 Floyd's model based approximation of TCP throughput.

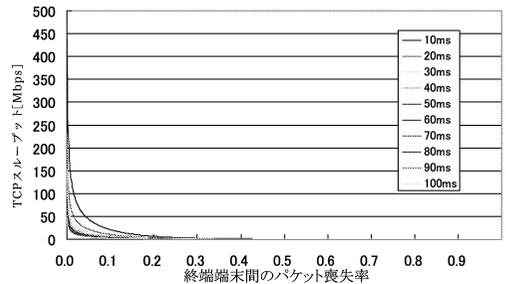


図 6 Padhye らのモデルによる TCP スループット
Fig. 6 Padhye's model based approximation of TCP throughput.

が変更を加えたモデル¹⁷⁾と Padhye らが提案したモデル¹⁸⁾が存在する．

Floyd らが提案したモデルを以下に示す．

$$T \leq \frac{1.5\sqrt{2/3} * B}{RTT * \sqrt{P}} \tag{1}$$

式 (1) は TCP の最大セグメントサイズ B ，経路における最小遅延 RTT ，終端端末間のパケット喪失率 P から TCP スループットの近似値 T を算出できる．次に Padhye らが提案したモデルを以下に示す．

$$T \approx \frac{1}{RTT\sqrt{\frac{2bP}{3}} + T_0 \min\left(1, 3\sqrt{\frac{3bP}{8}}\right)P(1+32P^2)} \tag{2}$$

式 (2) は経路における平均遅延 RTT ，終端端末間のパケット喪失率 P ，1 つの ACK によって確認できるパケット数 b ，タイムアウトが発生する時間 T_0 から TCP スループットの近似値 T を算出できる．

図 5，図 6 は式 (1) および (2) を用い，セグメントサイズを 1,500 バイト，遅延が 10ms ごとに 10ms から 100ms である場合の TCP スループットの推定値を算出したものである．MANET はローカルエリアネットワークの一種であり最大遅延は 100ms 以内であると想定し 100ms までの値を遅延として用いた．また図 6 において式 (2) のパラメータは通常の TCP

表 1 Floyd らのモデルによる TCP スループット

Table 1 Floyd's model based approximation of TCP throughput.

喪失率 \ 遅延	10 ms	20 ms	30 ms
0.01	146,979	73,485	48,990
0.02	103,923	51,962	34,641
0.03	84,853	42,426	28,284

表 2 Padhye らのモデルによる TCP スループット

Table 2 Padhye's model based approximation of TCP throughput.

喪失率 \ 遅延	10 ms	20 ms	30 ms
0.01	134,799	67,399	44,933
0.02	87,899	43,949	29,300
0.03	66,407	33,203	22,136

実装に従い b は 2, T_0 は $4RTT$ とした。

ETR は複数ある経路において TCP スループットが良好な経路を選択する機構である。そのため ETR では具体的な TCP スループットの値でなく、経路候補の相対的な TCP スループットの優劣が判明すればよい。相対的な TCP スループットの指標として図 5, 図 6 を比較した場合、グラフの形状が類似しており、結果も同様に類似したものとなる。規模の小さい MANET を想定した場合、今回実験に用いた DSR において、2 ホップ、3 ホップの遅延は、それぞれ 12 ミリ秒、16 ミリ秒となる。式 (1) および (2) で使用する TCP における終端端末間のパケット喪失率は、端末の移動がない 2 ホップの経路において 0.01 となった。また中間端末の DSR 層で $\rho = 0.4$ のパケット喪失を発生させた状態では 0.03 となる。この結果より、MANET で頻繁に起こりうる遅延 (10 ms, 20 ms, 30 ms) と終端端末間のパケット喪失率 (0.01 ~ 0.03) を考えた場合、両モデルの TCP スループットは以下の表 1, 表 2 となる。両モデルともに TCP スループットの優劣に関して、経路の持つパケット喪失率と遅延の組合せはほぼ同一である。これより使用するモデルの選択により、ETR の動作が大きく異なることは考えにくい。そこで ETR では経路指標として計算が簡単な式 (1) を用いる。

また、TCP スループットのモデル式はパケット喪失率 P が 0.05 以下の環境で有効とされている¹⁷⁾。前章の予備実験におけるパケット喪失率の ρ は DSR 層で人工的に発生させているパケット喪失率、つまり第 3 層のパケット喪失率である。DSR はパケット喪失が発生した場合、ホップごとに再送を行う機能を有しているため、TCP の第 4 層におけるパケット喪失率は非常に低くなる。2 ホップの経路において、第 4 層にお

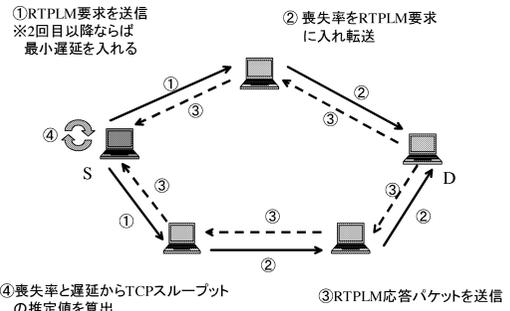


図 7 指標の算出手順

Fig. 7 Process of calculating metrics.

けるパケット喪失率を実際に測定した場合、 $\rho = 0$ ならば $P = 0.01$, $\rho = 0.4$ の場合においても $P = 0.03$ である。これより、本機構の想定環境では TCP 層におけるパケット喪失率が $P = 0.05$ 以上になる状態はあまり発生しないと考えられ、TCP のスループットモデル式は十分本機構に適用できると考えられる。また、本機構が TCP スループット推定式に用いるパケット喪失率は第 4 層のパケット喪失率である P を用いる。

4.2 指標の算出方法

ETR では、経路選択の指標である式 (1) を用いた TCP スループットの推定値を算出するために終端端末間の往復パケット喪失率 (RTPL: Round-Trip Packet Loss ratio) と遅延 (RTT: Round-Trip Time) が必要になる。また指標の取得に用いる経路は行き・帰りの経路が同一であることが条件である、そのため送信元は宛先までの全経路を把握しなければならない。そこで、既存の MANET を構築する経路制御プロトコルにおいて、ソースルーティングを行うプロトコル^{1),3),6),7),19),20)} が数多く提案されていることから、ETR では前提として送信元が宛先まで、ソースルーティングを行うものとする。

送信元が RTPL と RTT を取得し、TCP スループットの推定値を算出するまでの手順概要を図 7 に示す。図 7 では送信元の端末 S から宛先の端末 D への経路が 2 つ存在し、S から D へ TCP データを送信しているものとする。以下に指標の算出手順を記述する。

- [1] 各端末は起動時から現在までに転送したパケット数からリンクごとのパケット喪失率を計算する。S は D に対しデータ送信を始めると、RTT と RTPL の測定 (RTPLM: RTPL Measurement) を要求するパケット (RTPLM 要求) を一定間隔で送信する。
- [2] S から D の経路上にある中間端末は RTPLM

要求を受信すると、各リンクとのパケット喪失率を RTPLM 要求へ入れ、D 宛てに転送する。
 [3] D が RTPLM 要求を受信した際、各リンクのパケット喪失率から RTPL を算出し宛先へ応答するパケット (RTPLM 応答) に入れ S 宛てに送信する。

[4] RTPLM 応答を受信した S は RTPLM 応答から RTPL と、RTPLM 要求から応答までの時間より RTT を算出する。

経路の RTT と RTPL から式 (1) を用い、S から D への TCP スループットの推定値が算出される。

2 回目以降に S が同じ経路へ RTPLM 要求を送信する場合、S は経路の RTT を RTPLM 要求へ入れ D に送信する。これにより D も、D から S への TCP スループットの推定値を算出できる。

・各端末におけるパケット喪失率の測定方法

パケット喪失率の測定方法について述べる。各端末はタイムスロットで区切られた間隔ごとに転送したパケット数からパケット喪失率を測定する。もしタイムスロット中に十分なパケットが転送されなかった場合はその直前に測定したパケット喪失率を用いる。

各端末はそれぞれ隣接リンクでパケット喪失率を測定している。 $N_1, N_2, \dots, N_i, \dots, N_n$ の端末からなる経路において、 N_i から N_{i+1} に向かうリンクのパケット喪失率を $P_{i,i+1}$ とすると、 i 番目の端末である N_i は N_i から N_{i-1} へのパケット喪失率 $P_{i,i-1}$ と N_i から N_{i+1} へのパケット喪失率 $P_{i,i+1}$ を管理することとなる。

N_1 が RTPLM 要求を N_n 宛てへ送信し、中間端末である N_i が RTPLM 要求を受信した場合、 N_i は $P_{i,i+1}$ と $P_{i,i-1}$ を RTPLM 要求に入れ、 N_{i+1} に転送する。 N_n が RTPLM 要求を受信したとき、RTPLM 要求に含まれるパケット喪失率は $P_{1,2}, \dots, P_{n-1,n}, P_{n,n-1}, \dots, P_{2,1}$ となる。 N_n は式 (3) から算出した RTPL を RTPLM 応答へ格納し、 N_1 宛てに送信する。

$$RTPL = 1 - \prod_{i=1}^{n-1} (1 - P_{i,i+1})(1 - P_{i+1,i}) \quad (3)$$

もし送信元が RTPLM 要求を送信し、宛先からの RTPLM 応答がない場合、RTPLM 要求、応答が経路の途中で喪失したと考えられる。そこで式 (4) を用い RTPL を更新する。

$$\left. \begin{array}{l} \text{RTPLM 応答を受信した場合:} \\ \quad RT\hat{P}L \leftarrow \alpha r + (1 - \alpha)RTPL \\ \quad r \leftarrow RT\hat{P}L \\ \text{RTPLM 応答を受信しない場合:} \\ \quad r \leftarrow RT\hat{P}L \\ \quad RT\hat{P}L \leftarrow RT\hat{P}L + \beta \end{array} \right\} \quad (4)$$

式 (4) は無線環境において RTPL の変動が大きいと考えられるため、式 (3) によって算出した $RTPL$ を、平滑化係数 α を用いて平滑化を行い、 $RT\hat{P}L$ を算出する。そして平滑化を行い新しく算出された $RT\hat{P}L$ は、変数 r に格納される。また RTPLM 要求・応答が喪失し、RTPLM 応答が受信されなかった場合、直前まで算出していた $RT\hat{P}L$ を r へ格納し、 $RT\hat{P}L$ に加算定数である β を加算し RTPL を上げ、予測 RTPL を算出する。もし RTPLM 応答が受信不可能であった後に、新たな RTPLM 応答を受信した際、 $RT\hat{P}L$ の値を代入していた r より、それまで実測で算出していた RTPL に値が戻る。

・RTT の測定方法

RTT の測定方法について述べる。送信元は RTPLM 要求を送信する際、そのパケットヘタイムスタンプを付随させる。宛先は RTPLM 要求中のタイムスタンプを RTPLM 応答に入れ送信元へ送る。送信元が RTPLM 応答を受信した時間と、RTPLM 応答中のタイムスタンプの差を RTT とする。

・経路の切替え方法

MANET では端末の移動により、TCP スループットの推定値が激しく変動する状況が発生すると考えられる。そのような状況では、送信元において経路の切替えが頻繁に発生し、TCP スループットが低下してしまう。そこで切替えの閾値 h においてヒステリシスを用い、動的に h の値を変更する。現在使用中の経路における TCP スループットの推定値に比べ、他の経路における TCP スループットの推定値が h 以上の差があった場合、経路が切り替えられる。

また h は経路の状態によって最適な値となるよう、式 (5) を用い、RTPL の値によらず動的に変化する。式 (5) では、まず送信元が RTPLM 応答受信後に現在使用中の経路を変更した場合、閾値である h を定数 δ だけ上げる。また RTPLM 応答受信後、経路の切替えが発生しない場合、送信元は h を δ だけ下げる。ただし h が初期値の値である γ 以下となった場合、 h の値は γ となる。

$$\left. \begin{array}{l}
 \text{初期状態：} \\
 h \leftarrow \gamma \\
 \text{経路切替えが発生した場合：} \\
 h \leftarrow h + \delta \\
 \text{経路切替えが発生しない場合：} \\
 h \leftarrow h - \delta \\
 \text{if } h < \gamma \\
 h \leftarrow \gamma
 \end{array} \right\} (5)$$

4.3 端末のデータ送信

送信元が行う TCP データ送信の動作について詳細を述べる。送信元が初めてデータを宛先に送信する際、宛先までの経路探索を行う。一般的に経路探索を行う手法として、送信元はブロードキャストを行い経路探索の packets をすべての端末に送信する。そのため宛先は多くの経路探索 packets を受信し、最初に受信した経路探索 packets のみに応答する。しかし ETR は複数ある経路から TCP スループットの高い経路を選択する機構であるため、送信元は宛先に対して複数経路を所持しなければならない。そこで宛先は発見した最短経路のホップ数より、3 ホップまで長い経路候補に対して応答を行い、送信元に複数の経路候補を所持させる。

ETR はネットワークへの負荷を軽減させるため、送信元の経路表に存在する全経路に対し RTPLM 要求を送信するのではなく、実際に送信元がデータ送信を行う間のみ、データ送信に使用する宛先への経路とともに、他の経路候補に対し RTPLM 要求を送信する。そのため、送信元が宛先までの経路を発見後、初めて TCP データを送信する場合は RTT や RTPL が判明しておらず、経路選択の指標である TCP スループットの推定値を算出できない。この場合、送信元はまず最短経路を用いデータを送信する。ただしデータを送信する宛先の経路に対し、以前に算出した TCP スループットの推定値が存在し、かつその値が最近算出されたものであればそれを経路選択の指標として用いる。送信元がデータ送信を開始すると、通常データとともに RTPLM 要求が送信され、それぞれの経路に対する RTT, RTPL が判明する。送信元はそれらの値をもとに TCP スループットの推定値を算出し、経路の切替えを行う。

また送信元は RTPLM 要求の送信によって取得した情報より、TCP スループットの推定値を指標として最善な経路を選択し、経路を動的に切り替える。その際送信元がデータ送信中に経路を切り替えた場合、宛先で受信する packets の順序が入れ替わる可能性が

表 3 タイマで使用される変数
Table 3 Variables used in timer function.

変数名	説明
FLAG_DOWN_TIME	参照されない経路の FLAG を下げスループット推定値を削除する時間
PROBE_INTERVAL	RTPLM 要求を送信する時間
CALTIME	パケット喪失率を測定する時間

発生するが、その際 ETR では何もせず上位層の TCP に処理をまかせる。

5. ETR の実装

本章では、ETR の実装について述べる。ETR を Monarch プロジェクト¹²⁾ で公開されている DSR へ拡張する形で実装した。

表 3 は ETR の実装において使用されたタイマ変数の説明を示す。FLAG_DOWN_TIME は、経路参照が行われない場合に、送信元が RTPLM 要求の送信をとりやめ、TCP スループット推定値を削除する時間間隔を示している。PROBE_INTERVAL は送信元が RTPLM 要求を送信する時間間隔である。また、CALTIME は各端末がパケット喪失率を算出する時間間隔を示している。

また ETR は送信モジュール、転送モジュール、受信モジュールの 3 つに機能を分割できる。以下に各モジュールの詳細な説明を行う。

5.1 送信モジュール

送信モジュールはデータ送信元で動作し、以下の機能を提供する。

- 最善経路の選択
- RTPLM 要求送信
- RTPLM 応答受信

最善経路の選択は、送信元がデータを送信する際に使用する経路の選択を行う。まず、送信元は宛先までの経路を経路表から取得するため宛先のアドレスを引数にし、該当経路を経路表より取得する。もし宛先アドレスに対する経路候補が 1 つの場合はその経路が使用され、経路候補が複数存在する場合は経路に対する TCP スループット推定値があるかを調査する。もし経路候補に対する TCP スループット推定値が存在した場合、候補中で最大の TCP スループット推定値を持つ経路が使用経路として選択される。また TCP スループット推定値がない場合は、最短ホップの経路が選択される。その際、参照された宛先の経路へ RTPLM 要求の送信をするため RTPLM 要求送信機能へ参照された経路が通知される。また、FLAG_DOWN_TIME 秒間にその経路が参照されない場合、RTPLM 要求

の送信が必要なくなるため、RTPLM 要求送信の必要がなくなったことが RTPLM 要求送信機能へ通知される。

RTPLM 要求送信は PROBE_INTERVAL 秒間隔で RTPLM 要求を送信する。最善経路の選択機能より通知された経路が存在する場合、その経路に対して RTPLM 要求が送信されなければならない。そこで RTPLM 要求を送信すべき経路へ RTPLM 要求を送信する。

一方、RTPLM 応答受信は RTPLM 応答を受信した際の処理を行う。RTPLM 応答に格納されている RTPLM 要求の送信時刻と受信時刻の差より RTT を求める。その後 RTT と各リンクより算出された RTPL から TCP スループット推定値を求め、その値を対応する経路へ格納する。また更新されなくなった TCP スループット推定値を経路表より削除する関数をタイムに登録する。

5.2 転送モジュール

転送モジュールは中間ノードで動作し、以下の機能を提供する。

- パケット喪失率の測定
- RTPLM 要求の転送処理

パケット喪失率の測定は端末においてパケット喪失率の計測を行う。中間端末に限らず、MANET に参加するすべての端末は CALTIME 秒間ごとにパケット喪失率の測定を行い、各隣接リンクのパケット喪失率を線形リストで保持している。

RTPLM 要求の転送処理は転送すべき RTPLM 要求を受信した際、各隣接リンクへのパケット喪失率を RTPLM 要求へ格納する。

5.3 受信モジュール

受信モジュールは宛先で動作し、以下の機能を提供する。

- RTPLM 要求の受信処理

RTPLM 要求の受信処理は、宛先が受信した RTPLM 要求から各リンクのパケット喪失率より RTPL を算出し、RTPLM 応答を送信元へ返す。宛先は RTPLM 応答を送信後、RTPLM 要求より RTT およびその経路に対する TCP スループット推定値を算出し経路表へ格納する。

6. ETR の評価

本章では FreeBSD 4.2 Release 上で動作する DSR に ETR を実装したシステムと既存の DSR を使用して、評価を行った。シミュレータ上の評価では電波品質に影響を与える要素が大きく簡便化しており、電波

品質による転送性能の影響が少ない。そのため電波品質によって大きな影響がある本稿では、評価環境としてシミュレータではなく、実環境の MANET をテストベッドとして用いた。評価で使用した DSR は Monarch Project で公開されている実装をそのまま使用する。そのため、送信元による再送回数や再送間隔は通常の DSR と同一である。まず ETR における RTPLM 要求の送信間隔 (PROBE_INTERVAL) を変化させ、オーバーヘッドを評価し、次に両システムの TCP スループットを比較した。実験環境として屋内で通常の無線 LAN のように使用される小規模でかつノードの移動が少ない MANET を想定し、図 2 のような端末 S, A, D で構成される 2 ホップの経路 I, 端末 S, B, C, D で構成される 3 ホップの経路 II からなる小規模の MANET を実際に構築した。各端末のハードウェア仕様と使用した無線デバイスを以下に示す。

- 端末 S, D: Let's Note CF-B5R, CPU: Pentium III 600 MHz, メモリ: 64 MB
- 端末 A, B, C: Vaio PCG-505V, CPU: Petium 300 MHz, メモリ: 64 MB
- 無線デバイス: Melco WLI-PCM-L11 (802.11b)

6.1 オーバヘッドの評価

PROBE_INTERVAL を変化させ、RTPLM 要求・応答によるオーバーヘッドの評価を行った。実験ネットワーク上に経路 I, II を用いて TCP データを送信し、送信を始めてから 10 秒後に端末 A の DSR 層において意図的にパケット喪失を発生させた。またパケット喪失率は ETR により、経路の切替えが頻繁に発生しないよう比較的高い喪失率である $\rho = 0.4$ とした。

RTPLM 要求を送信する端末は S とし、PROBE_INTERVAL は 1, 2, 3, 4, 5 秒と変化させた。また ETR が TCP スループットの推定値を算出するために用いる RTPL の値は、DSR 層で送信したすべてのパケット数に対する DSR 層で 2 回の再送に失敗し破棄したパケット数の割合とした。

$$\eta = \frac{N_{RTPLM} + N_{Ack}}{N_{All}} \quad (6)$$

RTPLM 要求・応答によるオーバーヘッド (η) は式 (6) により算出される。 N_{RTPLM} は RTPLM 要求・応答のパケット数、 N_{Ack} は RTPLM 要求・応答に対する DSR 層での ACK パケット数、 N_{All} は DSR 層で送受信した全パケット数を示す。パケット数は各端末がパケットを送信した回数であり、複数ホップする場合は 1 回のホップごとに 1 パケットを送信したと想定して計算した。RTPLM 要求・応答によるオーバーヘッ

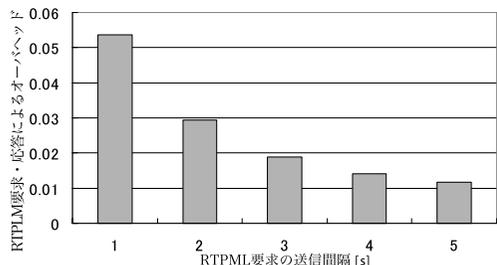


図 8 RTPLM 要求・応答のオーバーヘッド

Fig. 8 Overhead of RTPLM request and reply packets.

ドの結果を図 8 に示す．RTPLM 要求の送信間隔が長くなるにつれ，RTPLM 要求によるオーバーヘッドが少なくなる．RTPLM 要求の送信間隔が短ければ，経路状態の変化に対して送信元の感知が早くなるが，ネットワークへのオーバーヘッドが高くなるというトレードオフが存在することが分かる．

6.2 TCP スループットの評価

次に実験ネットワークへ以下の 2 条件で TCP データを送信し，TCP スループットの評価を行った．

条件 A 経路 I, II を用い，TCP 送信を開始し 20 秒後に端末 A において $\rho = 0.4$ のパケット喪失を発生させ続ける．

条件 B 経路 I, II を用い，TCP 送信を開始し 10 秒後に端末 A において $\rho = 0.1, 0.2, 0.3, 0.4, 0.5$ のパケット喪失を発生させ続ける．

本評価環境のような端末数が 10 台にも満たない小規模なネットワークの場合，DSR のみのルーティングパケットのオーバーヘッドは 0.005 以下であり，ルーティングパケット数が多いとされる AODV (Ad Hoc On Demand Distance Vector²⁾) に関しても 0.02 以下となる²¹⁾．そこで ETR において RTPLM 要求・応答のオーバーヘッドが 0.02 以下となるように，条件 A, B ともに PROBE_INTERVAL は 3 秒とした．

条件 A の結果を図 9 に示す．図 9 は通常の DSR (DSR) と，DSR に ETR を加えたシステム (ETR) の端末 S における TCP シーケンス番号の推移を比較したものである．

パケット喪失発生後，ETR では式 (1) より経路 I の方が経路 II に比べ TCP スループット下がると判断し，ETR の機能により経路 I から経路 II に切り替わった．そのため，パケット喪失が発生する経路 I を使い続ける DSR に比べ，ETR では経路状態が良好な経路 II へ切り替えることで TCP スループットが向上した．

条件 B の結果を図 10 に示す．図 10 は ETR と DSR の TCP スループットを比較したものである．30

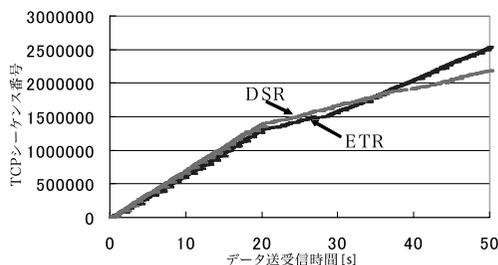


図 9 ETR, DSR の TCP シーケンス番号

Fig. 9 TCP sequence number of ETR, DSR.

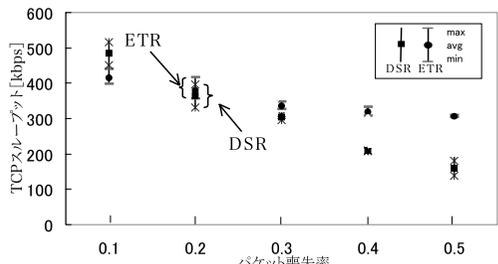


図 10 ETR, DSR の TCP スループット

Fig. 10 TCP throughput of ETR, DSR.

秒間のデータ送受信を行い，TCP スループットの最大値，平均値，最小値を示している．DSR ではパケット喪失が発生した後の TCP スループットを示し，ETR ではパケット喪失が発生し，経路が経路 I から経路 II へ切り替わった後の TCP スループットを示す．

$\rho = 0.1$ では，DSR, ETR とともに経路 I を使い続ける． $\rho = 0.2, 0.3, 0.4, 0.5$ では，DSR はそのまま経路 I を使用するが，ETR ではパケット喪失発生後経路 II を使用するようになった．DSR はリンクが切れた場合に経路要求を再び行い経路の再構築を図るが，今回の実験環境では経路の再構築の際に，経路 I を含む経路応答が受信されており，最短経路である経路 I を DSR が使い続ける結果となった． $\rho = 0.1$ のとき，ETR は RTPLM 要求の送信や処理のオーバーヘッドのため DSR に比べ 0.8 倍 TCP 転送性能が下がってしまう． $\rho \geq 0.2$ になると，ETR は DSR に比べて TCP スループットが向上する． ρ が高くなるにつれ DSR と ETR の TCP スループットの差は大きくなり， $\rho = 0.5$ のときには，TCP スループットが 1.9 倍に向上した．Couto らの論文¹¹⁾ より，実環境の MANET においてパケットロス率が 0.4~0.5 となる場合が多く発生していることから，ETR は DSR より優れた転送性能が実際の MANET において期待できる．

6.3 実環境における TCP スループットの評価

本節ではより現実的な MANET の実験環境を構築し，DSR および ETR の TCP スループットを評価し

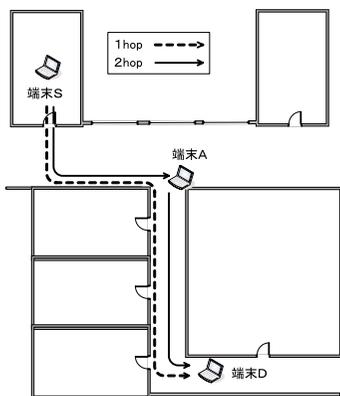


図 11 評価環境

Fig. 11 Measurement environment.

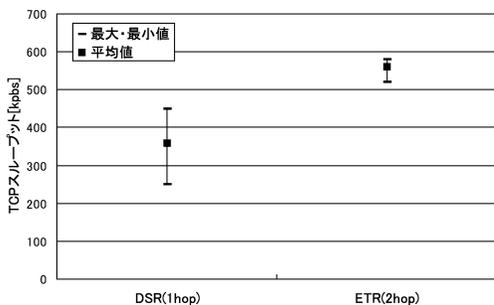


図 12 DSR, ETR の TCP スループット

Fig. 12 TCP throughput for DSR and ETR.

た．実験環境を図 11 に示す．まず，部屋内に存在する端末 S から端末 D へ TCP データの送信を行う．端末 S から端末 D への経路は直接接続できる 1 ホップの経路および，端末 A を経由する端末 S, A, D の 2 ホップの経路が存在する．本評価環境では前節の環境とは異なり意図的なパケット喪失を発生させず，またマックフィルタの使用を止め評価を行った．以下に評価に使用した端末および無線デバイスの詳細を示す．

- 端末 S, A, D : IBM ThinkPad T41p, CPU : Intel Pentium M 1.70 GHz, メモリ : 1 GB
- 無線デバイス : Melco WLI-PCM-L11 (802.11b)
 端末 S より端末 D へ 10 秒間の TCP データ送信を 10 回繰り返した．DSR は最短経路を選択するため 1 ホップの経路が使用される．ETR は TCP スループットが高いと予想される経路，つまり 2 ホップの経路を使用すると考えられる．また ETR の PROBE_INTERVAL は 3 秒とした．

その結果を図 12 に示す．図 12 におけるグラフの横軸は DSR, ETR を縦軸は TCP スループット (kbps) であり，グラフは DSR, ETR における TCP スループットの平均，最大，最小値を示している．1 ホップ

の経路を使い続ける DSR と異なり，ETR は 2 ホップの経路が 1 ホップの経路と比べ，TCP スループットが高くなると判断し，最短経路でない 2 ホップの経路を使用したと考えられる．その結果，TCP スループットは DSR と比べ平均 202 kbps ほど高い．これより実環境においても，ETR は TCP スループットの高い経路が選択できる．

7. 考 察

本章では ETR に関する考察を述べる．ETR は 4 章で述べられている想定環境において，効率良く動作し TCP スループットを向上させることができる機構である．つまり，

- 端末の移動が少なくトポロジの変化が少ない，
- ノード台数が少なくネットワークの規模が小さい，
- リンク間の電波品質の差が大きくパケット喪失率が高い，

● 送信データサイズが大きくセッション時間が長い，
 という条件下で ETR は効率的に動作する．トポロジの変化が少ないネットワークとは，メッシュネットワークのように各端末は一度設置されると頻繁に端末の移動が発生しないネットワークや，端末のモビリティモデルとして全端末が同じ方向に移動し，トポロジの変化が頻繁に発生しないネットワークとなる．実際に 6 章において行った想定環境と類似した評価環境では，ETR の優位性が顕著に現れている．6.3 節で行った，実環境に即した評価実験においても ETR の優位性が示しているが，上記の条件に加え室内のように壁などの障害物によって，各リンクにおける電波品質の差が大きい環境である方が，より ETR が効率的に動作できると考えられる．なぜなら，各リンクのパケット喪失率が大きく異なり最短経路が最善な経路でない確率が高いためである．また，本機構はネットワークへの負荷を軽減させるため，データ送信を行っていない経路に対するパケット喪失率，RTT の取得を行わず，データ送信時のみプローブパケットを送信し TCP スループットの推定値が判明できる．そのため，データサイズが大きくセッション時間が長いデータフローにおいて TCP スループット推定値が判明し，効率的な経路切替えが実現できる．

一方，ETR が効率的に動作しない環境として，ETR が効率的に動作するネットワークとは対極な性質を持つネットワークとなる．つまり，

- 端末の移動が多くトポロジの変化が頻繁に発生する，
- ノード台数が多くネットワークの規模が大きい，

- リンク間の電波品質の差が小さくパケット喪失率が低い、
- 送信データサイズが小さくセッション時間が短い、となる。端末の移動が多かつモビリティモデルがランダムウォークなどであり、トポロジの変化が頻繁に発生するネットワークでは使用経路が頻繁に途切れ、そのたびに送信元は経路要求を行う必要がある。経路切替えが発生した場合、パケット喪失の発生、パケットの重複や受信順序の入れ替わりが発生し、送信側のTCP輻輳制御が動作する。そのため、輻輳が起きていないにもかかわらずウィンドウサイズが小さくなり、TCPの転送性能が劣化する。ETRは宛先への経路に対するTCPスループット推定値が判明するまでは最短経路を用い、TCPスループット推定値が判明した後それを指標に経路を切り替える。そのため、移動による経路の切替えに加え、ETRにおいても経路の切替えが発生してしまい、TCPのウィンドウサイズが頻繁に小さくなるため非効率的な動作となる。次にネットワークの規模が大きい場合、送信元から宛先までのホップ数が増えRTTが増大し、ある宛先端末に対する経路候補も増える。ETRにおいてRTTが増大することによるデメリットは発生しない。しかし経路候補数が増える場合、ETRはパケットロス率、RTTを収集するため、定期的に各経路に対しプローブパケットを送信する。そのためETRによるプローブパケットが増え、ネットワークに対する負荷が増大する。最後に各リンクの電波品質の差が小さく、リンクにおけるパケット喪失率が低いネットワークでは、遅延の小さい最短経路がTCPスループットにおいて最適な経路になるため、ETRのTCPスループットに対する効果は少なくなってしまう。また、データが小さくセッション時間が短いデータフローでは、経路におけるTCPスループット推定値が判明する前にデータ送信が終了する可能性があり、効率的な動作は期待できない。しかし、データサイズが小さい場合はデータサイズが大きいデータフローに比べ、利用可能帯域の重要性は低くなる。

8. まとめと今後の課題

本稿では、MANET上でソースルーティングを行うプロトコルにおける、RTPLMとRTTを指標とした経路制御機構であるETRについて述べた。ETRは、経路状態の変化に対応して、最適なTCPスループットとなる経路を動的に選択する経路制御機構である。Monarchプロジェクトで公開されているDSRプロトコル上にETRを実装した。本機構は電波品質によ

る影響を大きく受けるためシミュレータ上ではなく、実機を用いて実際にMANETを構築し評価を行った。ETRにおけるRTPLM要求の送信間隔を変化させ、ネットワークへのオーバーヘッドを検証した。またTCPスループットの評価を行いETRを用いることでDSR層におけるパケット喪失率が0.5の場合、通常のDSRに比べTCPスループットが1.9倍向上することが判明した。

今後の課題として、以下のことがあげられる。

- ETRでは経路のRTPLMとRTTを測定する際、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ってしまい、より効率良く指標を取得する方法を提案し、RTPLM要求・応答によるネットワークへのオーバーヘッドを軽減させる。
- 今回、ETRをDSRに拡張する形で実装を行ったが、他の経路制御プロトコルにETRを適用させる。

謝辞 本研究は総務省「ユビキタスネットワーク制御・管理技術の研究開発(ubilaプロジェクト)」, 文部科学省「デジタルメディア・コンテンツ総合研究機構」の下に行われています。

参考文献

- 1) Broch, J., Johnson, D. and Maltz, D.: The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks, IETF Internet-Draft (2004). [Work in Progress].
- 2) Perkins, C.: Ad hoc On-Demand Distance Vector (AODV) Routing, RFC 3561 (2003).
- 3) Hass, Z. and Pearlman, M.: The Performance of Query Control Schemes for the Zone Routing Protocol, *Proc. ACM SIGCOMM'98*, pp.167-177 (1998).
- 4) Savage, S., Collins, A., Hoffman, E., Snell, J. and Anderson, T.: The End-to-End Effects of Internet Path Selection, *Proc. ACM SIGCOMM '99*, pp.289-299 (1999).
- 5) Andersen, D., Balakrishnan, H., Kaashoek, M. and Morris, R.: Resilient Overlay Networks, *Proc. ACM SOSP'01*, pp.131-145 (2001).
- 6) Lin, C.: An On-demand QoS Routing Protocol for Mobile Ad Hoc Networks, *Proc. IEEE INFOCOM'01*, pp.1735-1744 (2001).
- 7) Toh, C.-K.: Associativity-Based Routing for Ad-Hoc Mobile Networks, *Journal on Wireless Personal Communications*, Vol.4, No.2, pp.1-36 (1997).
- 8) Seshan, S., Stemm, M. and Katz, R.H.: SPAND: Shared Passive Network Performance

- Discovery, *USENIX Symposium on Internet Technologies and Systems*, pp.135–146 (1997).
- 9) Anjali, T., de Oliveria, J., Chen, L. and Smith, J.: A New Path Selection Algorithm for MPLS Networks Based on Available Bandwidth Estimation, *Proc. Qofls 2002*, pp.205–214 (2002).
 - 10) Holland, G. and Vaidya, N.: Analysis of TCP Performance over Mobile Ad Hoc Networks, *Proc. ACM MOBICOM'99*, pp.219–230 (1999).
 - 11) De Couto, D.S.J., Aguayo, D., Chambers, B.A. and Morris, R.: Performance of Multi-hop Wireless Networks: Shortest Path is Not Enough, *Proc. 1st Workshop on Hot Topics in Networks (HotNets-I)*, pp.83–88 (2002).
 - 12) The MONARCH Project at Carnegie Mellon University. <http://www.monarch.cs.cmu.edu/>
 - 13) Netperf: <http://www.netperf.org/>
 - 14) MIT Roofnet: <http://www.pdos.lcs.mit.edu/roofnet/>
 - 15) 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).
 - 16) Ott, T., Kemperman, J. and Mathis, M.: The Stationary Behavior of Ideal TCP Congestion Avoidance (1996).
 - 17) Floyd, S. and Fall, K.: Promoting the Use of End-to-End Congestion Control in the Internet, *IEEE/ACM Trans. Networking*, Vol.7, No.4, pp.458–472 (1999).
 - 18) Padhye, J., Firoiu, V., Towsley, D. and Kurose, J.: Modeling TCP Throughput: A Simple Model and its Empirical Validation, *Proc. ACM SIGCOMM'98*, pp.303–314 (1998).
 - 19) Park, V. and Corson, M.: A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks, *Proc. IEEE INFOCOM'97*, pp.1405–1413 (1997).
 - 20) Ko, Y. and Vaidya, N.: Location-Aided Routing (LAR) in Mobile Ad Hoc Networks, *Proc. ACM MOBICOM'98*, pp.66–75 (1998).
 - 21) Das, S., Perkins, C. and Royer, E.: Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks, *Proc. IEEE INFOCOM'00*, pp.3–12 (2000).

(平成 17 年 3 月 31 日受付)

(平成 17 年 10 月 11 日採録)



高橋ひとみ

2003 年慶應義塾大学環境情報学部卒業。2005 年同大学大学院政策・メディア研究科修士課程修了。現在、同大学院政策・メディア研究科後期博士課程に在学中。無線ネットワーク、モバイルアドホックネットワークの研究に従事。



斉藤 匡人 (学生会員)

2004 年慶應義塾大学大学院政策・メディア研究科修士課程 (Information Technology) 修了。ユビキタスアドホックネットワークにおけるセキュリティ基盤技術、ネットワーク情報の三次元視覚化等の研究に従事。現在、慶應義塾大学大学院政策・メディア研究科博士課程在籍、本学術振興会特別研究員 (DC2)。ACM, 電子情報通信学会各学生会員。



間 博人 (学生会員)

現在慶應義塾大学大学院政策・メディア研究科後期博士課程在学中。センサネットワーク、無線通信、通信プロトコルの研究に従事。IEEE 会員。



戸辺 義人 (正会員)

1984 年東京大学工学部電気工学科卒業。1986 年同大学院修士課程修了。同年株式会社東芝入社。産業用ネットワークの開発に従事。1997～2002 年慶應義塾大学にて研究員および特別研究助教授。2000 年博士 (政策・メディア)。2002 年から東京電機大学。現在、同大学教授。ユビキタスコンピューティング、センサネットワークの研究に従事。IEEE, ACM, 電子情報通信学会, 計測自動制御学会各会員。



徳田 英幸（正会員）

慶應義塾大学より工学修士・カナダ, ウォータールー大学より Ph.D. (Computer Science). 現在, 慶應義塾大学大学院政策メディア・研究科委員長, 同大学環境情報学部教授.

分散リアルタイムシステム, マルチメディアシステム, 通信プロトコル, 超並列・超分散システム, モバイルシステム等の研究に従事. IEEE, ACM, 日本ソフトウェア科学会各会員.
