

Mobile IP ネットワークにおける移動端末のデータ受信を対象とする、ハンドオフと伝送誤りの双方を考慮した TCP 高速化方式の提案と評価

海老原 成[†] 加藤 聰彦[†] 伊藤 秀一[†]

近年、携帯端末を用いたモバイルネットワーキングが急速に普及しており、高速に移動している状況でも無線インターネットを介して各種情報を受信・検索することが可能となっている。しかし、無線環境において TCP を利用した場合、無線リンクでの伝送誤りや、端末の移動にともなうハンドオフによる通信断に起因してパケット紛失が頻繁に発生することで、不必要な輻輳制御が多発し、スループットが低下することが指摘されている。このため、ハンドオフや伝送誤りを想定した、移動端末の TCP によるデータ受信を高速化するための研究が広く行われている。しかし、これらの研究の多くは、無線リンクでの伝送誤りまたはハンドオフのどちらか一方のみに主眼がおかれている、Mobile IP が基本プロトコルであるような IP レベルの可動性を提供するモバイルネットワークに対応できないなど、いくつかの問題がある。これに対して本稿では Mobile IP に基づくネットワークにおいて MN がデータを受信する場合、ハンドオフおよび伝送誤り双方に起因するパケット紛失に対して TCP のスループットを低下させない手法を提案する。また、ソフトウェアシミュレータにより提案手法の性能評価を行った結果について示す。

Proposal and Evaluation of High Speed TCP Communication Procedure for Data Receive by Mobile Nodes over Mobile IP Network Taking Account of Both Handoff and Transmission Errors

JOE EBIHARA,[†] TOSHIHIKO KATO[†] and SHUICHI ITOH[†]

Recently, the mobile communication environment is widely spread and we can receive various information through the Internet even when we move at high-speed. But, in such an environment, it is pointed that the packet losses caused by transmission errors in wireless links and handoff during terminal moving degrade the TCP communication throughput, because many unnecessary congestion controls cause. There are a lot of researches for this problem, but most of them focus on only either transmission errors in wireless links or handoff, and do not take account of the coordination with Mobile IP procedures which give IP level mobility support. In this paper, we propose a high speed TCP communication procedure over Mobile IP taking account of both handoff and transmission errors when mobile node receives data. And we also show the results of performance evaluation of our procedure examined by software simulator.

1. はじめに

近年、携帯端末を用いたモバイルネットワーキングが急速に普及しており、高速に移動している状況でも無線インターネットを介して各種情報を受信・検索することが可能となっている。しかし、無線環境において TCP を利用した場合、無線リンクでの伝送誤りや、端末の移動にともなうハンドオフによる通信断に起因してパケット紛失が頻繁に発生し、不必要な輻輳制御

が多発しスループットが低下することが指摘されている。このため、ハンドオフや伝送誤りを想定した、移動端末 MN (Mobile Node) の TCP によるデータ受信を高速化するための研究が広く行われている¹⁾。しかし、これらの研究を現実のモバイル環境に適用するにはいくつかの問題がある。

ハンドオフに対応する研究としては、Freeze TCP²⁾ と M-TCP³⁾ が代表的である。これらは、移動を検出した場合に、ウィンドウサイズを 0 とした新規の Ack セグメント (New Ack) を用いて、データの送信元をパーシストモードにするというアプローチを基本としている。Freeze TCP では、MN が無線の電波強度の低

[†] 電気通信大学大学院情報システム学研究科
Graduate School of Information Systems, University of
Electro-Communications

下によりハンドオフの発生を事前に予測すると、ウィンドウサイズ 0 の Ack をデータ送信元を送ることでデータ送信元をパーシストモードにし、不必要な輻輳制御の起動を防ぐ。また、M-TCP では、Supervisor Host (SH) と呼ばれる制御ノードに、データ送信元に中継する Ack の 1 バイトの保留と、その 1 バイトを使ったウィンドウサイズ 0 の Ack によるパーシストモード設定を行わせる方式である。SH は、MN と自分自身の間で発生したパケット紛失によりハンドオフの発生を推定し、ウィンドウサイズ 0 の Ack を送信する。しかしこれらの方法では、無線区間の電波状況が悪く伝送誤りが多発するような場合を想定しておらず、そのような状況では、ハンドオフを行っていないにもかかわらずウィンドウサイズ 0 の Ack を送信し、スループットを低下させると考えられる。また双方とも、Mobile IP⁴⁾ により IP レベルの可動性を提供するモバイルネットワークにおける移動管理の手順との協調については明示的には示されていない。

また、無線区間の伝送誤りへの対応に主眼を置き、TCP のスループットを改善する研究も多く行われている⁵⁾⁻¹¹⁾。これらは、WTCP⁵⁾、TCP Westwood⁶⁾、TCP VenO⁷⁾ のように TCP 自身を機能拡張するもの、I-TCP⁸⁾ のように無線基地局を中間にして TCP セッションを 2 つの部分に分割するもの、Snoop⁹⁾ のように無線基地局が TCP の振舞いを監視し必要に応じ再送処理を行うもの、IEEE802.11¹⁰⁾ や第三代携帯電話網¹¹⁾ のように無線区間のデータリンクレベルで再送制御を行うものに大別される。しかし、いずれの方法においても MN の移動によるハンドオフに起因するパケット紛失には対応できない。

ハンドオフと無線区間の伝送誤り双方を考慮し、TCP のスループットを改善する研究も行われている。New Snoop¹²⁾ では Snoop に、ハンドオフに対応する機能を追加するために、Snoop モジュールを階層的に配置している。無線基地局に無線区間の伝送誤り対策用の Snoop モジュールを、さらに上流にハンドオフ対策用の Snoop モジュールを配置し、ハンドオフを回避する。しかし、この方式は、3GPP ネットワーク上で用いられることを想定しており、3GPP の移動検出機能に依存しているとともに、特定の Mobile Switching Center を経由することを前提としている。このため、M-TCP と同様な理由で Mobile IP を用いるネットワークでは適用することはできない。

これらに対して、本稿では、Mobile IP に基づくネットワークにおいて、ハンドオフおよび伝送誤りに起因するパケット紛失の双方に対して、TCP のスループット

を低下させない方式を提案する。本稿では以下のような新たな提案を行う。

- Mobile IP の FA (Foreign Agent) と HA (Home Agent) に対して、ハンドオフに対応する機能を実現させる。M-TCP の SH の機能を FA に分散させ、かつ移動直後の Mobile IP の移動登録手順において HA と FA が協調して、すばやいデータ転送の再開を可能とする手順を示す¹³⁾。
- 伝送誤りに対しては、FA に無線区間でのパケット紛失に対する再送機能を実現させる。ハンドオフと伝送誤りの双方に対応するために、FA が MN へ転送したデータパケットの紛失を検出した場合は、両方の可能性を考慮し、FA から MN へのデータの再送と、CN (Correspondent Node) へのウィンドウサイズ 0 の Ack の送信の双方を行う。その後、MN からのパケットの受信、Mobile IP の移動登録による移動の通知、一定時間以上の通信断などにより通信状況が判定された時点で対応する処理を行う方式を示す。
- MN へのパケット再送に対しては、Sack (Selective Acknowledgment)¹⁴⁾ による効率的な再送を行う。Sack を用いた再送については文献 15) においても言及されているが、本稿では MN がデータを受信する場合について、Sack を用いて必要最低限のデータセグメントを再送するための手順を示している。

本稿では、提案する方式の詳細な手順を示すとともに、ソフトウェアシミュレータによりその性能評価を行った結果について示す。以下、2 章で本方式の詳細手順を示し、次に 3 章で本方式を用いた場合の通信シーケンス例を示す。さらに 4 章で本方式をソフトウェアシミュレータにより性能評価した結果を示し、5 章で結論を述べる。

2. プロトコルの詳細手順

2.1 設計方針

提案方式の設計にあたり、以下のような設計方針を立てた。

1. Mobile IPv4 に従って移動管理された MN が、インターネットに接続された固定ホストである CN からデータを受信する場合を想定する。MN がある FA を離れ別の FA の下に移動し HA へ移動登録を行うまでの通信断と、FA と MN の間の無線リンクにおける伝送誤りによるパケット紛失に起因して起動される輻輳制御によって生ずるスループット低下を防ぐことを目的とする。このため、

FA と HA には TCP に関する処理を実行する機能を持たせる。また、一般のノードである CN の変更は行わず、MN に対しては下記のように Sack をサポートするという要件のみを課し、それ以外の変更は行わないこととする。

2. FA は、CN から MN へのデータセグメントとそれに対する MN からの Ack セグメントを TCP コネクションごとに監視する。FA は CN からのデータセグメントを MN が受信応答するまで保持し、FA と MN の間で伝送誤りによるパケット紛失が発生したと判断した場合には、そのデータセグメントを再送する。また FA は MN が自分の配下から移動したと判断した場合は、CN に対してウィンドウサイズ 0 の Ack を送信し、CN をパシストモードとする。また MN が他の FA の下に移動すると、新たな FA、HA、それまでの FA が協調し、CN に対してウィンドウを開く Ack を送信しデータ転送を再開させる。
3. FA は、以下の方法でパケット紛失の検出を行い、再送する。
 - 無線 LAN や携帯電話網で使用されている多くの無線リンクにおいては、フレームの順序逆転は生じないため^{10),11)}、MN からの Sack オプションにより、FA が転送したにもかかわらず MN が受信していないデータセグメントを判定し、速やかな再送を行う。
 - TCP コネクションごとに再送タイマを設け、通常の TCP と同様に、RTT の推定と再送タイムアウトの検出を行う。再送タイマの計算は、FA が転送するデータセグメントと受信する Ack セグメントに対して、TCP の再送タイムアウトの計算を行う¹⁶⁾ (ただし 1 秒以下の値の丸め込みは行わない)。再送タイマがタイムアウトすると対応するセグメントを再送する。ここで、再送タイマは小さなインターバルタイマを用いて正確に計測可能であるとするとする。

なお、Sack オプションによる再送を行っているため、FA は重複 Ack による再送は行わない。また、FA と MN 間の無線リンクにおける紛失に起因する重複 Ack は、データを含まない限り CN へは転送しない。逆にインターネットで発生したパケット紛失に対する重複 Ack は CN へ転送する。
4. MN のハンドオフを検出するために、FA は MN ごとに移動検出タイマを用意する。FA は、ある MN に対して (移動検出タイマが起動していない

場合に) データセグメントを転送すると、そのタイマをスタートさせ、MN から何らかのセグメントを受信すると、そのタイマをリスタートさせる。また FA は、通常 MN からの Ack に対して確認応答シーケンス番号を 1 バイト分少なく CN に返送し、移動検出タイマがタイムアウトすると、その 1 バイトを用いてウィンドウ 0 の New Ack を送出する。なおこの時点でインターネットにおいてパケット紛失が発生した場合は、輻輳制御を起動するため、ウィンドウ 0 の Ack の送信は行わない。移動検出タイマのタイムアウト値は、FA で計測した TCP の再送タイムアウト値を用いる。MN が別の FA の下に移動し HA に登録する時点で、HA は、移動元の FA から MN の行っている TCP 通信の情報を入手し、移動先の FA に通知するとともに、通信中の CN に対して通信を再開させる。このために Registration Request/Reply に用いる新たな Extension と、HA と移動前の FA との間での新たなメッセージを導入する¹³⁾。

2.2 詳細設計

2.2.1 FA におけるハンドオフと伝送誤りに対応するための手順

本項では、MN の移動にともなうハンドオフに起因する通信断および MN と FA 間の無線リンクにおける伝送誤りの双方を検出し、CN へのウィンドウ 0 の Ack 送信および紛失したデータの再送を行うための手順について示す。このために FA は、以下のようなデータ構造を用いて、TCP コネクションを管理する。

- TCP コネクションテーブル

MN と CN の IP アドレス・ポート番号で識別される個々の TCP のコネクションに対して、制御用状態 (non-persist 状態か persist 状態か)、再送リストへのポインタ、MN から通知された確認応答シーケンス番号・ウィンドウサイズ、CN に通知したシーケンス番号・ウィンドウサイズ、RTT 計測や再送タイマのための管理情報、移動検出タイマテーブルへのポインタ、インターネットでの輻輳の発生を示すフラグ (輻輳フラグ) の各情報を保持する。

- 再送リスト

MN へ転送したデータセグメントに対して、シーケンス番号、長さ、Sack されたことを示すフラグ (Sack フラグ) の各情報を保持する。データセグメントごとの情報は MN に転送または再送された古い順に並べられる。

- 移動検出タイマ管理リスト

MN が FA の配下から移動していないかどうかを検

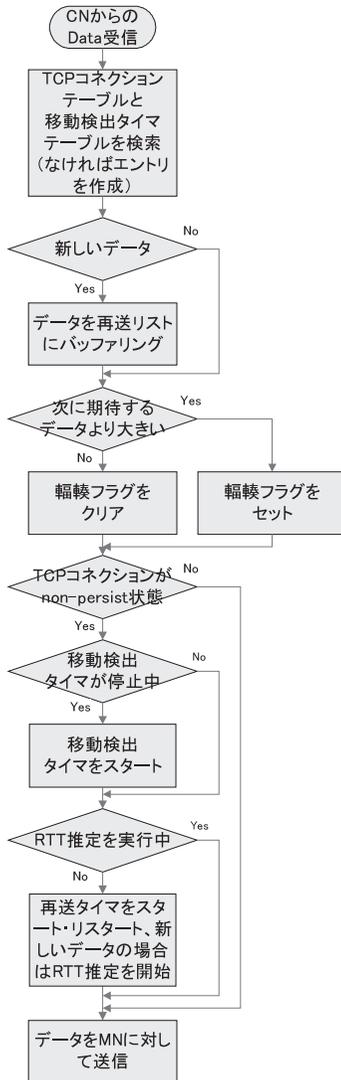


図 1 CN から受信したデータセグメントに対する FA の処理
 Fig. 1 Procedure for received data from CN at FA.

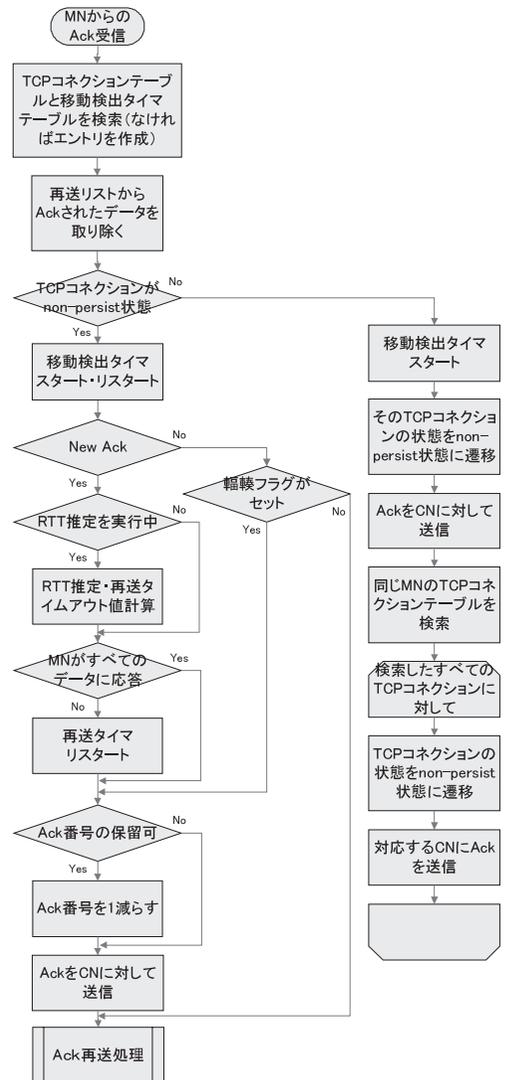


図 2 MN から受信した Ack セグメントに対する FA の処理
 Fig. 2 Procedure for received ACK from MN at FA.

出するために、MN ごとに移動検出タイマ管理情報を保持する。

FA が CN からデータセグメントを受信した場合の処理を図 1 に示す。FA はまず受信したデータセグメントに対応する TCP コネクションテーブルと移動検出タイマテーブルを検索する。新たなコネクションに対するセグメントである場合はエントリを作成する。次に受信したデータセグメントが新たなデータを含むかを検査し、含む場合は再送リストの最後に追加する。またデータセグメントが次に受信を期待するシーケンス番号を持つ場合はコネクションテーブルの輻輳フラグをクリアし、そうでない場合は輻輳フラグをセットする。このフラグは、重複 Ack の受信時と、移動検出

タイマのタイムアウト時の処理において使用される。続いて、TCP コネクションが non-persist 状態の場合は移動検出タイマと再送タイマを必要に応じて起動する。なお、再送タイマについては RTT の推定を行っていない場合（再送タイマが停止中か、RTT 推定のための Ack は受信したがすべてのデータが応答されていないため再送タイマが動作中の場合）は、同タイマをスタートまたはリスタートする。またそのデータが CN から再送されたものでなければそのセグメントについて RTT 推定も行う。最後にデータセグメントを MN に対して転送する。

次に、FA が MN から Ack セグメントを受信した場

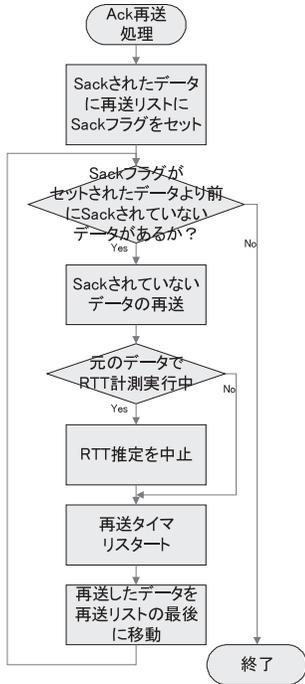


図3 Sack オプションを用いた FA における再送手順

Fig.3 Retransmit procedure at FA using Sack option.

合の処理を図2に示す。この場合も初めに2つのテーブルを検索し対応するエントリを得る。次にAckセグメントの確認応答シーケンス番号で受信が確認されたデータセグメントを再送リストから削除する。以下の処理はその時点でTCPコネクションがnon-persist状態であるか否かで異なる。non-persist状態の場合は、まず移動検出タイマをリスタートさせ、続いて受信したAckセグメントがNew Ackか重複Ackかを判断する。New Ackであれば、AckされたデータセグメントでRTT推定を行っていた場合はRTT値の推定と再送タイムアウト値の計算を行い、まだ受信応答されていないデータが存在する場合は再送タイマをリスタートする。さらに確認応答シーケンス番号(図中ではAck番号)を1減らしCNに対してそのAckセグメントを送出する。一方受信したAckセグメントが重複Ackであれば、輻輳フラグがセットされている場合(インターネットで輻輳が発生していると考えられる場合)はAckセグメントをCNに対して送信し、そうでなければデータを含まないものに対しては破棄する。最後に、後述するような手順でSackオプションを用いた再送を行う。

また、TCPコネクションの状態がpersist状態である場合は、MNが移動したと判断した後にそのMNからのセグメントを受信したことになるため、対応する

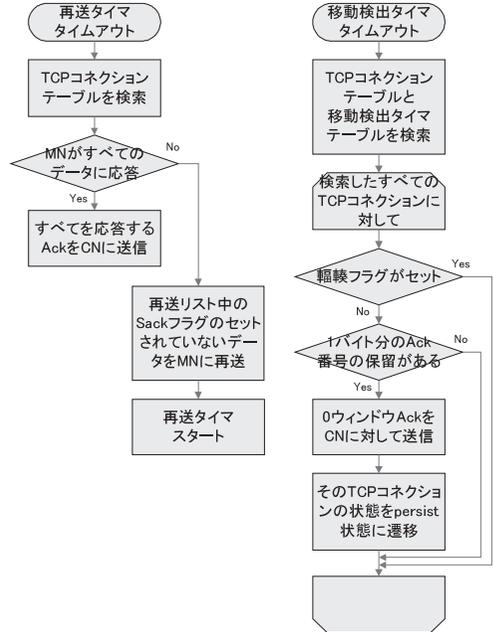


図4 再送タイマおよび移動検出タイマのタイムアウト処理 Fig.4 Timeout procedure of retransmit timer and movement detection timer.

TCPコネクションをnon-persist状態に変更し、さらに同じMNが確立しているTCPコネクションに対してもnon-persist状態に変更し、それぞれのコネクション上でAckセグメントを生成し送信する。

次に、FAがMNから受信したAckのSackオプションを用いて無線リンクで紛失したと考えられるセグメントを積極的に再送する手順を図3に示す。この処理ではAckセグメントで新たにSackされたデータよりも以前に転送済みのデータセグメントを再送し、そのセグメントを再送リストの最後尾にリンクする。これにより、再送したセグメントも含めてデータセグメントは転送した順に並べられることになり、新たにSackされたデータの前にMNが受信していないデータセグメントがあれば、その転送または再送が失敗したことが分かる。したがって紛失したデータセグメントのみを効率的に再送することができる。なおデータセグメントを再送する場合は、元のセグメントでRTT推定中の場合はその処理を中止し、またデータ転送時には再送タイマをリスタートする。

再送タイマおよび移動検出タイマがタイムアウトした場合の処理を図4に示す。再送タイマはコネクションごとに設定されているため、まずTCPコネクションテーブルを検索する。そのコネクションですべてのMNがデータに回答している場合は、CNからのデータ転送が完了していると判断し、FAで保留した1バ

イトの確認応答シーケンス番号を加えた Ack セグメントを CN に送出する。一方、MN が応答していないデータがある場合は、そのセグメントを MN に対して再送し、再送タイマをリスタートする。

また移動検出タイマは、MN ごとに用意されているため、このタイムアウトの場合は移動検出タイマテーブルと TCP コネクションテーブルの双方を検索し、MN が確立しているすべての TCP コネクションに対して、インターネットにおいて輻輳が発生していない場合にのみ、1 バイト分の確認応用シーケンス番号の保留があれば、ウィンドウ 0 の Ack を対応する CN に送信し、その TCP コネクションの状態を persist 状態に遷移させる。1 バイトの保留がないならば、データ転送が終了したと考えられるため、そのままの状態とする。

TCP 関連の情報のうちでは、再送リストがウィンドウサイズに相当するメモリサイズを必要とし規模が大きくなる。しかし、たとえばウィンドウサイズが 64 K バイトであるとしても、FA に同時に 1000 本の TCP コネクションが存在した場合でも、64 メガバイトのメモリで処理することができる。さらに、メモリが確保できない場合は、バッファを確保せず再送を行わないという処理で対応することができる。

2.2.2 Mobile IP の移動登録時にデータ転送を再開するための手順

MN が別の FA の下に移動すると、Mobile IP の手順に従って移動登録が開始されるが、提案方式では、移動先の FA が MN から送信された Registration Request を HA に転送する際、新たに FA-TCP Status Query Extension を追加する。この Extension 付きの Registration Request を受信した HA は、MN の移動元の FA が保持している MN の TCP 通信の情報を入手するために、MN の移動元の FA に対し、TCP Status Query メッセージを送信する。これを受信した移動元の FA は、HA に TCP Status Report メッセージを返す。このメッセージは MN が確立中である TCP コネクションごとに以下のものを含む。

- MN の IP アドレス
- CN の IP アドレス
- MN と CN のポート番号
- MN から受信した確認応答シーケンス番号
- CN に送信した確認応答シーケンス番号
- FA にウィンドウ 0 の Ack を送信したかどうかのフラグ
- MN から受信したウィンドウサイズ
- 輻輳フラグ

TCP Status Report メッセージを送信した後、移動元の FA は、対応する MN の TCP コネクションテーブルと再送リストを解放し、対応する移動検出タイマの管理情報を削除する。なお、移動が通知されない場合でも、FA におけるビジタリスト (Mobile IP で使用される FA の配下の MN の管理情報) からタイムアウトにより、MN の情報が削除される場合は、その MN に対応する TCP 関連の情報も解放する。

TCP Status Report メッセージを受信した HA は、これらの情報を含む FA-TCP Status Report Extension を Registration Reply に新たに追加し、MN の移動先の FA に送信する。またこのとき、CN に対してウィンドウ 0 の Ack セグメントを送信していた場合は、HA が CN に対してウィンドウを開く Ack を送信し、通信を再開させる。この処理において CN に送信する Ack セグメントの確認応答シーケンス番号とウィンドウサイズは TCP Status Report メッセージの情報から決定する。

本方式では移動直後も移動前のウィンドウサイズを用いてデータ転送を再開する。移動直後には輻輳ウィンドウを減らすことで輻輳を回避するという方式も考えられるが、これについては考慮していない。移動直後のバースト的な連続データ転送で輻輳が生じた場合は、本方式を用いた場合も輻輳制御が実行される。

3. 通信シーケンス例

本章では、シーケンス例を用いて提案方式を具体的に説明する。図 5 はホームエージェント HA に属する移動端末 MN が、CN からのデータを受信中に FA1 から FA2 のネットワークに移動した場合の全体的な手順例である。最初に CN は MN に対し 4 つのデータセグメントを送信している。図の「1:1025 (1024)」はシーケンス番号 1 で長さ 1024 バイトのデータセグメントであることを示す。また HA から FA1 までの太い矢印は、カプセル化して転送されていることを示す。FA1 はシーケンス番号 1 のデータを受信すると、まず自身の TCP コネクションテーブルに新たなエントリを作成し、次に受信したセグメントを再送リストにバッファリングする。その後、再送タイマ、移動検出タイマをスタートさせ、そのデータセグメントを MN に転送する。続いて FA1 は、シーケンス番号 1025、2049、3073 のデータセグメントを受信すると、受信した順番で再送リストへバッファリングし、MN に転送する。この時点では再送タイマが起動されており、シーケンス番号 1 のデータセグメントに対して RTT 推定を行っている。シーケンス例では、FA1 が転送し

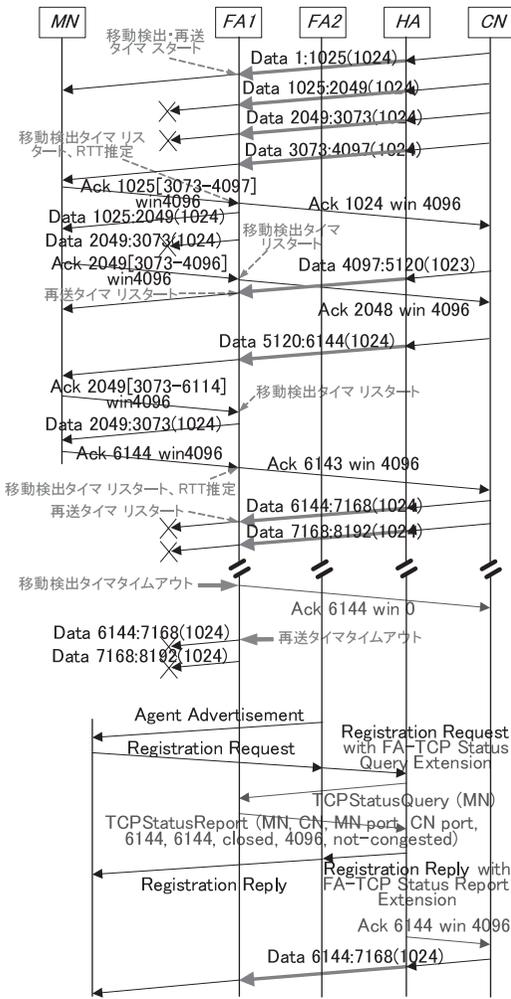


図 5 全体的なシーケンス例
Fig. 5 Overall sequence example.

たデータセグメントのうち、シーケンス番号 1 と 3073 のものは MN により受信されるが、シーケンス番号 1025 と 2049 のものは無線区間での伝送誤りで紛失したと想定している。

MN はシーケンス番号 3073 のデータを受信すると、Sack オプション付きの Ack セグメントを返す。「Ack 1025[3073-4097]」は、確認応答シーケンス番号 1025 と、3073 から 4096 までの番号のデータの受信を示す Sack オプションが含まれていることを、「win 4096」はウィンドウサイズが 4096 であることを、それぞれ示す。FA1 はこれを取り込むと、再送リストから MN の受信が確認できたシーケンス番号 1 のデータセグメントを取り除くと同時に、移動検出タイマをリスタートさせる。この Ack は New Ack であり、FA1 はシーケンス番号 1 のデータセグメントに対して RTT 推定

を行っているため、この Ack の受信を用いて RTT 推定値を更新し、再送タイムアウト値を再計算する。しかしまだ応答されていないデータセグメントがあるため、再送タイマはそのままとする。さらに FA1 は確認応答シーケンス番号を 1 減らして 1023 までのデータを受信したとして CN に Sack オプションのない通常の Ack を転送する。続いて、再送リスト中の 3073 のデータセグメントに Sack フラグをセットする。この処理により FA1 の再送リストではシーケンス番号 1025 と 2049 のデータセグメントが Sack フラグをセットしたデータよりも前になることから、FA1 はこれらのデータが無線区間中で紛失したと判断し、この順番で再送しこれらのセグメントを再送リストの最後に移動する。また、再送タイマをリスタートする。

このとき FA1 によって再送されたデータセグメントのうち、シーケンス番号 2049 のセグメントは伝送誤りにより紛失したと想定する。

次に MN が 200 ミリ秒タイマの発火により確認応答シーケンス番号が 2049 で、3073 から 4096 までの番号のデータの受信を示す Sack オプション付きの Ack を返す。FA1 はこれを受信すると、再送リストの中のシーケンス番号 1025 のセグメントを取り除く。またこの Ack は New Ack であるがこの時点では RTT 推定を行っておらずまた応答されていないデータセグメントが存在するため、再送タイマはそのまま動作させる。次に FA1 は確認応答シーケンス番号を 1 減らした Ack セグメントを CN に転送する。また再送リストには、再度紛失したシーケンス番号 2049 のデータセグメントの方が、Sack フラグがセットされたシーケンス番号 3073 のデータよりも後にあるため、この時点では再送されない。

一方、FA1 が先に送信している確認応答シーケンス番号が 1024 の Ack セグメントが CN に通知され、ウィンドウが開き、シーケンス番号 4097 を持つ次のデータセグメントが送信される。FA1 はこのセグメントを MN に転送し、再送タイマをリスタートし RTT 推定を開始する。さらに FA1 が転送した確認応答シーケンス番号が 2048 の Ack セグメントが CN に通知され、これに対し CN がシーケンス番号 5120 のデータセグメントを送信する。このセグメントが FA1 により MN に転送されると、MN は図に示すように確認応答シーケンス番号が 2049 で、3073 から 6114 までのデータの受信を示す Sack オプション付きの Ack セグメントを送信する。これを FA1 が受信するとシーケンス番号 2049 のデータセグメントを再送し、その一方でこの Ack セグメントが重複 Ack がかつ輻輳が

発生していないため、廃棄する。さらに FA1 が再送したデータセグメントが MN により受信され、MN がこれまでのデータセグメントをすべて受信する Ack を送信する。

次に MN が移動を開始し、CN が新たに MN に対して送信したシーケンス番号 6144 と 7168 のデータセグメントは MN に到達することができなくなる。図のように、FA1 において、まず移動検出タイムがタイムアウトし、確認応答シーケンス番号 6144 の Ack をウィンドウ 0 で送信し、CN をパーシストモードにする。次に、再送タイムがタイムアウトし、2 つのデータセグメントの再送が行われる。

一方 MN は、FA2 からの Agent Advertisement により移動を検知し HA に登録を行う。このとき FA2 は、新たに移動してきた MN に対する TCP の状態の問合せを要求する FA-TCP Status Query Extension を Registration Request に追加して HA に転送する。HA はこれを受信して MN が移動前に属していた FA1 に対して、TCPStatusQuery メッセージを送出する。FA1 はこれに対し、対応する MN の TCP 通信の状態を TCPStatusReport メッセージにより通知する。TCP 通信の状態は、コネクションごとに、MN と CN の IP アドレス、MN と CN のポート番号、MN から受信した確認応答シーケンス番号 (6144)、CN に通知した確認応答シーケンス番号 (6144)、ウィンドウを閉じたかどうかのフラグ、MN から受信したウィンドウサイズ (4096)、輻輳フラグを含む。また、この時点で FA1 は MN に対するシーケンス番号 6144 と 7168 のデータセグメントの再送を止め、この TCP コネクションに関する情報を廃棄する。HA は通知された TCP 通信の情報を Registration Reply メッセージの FA-Status Report Extension により FA2 に通知する。さらに HA は MN が正常にハンドオフを終了したことを知るので、確認応答シーケンス番号 6144、ウィンドウサイズ 4096 の Ack を CN に 1 つ送信する。これにより CN はパーシストモードから解除され、速やかにデータ転送を再開することができる。

しかし提案方式においても、MN が移動していないにもかかわらず、FA がウィンドウサイズ 0 の Ack セグメントを CN に送信しデータ転送を停止させる、逆に MN の移動後にウィンドウサイズ 0 の Ack セグメントが届く前に、CN がタイムアウトを発生させるなどの状況が起こりうる。たとえば、ウィンドウサイズ内の最後のデータセグメントや、Ack セグメントが無線区間で紛失した場合は、MN からの応答がないため移動検出タイムがタイムアウトし、FA がウィンドウ

サイズ 0 の Ack セグメントを CN に送信し、CN からのデータを停止させる。しかしこの場合も、FA においてタイムアウト再送を行うため、再送されたデータセグメントが MN に届き、その Ack セグメントが FA に返されれば、FA からウィンドウを開く Ack として CN に転送される。その場合は、CN における輻輳制御が起動されないため、速やかなデータ転送が可能となる。

逆に、ハンドオフまたは伝送誤りによるパケット紛失により、CN においてタイムアウトが生ずる場合がある。この場合は、CN において輻輳ウィンドウが 1 セグメント分に設定されてしまい、その後に FA からウィンドウサイズ 0 の Ack セグメントとウィンドウを開く Ack セグメントを送信しても、その状況は変化しない。このため、CN における不必要なタイムアウトは避ける必要がある。

4. 性能評価

この章では、ネットワークシミュレータ ns2¹⁷⁾ を用いて、計算機シミュレーションにより性能評価を行った結果を示す。

4.1 評価した方式およびソフトウェアの構成

評価においては以下を対象とした。

- 提案手法：2 章で述べた方式
- 提案手法を HA に実装した方式 (提案手法 in HA) : Mobile IP においてはバックボーンルータにおける Ingress Filtering の影響を避けるために逆方向トンネリングを用いることが推奨される。この場合は双方向の通信が HA を経由するため、提案手法を FA ではなく HA に実装することができる。この手法では 2.2.1 項に述べたハンドオフと伝送誤りに対する処理を HA で行わせるとともに、2.2.2 項に述べた移動登録時の処理を、明示的な追加オプションとメッセージを用いずに HA の内部で実行する。この手法では MN から CN への通信において HA を経由する遅延時間が追加されるが、移動登録時の HA と移動前の FA における TCP Status Query と TCP Status Report の交換のオーバヘッドが減る。
- Snoop : Mobile IP 環境で利用できるように、すべての FA において、他の FA とは独立に Snoop の手順を行わせたもの。
- M-TCP : 同様にすべての FA において、他の FA とは独立に M-TCP における SH の動作を行わせたもの。Mobile IP の移動手順と SH の切替えは関連していない。リンクレベルの再送の最大値を

増加させることも考慮する。

• オリジナルの Mobile IP

シミュレーションにあたり、まず ns2 自身のソフトウェアを以下のように変更した。ns2 では、TCP の処理を実現する TCP エージェントにおいて、便宜上パケット単位にシーケンス番号を割り当てている。しかし、提案手法や M-TCP では、FA において Ack の確認応答シーケンス番号を 1 バイト分減らす処理を行う必要がある。このため、本来の TCP と同様にシーケンス番号をバイト単位に付与するように TCP エージェントのソフトウェアを変更した。さらに TCP エージェントに対し、パーシストタイムにともなうウィンドウ検査の送出機能など、パーシストモードを実現するための機能を追加した。

次に、提案手法の機能を以下のように実装した。FA または HA において TCP セグメントや制御メッセージを処理するために、FA または HA を通過するすべてのパケットをキャプチャし、必要な処理を行う MIP-TCP ノードを新たに作成した。MIP-TCP ノードではメッセージの監視と提案手法の処理を行う MIP-TCP エージェントを動作させた。シミュレーションにおいては、MIP-TCP ノードを FA または HA と広帯域、遅延なしの有線で接続し、便宜上、提案手法を処理する 1 つのノードと見なした。FA と接続された MIP-TCP ノード中の MIP-TCP エージェントは、2.2.1 項で示したアルゴリズムを処理し、また 2.2.2 項の FA 側の実行する機能を持つ。HA に接続された MIP-TCP ノードの MIP-TCP エージェントは、2.2.2 項で示したように、パーシストモードの CN に対して、TCP Status Report の情報によりウィンドウを開く Ack を送信するなどの機能を持つ。また、提案手法を HA で動作させるために、ns2 の Mobile IP プログラムに逆方向トンネリングの機能を実装し、上記の MIP-TCP ノードを機能改修したものを HA に接続させた。

さらに、Snoop と M-TCP を実現するため、新たに Snoop エージェントと M-TCP エージェントを作成し FA に接続された MIP-TCP ノード内で動作させることとした。Snoop エージェントは無線区間でのパケット紛失に起因した 1 個目の重複 Ack により、紛失したセグメント再送する機能を持ち、また、無線区間でのパケット紛失に起因したすべての重複 Ack を破棄する機能を持つ。さらに FA-MN 間の RTT からタイマによる紛失パケットの再送を行う。M-TCP エージェントは Ack の確認応答シーケンス番号を 1 バイト分減らす処理を実行し、FA-MN 間の RTT からタイマにより CN に対して保留している確認応答シーケ

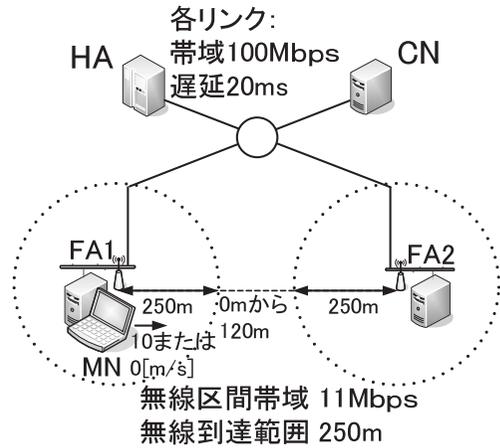
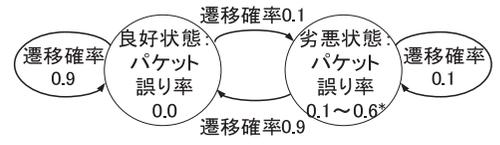


図 6 シミュレーションのネットワーク構成
Fig. 6 Network configuration for simulation.



*: 但しAckセグメントに対しては×0.05とする

図 7 2 状態マルコフエラーモデル

Fig. 7 2 state Markov error model.

ンス番号でウィンドウサイズ 0 の Ack を送信する機能を持つ。

4.2 シミュレーションのシナリオと評価項目

図 6 にシミュレーションのネットワーク構成図を示す。HA に属する移動端末 MN が、FA1 または FA2 の下で CN からのデータを受信する。CN の送信するデータセグメントは IP ヘッダを含めて 1500 バイトとし、TCP のウィンドウサイズは 65000 バイトとする。無線リンクは IEEE802.11 を使用し、帯域 11 Mbps、無線到達範囲 250 メートル、リンクレベルでの再送回数を最大 4 回(リンクレベルの再送回数を変化させる評価では 4 回から 7 回と 15 回)、RTS/CTS による仮想キャリアセンスありとした。なお、無線区間の伝送誤りとして、データおよび Ack セグメントの送信の双方に対して、電波が良好な状態と劣悪な状態を確率的に遷移させる 2 状態マルコフモデルを用いた。ここでは、図 7 のように良好状態から劣悪状態への遷移確率を 0.1、その逆の遷移確率を 0.9 とし、各状態は 1 秒間継続されることとした。また、良好状態としてパケット誤り率 0 を、劣悪状態としてデータセグメントを含むパケットの誤り率に対して 0.1 から 0.6 を用いた。また Ack セグメントを含むパケットに対しては、劣悪状態において IEEE802.11 のフレーム長

(データセグメントは 1536 バイト, Ack セグメントは 76 バイト, PLCP ヘッダは含んでいない) に比例したパケット誤り率を用いることとした。有線リンクについては帯域 100 Mbps, 各リンクの伝播遅延 20 ミリ秒, ビット誤りなしとした。MN と CN の往復遅延は, 提案手法, Snoop, M-TCP, Mobile IP のみによる手法では 120 ミリ秒, 提案手法を HA に実装した手法では 160 ミリ秒となる。また M-TCP においては CN におけるパーシストタイムを 5 秒とした。

このような条件下で, 次のような評価を行った。

評価 ①: まず, 伝送誤りとハンドオフの双方が生ずる環境において, TCP 通信のスループットを測定した。この評価では, シミュレーション開始 10 秒後に FA1 の下で CN から TCP によるデータ受信を開始する。さらに MN は通信開始 10 秒後に FA2 の下に向かって FA 間を結ぶ直線上を秒速 10 メートルで移動を開始し, 通信不能範囲を通った後, FA2 の直下に到着するまで移動し, その後は FA2 の下で停止するものとした。FA1 と FA2 の間の通信不能距離については, 0 メートルから 120 メートルまで変化させた。このような環境でシミュレーション開始後 100 秒までの 90 秒間の TCP 通信のスループットを測定した。

評価 ②: 次に各手法の詳細な解析を行うために, MN が移動していない状況において下記の項目を測定した。この評価ではシミュレーション開始 10 秒後から 50 秒後までの 40 秒間, MN と CN の間で TCP 通信を行わせた。

- 評価 ② における TCP スループット
- CN において起動された TCP レベルの再送回数
- 伝送誤りを回復するために FA (または HA) により TCP レベルで行われた再送回数とそのうちの不要なものの回数
- FA (または HA) から送信されたウィンドウサイズ 0 の Ack の送信回数
- IEEE802.11 のフレーム再送回数の最大値を増加させた場合の TCP スループット
- 提案手法の再送タイムおよび移動検出タイムのタイムアウト値の時間的变化

4.3 シミュレーション結果と考察

評価 ① の結果を図 8, 図 9, 図 10 に示す。各図は横軸が通信不能区間の距離, 縦軸が 90 秒間の TCP 通信のスループットである。2 状態マルコフモデルの劣悪状態におけるデータセグメントを含むパケットの誤り率が, 0 (すなわち伝送誤りなし), 0.2, 0.5 の場合の結果が図 8, 9, 10 に対応する。図では, それぞれ提案手法を「提案手法」, 提案手法を HA に実装した

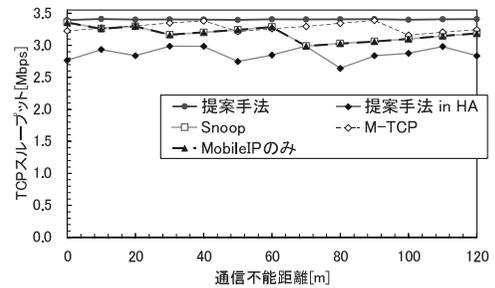


図 8 通信不能距離と TCP スループット (劣悪状態のデータパケット誤り率 = 0.0)

Fig. 8 TCP throughput vs. distance of inter radio range (Data packet error rate in bad condition = 0.0).

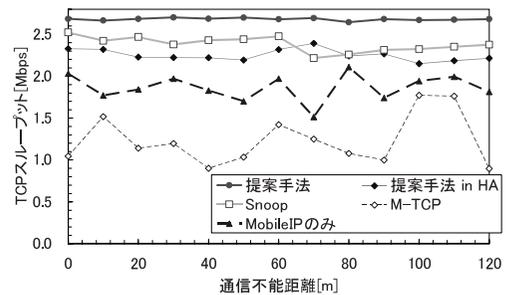


図 9 通信不能距離と TCP スループット (劣悪状態のデータパケット誤り率 = 0.2)

Fig. 9 TCP throughput vs. distance of inter radio range (Data packet error rate in bad condition = 0.2).

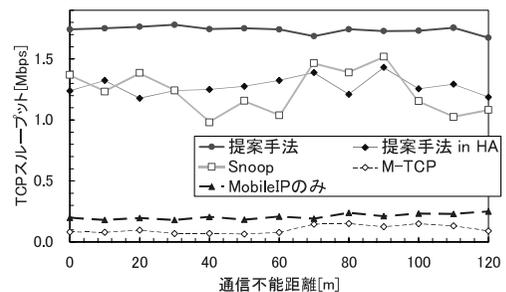


図 10 通信不能距離と TCP スループット (劣悪状態のデータパケット誤り率 = 0.5)

Fig. 10 TCP throughput vs. distance of inter radio range (Data packet error rate in bad condition = 0.5).

た手法を「提案手法 in HA」, Snoop を用いた手法を「Snoop」, M-TCP を用いた手法を「M-TCP」, オリジナルの Mobile IP による通信を「Mobile IP のみ」と示してある。

図 8 を見ると, 伝送誤りがなくハンドオフのみの場合はスループットの大きな違いは見られないが, なかでは提案手法のスループットが最も高く, 通信不能区間の距離を変えた場合もスループットに大きな変化は見られない。また, 提案手法 in HA のスループット

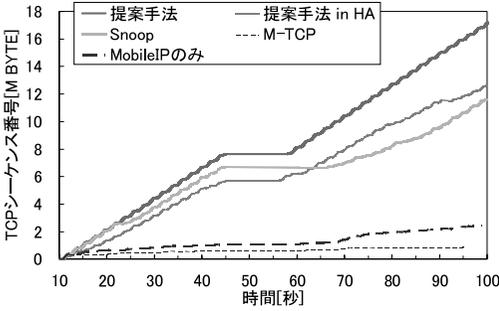


図 11 受信のシーケンス番号の時間的変化 (劣悪状態のデータパケット誤り率 = 0.5)

Fig. 11 Sequence number at MN vs. receiving time (Data packet error rate in bad condition = 0.5).

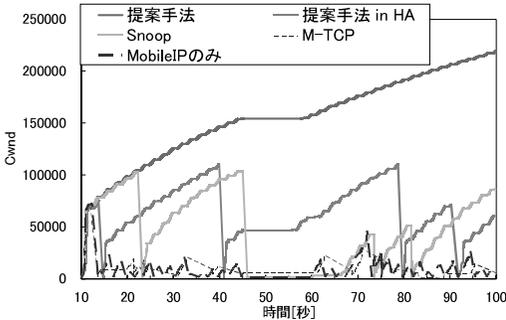


図 12 送信側の Cwnd の時間的変化 (劣悪状態のデータパケット誤り率 = 0.5)

Fig. 12 Cwnd at CN vs. sending time (Data packet error rate in bad condition = 0.5).

が最も低いのは、逆方向トンネリングによる通信遅延の増加に起因している。Snoop と Mobile IP のみがほぼ同様で、M-TCP と異なっているのは、前者が再送タイマにより、後者がパーシストタイマによりそれぞれ動作を再開しているためである。このように若干のちがいはあるものの、いずれの手法においても、ハンドオフのみの TCP スループットへの影響は小さいといえる。

図 9 と図 10 より、2 状態マルコフモデルでの劣悪状態 (1 秒ごとに確立 0.1 で遷移) のデータパケット誤り率が 0.2、0.5 と増加するにつれて、提案手法以外の手法のスループットが低下することが分かる。0.2 の場合は、M-TCP と Mobile IP のみの手法のスループットが低下する。0.5 になるとこの両者のスループットが非常に低下し、提案手法 in HA と Snoop の手法のスループットも提案手法に比べて低くなる。

劣悪状態のパケット誤り率が 0.5 で、通信不能距離が 100 メートルの場合に対して、MN で受信したデータセグメントのシーケンス番号と、CN での輻輳ウィン

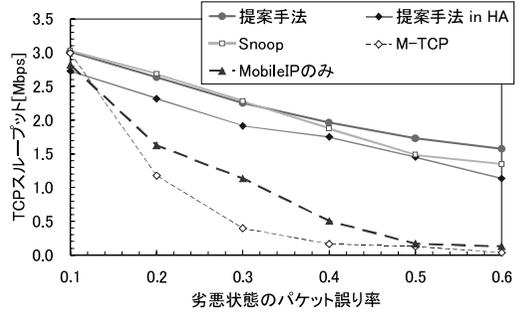


図 13 評価 ② における TCP スループット

Fig. 13 TCP throughput in the second evaluation.

表 1 CN における TCP レベルの再送回数

Table 1 TCP retransmissions at CN.

劣悪状態の誤り率	提案手法		提案 in HA		Snoop		M-TCP		MIPのみ	
	Ack 再送	TO 再送	Ack 再送	TO 再送	Ack 再送	TO 再送	Ack 再送	TO 再送	Ack 再送	TO 再送
0.1	0	0	0	0	0	0	0	0	1	1
0.2	0	0	0	0	0	0	10	0	7	6
0.3	0	0	0	1	0	0	14	0	14	8
0.4	0	0	0	0	0	1	15	0	23	14
0.5	0	0	0	1	0	1	13	0	16	14
0.6	0	1	0	2	0	0	7	0	14	15

注 提案 in HA: 提案手法 in HA, MIPのみ: Mobile IPのみ, Ack再送: 重複Ackによる再送回数, TO再送: タイムアウトによる再送回数

ドウ (Cwnd) の時間的変化を、図 11 と図 12 にそれぞれ示す。これらの結果から、提案手法と提案手法 in HA では CN における Cwnd の減少もなく順調なデータ通信が可能となっていることが分かる。また Snoop は、伝送誤りのみの場合は提案手法と同程度の性能を実現するが、ハンドオフ後は、Cwnd の減少があり通信の再開が緩やかとなる。これに対して M-TCP と Mobile IP のみの手法では、ほとんど Cwnd が増加していない。

次に評価 ② の結果を以下に示す。まずこの評価において、2 状態マルコフモデルの劣悪状態でのデータパケット誤り率を変更した場合の 40 秒間の TCP スループットを図 13 に示す。この図に示すように、パケット誤り率が低い場合は、提案手法と Snoop がほぼ同一のスループットを得ているが、誤り率が高くなるにつれて差が生ずる。一方 TCP レベルの再送機能がない M-TCP と Mobile IP のみの各手法では、マルコフモデルの劣悪状態のパケット誤り率が高くなるにつれて大幅にスループットが低下している。

表 1 に、CN において起動されたデータセグメントの再送の回数を示す。この再送は伝送誤りが原因となるため、本来は避けるべきものである。提案手法、提案手法 in HA, Snoop の各手法では、誤り率が増加

表 2 FA における TCP レベルの再送回数と不要再送回数
Table 2 TCP retransmissions at FA including unnecessary ones.

劣悪状態の誤り率	提案手法		提案手法 in HA		Snoop	
	FAによる再送	内不要な再送	HAによる再送	内不要な再送	FAによる再送	内不要な再送
0.1	0	0	104	102	3	2
0.2	3	0	95	77	16	14
0.3	24	1	77	60	41	27
0.4	56	1	104	61	82	29
0.5	89	6	116	52	102	32
0.6	120	2	164	48	138	34

表 3 FA から通知されたウィンドウサイズ 0 の Ack 数
Table 3 Number of zero window Acks sent by FA.

劣悪状態の誤り率	提案手法	提案 in HA	M-TCP
0.1	0	0	1 (0)*
0.2	0	1	6 (3)
0.3	1	0	6 (4)
0.4	1	0	7 (5)
0.5	3	0	7 (5)
0.6	4	1	5 (4)

*: M-TCPに対しては、CNが5秒間パーシストモードとなった回数をカッコ内に示す

してもほとんど生じないが、M-TCP と Mobile IP のみの各手法では、CN からの再送が発生しておりこれが輻輳制御を起動している。M-TCP では CN がタイムアウトする前に FA からウィンドウサイズ 0 の Ack セグメントが送られるため重複 Ack による再送のみが発生していることが特徴的である。

表 2 に、FA (提案手法 in HA では HA) から TCP レベルで行われた再送回数と、そのうちで不要だったものの数を示す。不要であった再送とは、FA または HA から再送により、同一セグメントが複数 MN に到着した場合を意味する。この表より提案手法では、無線リンクで紛失したデータセグメントを FA が再送し、不要な再送も少ないことが分かる。一方、提案手法を HA に実装した場合は不要な再送が増加している。この再送はすべて再送タイムアウトによる再送であり、再送タイムが正しく推定できなかったためと予想される。なおこの不要な再送数の全体の送信セグメントに対する割合は 1%程度である。また Snoop でも不要な再送が若干生じている。これらはすべて Ack セグメントが紛失したことによるタイムアウト再送であった。提案手法では Sack を用いた再送を行っているため、Ack セグメントが紛失した場合も次の Ack の中の Sack オプションによりデータセグメントが判別できるため、この問題を解決している。

次に表 3 に、FA (提案手法 in HA では HA) から

表 4 リンクレベルの再送回数を変更した場合の TCP スループット (Mbps)

Table 4 TCP throughput for various link level retransmission limits (Mbps).

劣悪状態の誤り率	提案手法					M-TCP				
	4	5	6	7	15	4	5	6	7	15
0.1	3.01	3.01	3.01	3.01	3.01	2.99	2.99	2.99	2.99	2.99
0.2	2.64	2.65	2.64	2.64	2.64	1.18	2.50	2.66	2.66	2.66
0.3	2.25	2.28	2.30	2.30	2.33	0.40	1.21	1.17	1.21	1.56
0.4	1.96	1.91	1.94	1.93	1.91	0.17	0.32	0.47	0.97	0.97
0.5	1.73	1.73	1.76	1.66	1.61	0.13	0.16	0.43	0.26	0.33
0.6	1.58	1.52	1.42	1.32	1.28	0.04	0.09	0.13	0.08	0.06

CN に送信されたウィンドウサイズ 0 の Ack セグメントの送信回数を示す。提案手法と提案手法 in HA の各手法に比べて、M-TCP ではやや増加している。M-TCP では、FA がウィンドウサイズ 0 の Ack を送信した後、MN から Ack を受信しないと CN に何も転送せず、CN はパーシストモードを保つ。パーシストモードではタイムアウト再送が動作しないため、この状況ではパーシストタイマの 5 秒間データ転送が停止する。その回数を表の M-TCP の部分に括弧で示した。誤り率が増加するとその数が増加する。たとえば 5 回パーシストタイマが動作すると 40 秒の評価時間のうち 25 秒が通信停止状態となる。これにより M-TCP のスループットが大きく低下したと考えられる。これに対して提案手法および提案手法 in HA では、CN がパーシストモードとなりタイムアウト再送を行わなくても、FA または HA が代わりにタイムアウト再送を行うため、その再送に対する Ack セグメントが CN に到達しパーシストタイマのタイムアウトより早くパーシストモードから抜け出すことが可能となっている。

次に表 4 に、IEEE802.11 のフレーム再送回数の最大値を増加させた場合の TCP スループットを示す。伝送誤りからの回復を行うためにはリンクレベルの再送を増やす方法が考えられる。そこでハンドオフへの対応は M-TCP で、伝送誤りへの対応はリンクレベルの最大再送回数を増加させるという手法の性能について評価した。表 4 の結果から、リンクレベルの再送を増やしてもスループットの大幅な改善は得られず、さらにスループットが低下する場合もあるといえる。特に劣悪状態のパケット誤り率が 0.6 の場合は、低下の傾向が強い。通信中のリンクレベルの再送フレームの割合を評価すると、最大再送回数が 4, 5, 6, 7, 15 のときに、それぞれ 12.2%, 11.8%, 13.4%, 13.3%, 13.8%であった。このように最大送信回数を増加させることによりリンクレベルの再送のオーバーヘッドが増

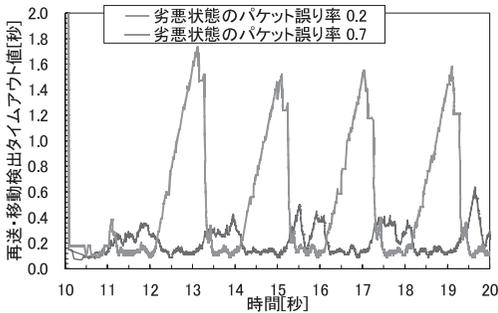


図 14 再送・移動検出タイムアウト値の時間的变化

Fig. 14 Retransmission/move detection timeout vs. time.

加する傾向にあり、このためスループットが低下したと考えられる。これらの結果から、TCP レベルの再送の方がより効果が高いと考えられる。

最後に、提案手法における再送タイムおよび移動検出タイムのタイムアウト値の時間的变化を図 14 に示す。2 状態マルコフモデルの劣悪状態のパケット誤り率が増加すると、リンクレベルの再送回数が増え、伝送遅延時間も増加する。このため、図に示すように、誤り率が 0.5 である場合、0.2 の場合に比べて、再送・移動検出タイムアウト値の最大値が大きくなっている。またタイムアウト値が振動しているのは、パケット誤り率の良好状態と劣悪状態を定期的に遷移しているためと考えられる。このように提案手法のタイムは状況に応じてその値を変化していることが分かる。

5. おわりに

本稿では、Mobile IP に基づくネットワークにおいて、ハンドオフおよび無線区間の伝送誤りに起因する TCP のスループット低下を防ぐ手法を示した。提案手法では、FA において CN からのデータセグメントを MN が受信応答するまで保持し、Sack と再送タイムを用いてパケット紛失を検出した場合、再送を行い、無線区間の伝送誤りを隠蔽した。また、MN の移動に対しては FA が CN にウィンドウサイズ 0 の Ack を送信し、CN をパーシストモードとさせることで対応し、かつ移動直後に Mobile IP の移動登録手順と協調させることで、すばやいデータ転送の再開を可能とさせた。

また、本稿ではソフトウェアシミュレータ ns2 を用いて提案手法を評価した結果を示した。シミュレーションの結果、提案手法は、Snoop、M-TCP などの既存の手法と比べて、ハンドオフと伝送誤りの双方が存在する状況で高速な TCP 通信とすることができることが明らかとなった。特に、M-TCP では伝送誤り

でウィンドウサイズ 0 の Ack が送信されると、CN がパーシストタイムのタイムアウト後にしかデータ通信を再開できない場合があり、性能が大幅に低下する。この傾向はリンクレベルの再送回数を増大しても同様である。これに対し提案手法は、ウィンドウサイズ 0 の Ack が送られた後も FA においてタイムアウト再送を行うため、この問題を解決している。また、Mobile IP において逆方向トンネリングが行われる場合は、提案手法を HA に実装することも可能であり、無線区間で不要な再送が増加する、双方向トンネリングによる遅延の増加でスループットが低下するなどの問題はあられるものの、伝送誤りが大きい場合も高いスループットを実現できる。

なお、オペレーティングシステムによってはウィンドウサイズ 0 の Ack の通知の動作が異なる場合があるため、本方式を実装できる環境が制限されることが考えられる。具体的には NetBSD1.6, 4.BSD, Linux2.1.7 では本稿で示した動作が確認されているが⁽¹⁸⁾、より最新の Linux ではウィンドウサイズ 0 の Ack により、受信確認を行っていないデータセグメントの再送信が発生することは不可能となっている。

参 考 文 献

- 1) 山本 幹, 松田崇弘: 無線 TCP の研究動向, 信学技報, IN2003-111 (Nov. 2003).
- 2) Goff, T., Moronski, J., Phatak, D. and Gupta, V.: Freeze-TCP: A True End-to-End TCP Enhancement Mechanism for Mobile Environments, *Proc. IEEE INFOCOM 2000*, Tel-Aviv, Israel, pp.1537-1545 (Mar. 2000).
- 3) Brown, K., et al.: M-TCP: TCP for Mobile Cellular Network, *ACM Computer Communications Review*, Vol.27, No.5, pp.19-43 (Oct. 1997).
- 4) Perkins, C. (Ed.): IP Mobility Support for IPv4, RFC3344 (Aug. 2002).
- 5) Sinha, P., Venkitaraman, N., Sivakumar, R. and Bharghavan, V.: WTCP: A Reliable Transport Protocol for Wireless Widearea Networks, *Proc. ACM Mobicom'99*, Seattle, WA, pp.231-241 (Aug. 1999).
- 6) Casetti, C., Gerla, M., Mascolo, S., Sanadidi, M. and Wang, R.: TCP Westwood: End-to-End Congestion Control for Wired/Wireless Networks, *ACM Wireless Networks*, Vol.8, Issue 5, pp.467-479 (Sep. 2002).
- 7) Fu, C.P. and Liew, S.: TCP Veno: TCP Enhancement for Transmission over Wireless Access Networks, *IEEE Journal on Selected Areas in Communications*, Vol.21, No.2, pp.216-228

- (Feb. 2003).
- 8) Bakre, A. and Badrinath, B.: I-TCP: Indirect TCP for Mobile Hosts, *Proc. 15th IEEE International Conference on Distributed Computing Systems (ICDCS'95)*, Vancouver, Canada, pp.136-143 (May 1995).
 - 9) Balakrishnan, H., et al.: Improving TCP/IP Performance over Wireless Networks, *Proc. 1st ACM Int. Conf. Mobile Computing and Networking (Mobicom)*, Berkeley, CA (Nov. 1995).
 - 10) IEEE Std. 802.11-1999: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (1999).
 - 11) 3GPP TS 24.022: Radio Link Protocol (RLP) for Circuit Switched Bearer and Teleservices (Release 1999) (2003).
 - 12) Hu, J., et al.: Hierarchical Cache Design for Enhancing TCP over Heterogeneous Networks with Wired and Wireless Links, *IEEE Trans. Wireless Commun.*, Vol.2, No.2, pp.205-217 (Mar. 2003).
 - 13) Ebihara, J., Kato, T. and Itoh, S.: High Performance TCP Communication over Mobile IP Handoff for Mobile Node Receiving Data Segments, *Proc. IASTED International Conference on Communication Systems and Networks (CSN 2004)*, Marbella, Spain, pp.35-40 (Sep. 2004).
 - 14) Mathis, M., Mahdavi, J., Floyd, S. and Romanow, A.: TCP Selective Acknowledgment Options, RFC2018 (Oct. 1996).
 - 15) Border, J., et al.: Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations, RFC3135 (June 2001).
 - 16) Paxson, V. and Allman, M.: Computing TCP's Retransmission Timer, RFC 2988 (Nov. 2000).
 - 17) The Network Simulator - ns2.
<http://www.isi.edu/nsnam/ns/>
 - 18) Igarashi, K. and Yabusaki, M.: Mobility-Aware TCP: Making Robust TCP against Segment Loss in Wireless Networks, *IEICE Trans.*

Fundamentals, Vol.E86-A, No.7, pp.1636-1644 (July 2003).

(平成 18 年 5 月 19 日受付)

(平成 18 年 11 月 2 日採録)



海老原 成 (学生会員)

平成 14 年法政大学工学部電気電子工学科卒業。平成 16 年電気通信大学大学院情報システム学研究科博士前期課程修了。Mobile IP における TCP 通信高速化の研究に従事。現在、同博士後期課程在学中。



加藤 聰彦 (正会員)

昭和 53 年東京大学工学部電気工学科卒業。昭和 58 年同大学大学院博士課程修了。同年国際電信電話(現 KDDI)(株)入社。平成 14 年まで同社研究所および(株)KDDI 研究所において、OSI やインターネットの通信プロトコルの研究に従事。昭和 62~63 年米国カーネギーメロン大学客員研究科学者。平成 14 年より電気通信大学大学院情報システム学研究科助教授。高速/モバイルインターネット、アドホックネットワーク等の通信プロトコルの研究に従事。工学博士。平成 1 年元岡賞受賞。電子情報通信学会、IEEE 各会員。



伊藤 秀一 (正会員)

昭和 39 年東京大学工学部電子工学科卒業。昭和 44 年同大学大学院博士課程修了。同年電気通信大学講師。現在、同大学大学院情報システム学研究科教授。データ圧縮、パターン識別等の情報理論とその応用の研究に従事。IEEE, 電子情報通信学会, 情報理論とその応用学会各会員。