

## リンク層情報を利用したハンドオーバ時のTCPの効率化手法

韓 閔燮† 神谷 弘樹†† 西田 佳史††† 寺岡 文男†

† 慶應義塾大学大学院理工学研究科 〒223-8522 神奈川県横浜市港北区日吉3-14-1

†† 慶應義塾大学大学院理工学研究科 神奈川県横浜市港北区日吉3-14-1 ※現 NTT 未来ねっと研究所

††† ソニーコンピュータサイエンス研究所 〒141-0022 東京都東五反田3-14-13

E-mail: †{hanstone,tera}@tera.ics.keio.ac.jp, ††koki@ma.onlab.ntt.co.jp, †††nishida@csl.sony.co.jp

あらまし 本研究では無線ネットワークにおいてリンク層情報を取得し、その情報を利用してハンドオーバ時のTCPの最適化を行う機構を提案する。近年、インターネットを通じて映像や音声などを扱うサービスを利用するためには、途切れることのない連続的な通信が必要不可欠となってきた。そのため、異なる無線ネットワーク間を、通信を継続したまま移動する技術が必要となっている。しかし、モバイルノードが異なる無線ネットワーク間を移動するハンドオーバ時には、通信品質の劣化が起こるといった問題点がある。これらの発生を防ぐためにはハンドオーバ処理による通信不能時間の短縮、そしてハンドオーバに適応した輻輳制御が必要となる。本研究では、まず無線リンクの品質の劣化に着目し、ハンドオーバ後の帯域情報の取得方法および伝達方法に関して検討を行う。その結果を元に、上記に挙げた問題点を考慮したリンク情報を利用したハンドオーバ時のTCPの効率的な再開方法を提案し、オペレーティングシステムに実装して実験による有効性の評価を行う。

キーワード 高速ハンドオーバ, TCP, L2 Triggers, モバイルコンピューティング, 無線LAN

## An efficient TCP handover mechanism using link-layer information

Yunsop HAN†, Koki MITANI††, Yoshifumi NISHIDA†††, and Fumio TERAOKA†

† Keio University 3-14-1 Hiyoshi, Kohoku-ku, Yokohama-shi, Kanagawa, 223-8522 Japan

†† Keio University 3-14-1 Hiyoshi, Kohoku-ku, Yokohama-shi, Kanagawa, 223-8522 Japan

††† 3-14-13, Higashigotanda Shinagawa-ku, Tokyo, 141-0022 Japan

E-mail: †{hanstone,tera}@tera.ics.keio.ac.jp, ††koki@ma.onlab.ntt.co.jp, †††nishida@csl.sony.co.jp

**Abstract** In this paper, we propose a mechanism in which optimizes TCP throughput using the link information on wireless networks. Today, we need continuous network transmissions in order to use Internet services such as video and audio streaming. Therefore, the demands are rising toward the mobility between different wireless networks. However, when the mobile nodes migrate from one wireless network to the other using a handover technology, the degradation of transmission qualities happens. To prevent this degradation, we need a mechanism in which senses the degradation, reduces the handover time, and satisfies the items of the estimated band. In this paper, we discuss the indicators of the optimal handover determination focusing on the qualities of the wireless links. By using the result, we propose a handover mechanism using the link information in which resolves the problems listed above. And last but not least, we evaluate the effectiveness of this mechanism by implementing on the operating system.

**Key words** Fast Handover, TCP, L2 Triggers, Mobile Computing, Wireless LAN

### 1. はじめに

近年、無線ネットワーク環境の普及により、いつでも、どこでも通信が可能なモバイルコンピューティングが一般化しつつある。従来はWEBやメールが主であったのに対して、現在は映像や音声などのマルチメディアコンテンツも増加してい

る。しかし、現在のインターネット通信の大部分を占めるTCP (Transmission Control Protocol) 通信ではモバイルノードのハンドオーバに伴い、IPアドレスが変化すると通信が切断されてしまう。IPアドレスが変化しても通信を継続できる移動透過性の研究は盛んに行われている。移動透過性を実現するための技術として Mobile IPv6 [1] や LING (Location Independent

Networking for IPv6) [2] [3] などが提案されている。モビリティ環境における移動ノードは、移動中にサブネットを切り替えていくハンドオーバ処理を行う。このハンドオーバの中はパケットロスが発生する。さらにハンドオーバにより TCP での通信遅延や不要な輻輳制御の発生などの問題にも発生する。無線環境は有線とは異なり、高エラー率、高遅延、電波の干渉などの性質を有する。そのため、低品質な無線リンクにおけるパケットロスが発生するという問題があり、このパケットロスがさらに TCP の通信効率の低下を生じさせている。

本論文では、無線ネットワークにおいてハンドオーバ発生時に下位レイヤの切断完了、接続完了などの下位レイヤの情報を取得し、ハンドオーバ時間を短縮し、TCP レイヤでの通信の処理の遅延を短縮する、また、移動前に移動先のリンク層の利用可能な帯域情報を取得し、最適なウィンドウサイズを選択する。こうして取得した帯域情報を送信相手に通知することによる最適な TCP 通信のための機構を提案する。

## 2. 無線環境が TCP に与える影響

従来の TCP は End-to-End での動作が基本となるため、輻輳制御はエンドホストで得られる情報のみから実現される。受信ホストがパケットを正しく受け取ったことを示す確認応答 (ACK) により輻輳制御が行われる。TCP で使用される ACK は累積 ACK (Cumulative ACK) であり、ACK に記述されている確認応答番号は送信相手がこれまでに正しく受信したデータのシーケンス番号の累積値を示している。パケットを送信してから一定時間 ACK 返って来ない場合、送信ホストは輻輳ウィンドウサイズを初期値である 1 パケットに設定し、再送を行う。また、同一の確認応答番号を持つ重複 ACK が送信ホストで三つ受信された場合、軽度の輻輳によってパケットが損失したとみなし、タイムアウトを待たずにパケットを送信しウィンドウサイズを半分以下に下げ送信を再開する。このように TCP はネットワーク内部で発生するパケットロスは輻輳によるルータでのバッファがあふれることによるのみ発生するという前提の下で設定されている。しかし、無線ネットワークでは輻輳以外の原因によりパケットロスが発生する。無線リンクにおけるパケットロスとハンドオーバによるパケットロスである。無線リンクにおけるパケットロスは、無線リンクにおける雑音による伝送誤りに起因する。ハンドオーバによるパケットロスは、移動ノードが移動により接続する無線基地局を変更する際に発生する。しかし、従来の TCP では無線リンクのロスとハンドオーバロスを判別することができないため、すべてのパケットロスを輻輳と判断し、必要以上に輻輳ウィンドウを減少させ、TCP 通信効率を低下させてしまう。また、ハンドオーバ完了から TCP 通信再開までの遅延やハンドオーバ後の帯域の変化を考慮していないため、効率的な通信再開が行えないという問題がある。

## 3. 無線環境を考慮した TCP の関連研究

現在、無線環境において効率的な TCP を実現するための研究が多数ある。End-to-End での解決方式の例として、Freeze-

TCP [4] がある。また、レイヤ間情報交換方式の例として、ILC-TCP [5], Mobility manager [6] について簡単に説明する。

### 3.1 End-to-End 方式

End-to-End 方式は、エンドシステム上の TCP の実装に変更を加えるためのみを変更し、ルータや基地局での変更を必要としない。つまり、既存の中間ノードに影響を与えないため、そのすべてをエンドホストで行う方式である。ここでは、End-to-End 方式に関する研究として、Freeze-TCP を挙げる。

Freeze-TCP を用いた受信側ノードは、無線リンクの電波強度の低下などから切断を予測すると、送信ノードにウィンドウサイズ 0 を通知する (ZWA: Zero Window Advertisement)。送信ノードはこれを受信すると Persist mode となり、ZWP (Zero Window Probe) を定期的に送信する。この時、ZWP が失われても輻輳ウィンドウを減少させずに保ち続けるという規定がある。Freeze-TCP ではハンドオーバ後の送信ノードは以前の輻輳ウィンドウサイズを用いて送信を再開する。しかし、ハンドオーバ後は通信経路が変わっている可能性や移動先の無線セルが混雑している可能性もある。したがって、ハンドオーバ前の輻輳ウィンドウサイズをそのまま使用するのには問題がある。

### 3.2 レイヤ間情報交換方式

レイヤ間情報交換方式では他レイヤの情報を取得できることにより変化する環境に動的に対応することができる。現在の OSI (Open Systems Interconnection) 基本参照モデルでは各レイヤが独立して階層構造を成すように設計されており、基本的には他レイヤの情報は取得できない構造となっている。レイヤ間情報交換を利用し、TCP の性能を改善する研究として、ILC-TCP、Mobility Manager を挙げる。

ILC-TCP は、既存のレイヤに並列した新しいレイヤとしての State Manager が、リンク層の状態が良いか悪いかなどの L2 情報や IP 層での通信の可能かどうかの L3 情報を定期的に取得し、それらの情報を元に TCP 層の挙動を決定することで、TCP の効率を改良する機構である。ハンドオーバによりリンクが切断されていると判断した場合、TCP に FREEZE を通知することにより、輻輳ウィンドウの減少やタイムアウトなどを回避し、通信効率の低下を防ぐことができる。しかし、ILC-TCP はシミュレータによるものであり、実際のオペレーティングシステム上に実装されたものではない。また、移動後のネットワークの有効帯域を利用することが困難である。

Mobility Manager は、Mobile IP のようなモビリティ管理プロトコル、利用可能なネットワークステータスなどを収集、処理し、モバイルホスト上のアプリケーションに伝達する方式である。しかし、これらは共にアプリケーションとの連携に焦点を当てた方式であり、任意のレイヤ間での情報交換を行うことはできない。

## 4. リンク情報を利用した効率的 TCP 再開方式

現在の TCP は、通信中の有効帯域を正確に推定し、最適なウィンドウサイズを決定する機能がない。そのため無線ネットワークでハンドオーバが発生し、パケットロスが生じた場合

TCP はネットワークの許容帯域を超えたものと判断し、ウィンドウサイズを減少させ輻輳制御を行う。その結果、無線ネットワーク環境において必要以上にスループットが低下してしまう。また、パケットロスを減少させるためにはハンドオーバを高速に行い、切断時間を短くすることも重要である。

そこで本研究では、ハンドオーバ時のスループットの向上を図るために、TCP 層のみで動作するのではなく、リンク情報を利用してハンドオーバ前のウィンドウサイズをそのまま保有して通信を再開する手法を提案する。この手法を実現するには、ハンドオーバ後の帯域情報の取得、TCP レイヤへの帯域情報の通知、帯域情報を利用した TCP の再開処理の三つの機能が必要である。

#### 4.1 ハンドオーバ後の帯域情報の取得

提案方式では、アクセスルータ (AR) の改良により End-to-End の経路の情報と無線リンクの有効帯域情報をハンドオーバ前後の MN に送信する。これによってハンドオーバ後の MN は End-to-End 間の帯域の変化を取得することが可能となる。まず、モバイルノード (MN) は移動先の経路を知るために AR から IP アドレスを取得し、移動前後の AR の IP アドレスを比較することにより移動前後の通信経路の相違の程度を判断する。ハンドオーバ前後の End-to-End の経路は以下の 3 種類に分類できる。

- (a) ハンドオーバ前後で AR が同一の場合
- (b) ハンドオーバ前後で AR の 1 つ上流のルータが共通の場合
- (c) 経路がまったく異なる場合

次に、それぞれの経路において帯域情報を取得する方法を説明する。

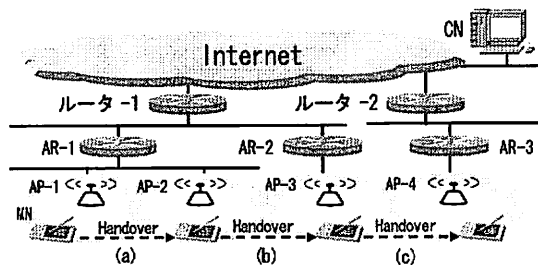


図 1 ハンドオーバ時の経路の分類

図 1(a) はハンドオーバ前後で AR が同一の場合を示している。ここでは AR の下に 2 台のアクセスポイント (AP) があり、MN が AP-1 から AP-2 にハンドオーバする場合を考える。ハンドオーバ前後で、送信相手 (CN) から AR-1 間の経路は変化しないが、AR-1 から MN 間の経路が変化する。したがって、MN は AP-2 の無線リンクの有効帯域情報を移動先の AR-1 から受信することで、ハンドオーバ後の End-to-End 間の有効帯域の取得が可能になる。図 1(b) は MN がハンドオーバ前後で AR の一つ上流のルータが共通の場合を示している。AR の下には AP が 1 台がある。MN が AP-2 から AP-3 にハンドオーバする際、CN からルータ-1 間の経路は変化しないが、ルータ-1 から MN 間の経路は変化する。したがって、MN は AP-3 の無線リンクの有効帯域情報を移動先の AR-2 から受信することで、ハンドオーバ後の End-to-End の有効帯域の取得が可能となる。図 1(c) は移動前後で経路がまったく異なる場合を示している。MN は AP-3 から AP-4 にハンドオーバする前後では、CN から MN 間の End-to-End の経路が変化する。したがって、MN は有効帯域情報を得ることが困難である。

タ-1 から MN 間の経路は変化する。したがって、MN は AP-3 の無線リンクの有効帯域情報を移動先の AR-2 から受信することで、ハンドオーバ後の End-to-End の有効帯域の取得が可能となる。図 1(c) は移動前後で経路がまったく異なる場合を示している。MN は AP-3 から AP-4 にハンドオーバする前後では、CN から MN 間の End-to-End の経路が変化する。したがって、MN は有効帯域情報を得ることが困難である。

#### 4.2 TCP レイヤへの帯域情報の通知

現在の TCP はネットワークが接続されたという情報とハンドオーバ後の End-to-End 間の帯域情報を TCP 自身で検知することができないため、ハンドオーバ後無駄な待ち時間が発生するなどの原因で TCP のスループットが悪化する。これに対し、下位層の接続完了を TCP に通知することにより待ち時間を短縮することが可能である。また、通信相手にハンドオーバ後の有効帯域情報を通知することによって End-to-End の TCP の効率的な再開処理が可能となる。

本研究では、レイヤ間情報交換機構 (LIES) [7] により、リンク層情報を抽象化し、TCP 層で下位層の情報を利用する。さらに通信相手への有効帯域情報の通知は TCP のオプションを利用する。図 2 は提案手法における LIES と TCP オプションによるハンドオーバ時のリンク層情報と帯域情報の流れを示す。図 2 において、L2、L3、L4 は MN のリンク層、ネットワーク層、トランスポート層であり、CN-L4 は通信相手のトランスポート層を示す。

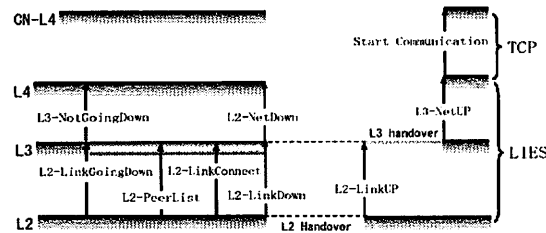


図 2 ハンドオーバ時レイヤ間の通知メッセージ

MN が現在接続しているリンクが、電波強度の低下などにより切断間近であるという切断予測通知 (L2-LinkGoingDown) がリンク層 (L2) からネットワーク層 (L3) に送られる。これを受けた L3 は、まず自身の L3 ハンドオーバ準備を開始し、さらにネットワーク切断予測 (L3-NetGoingDown) を生成して、TCP 層 (L4) に通知する。これを受けた TCP はハンドオーバの開始を予測し準備を行う。L3 は、自身の L3 ハンドオーバ準備処理が完了した時点で、L2 に指定リンクへの接続指示 (L2-LinkConnect) を送る。このような L3 による接続指示を行うため直前に L2-PeerList により近隣の Peer 情報を取得しておく。接続指示を受けた L2 は接続リンクを切断し、L2 ハンドオーバを開始する。これと同時にリンク切断通知 (L2-LinkDown) を L3 に送る。リンク切断通知 (L2-LinkDown) を受けた L3 は、ネットワーク切断通知 (L3-NetDown) を生成し、L4 に送る。TCP はこれを受けて、ハンドオーバ最中の無駄な TCP による送受信を止める。L2 ハンドオーバが完了しリンクが接続さ

れると、リンク接続情報(L2-LinkUP)がL3に通知される。これを受けてL3はL3ハンドオーバーを行い、ネットワーク接続通知(L3-NetUP)をL4へ送信する。そして、MNのL4は送信相手(CN)にTCPオプションを用いて帯域情報と通信開始(start Communication)の指示を送信する。

#### 4.3 帯域情報を利用したTCPの再開処理

MNとCNは取得した有効帯域情報を利用してハンドオーバー後のTCPの最適なウィンドウサイズを決定する。これによりTCPの無駄な輻輳制御によるスループットの低下を防止する。ウィンドウサイズ決定の手法はハンドオーバー前後の帯域の大小によって2種類に分類することができる。まず、ハンドオーバー後の帯域がハンドオーバー前の帯域と比べて、等しいか、それ以上であった場合、式(1)のようにハンドオーバー前のスロースタート閾値を利用して通信を行う。

$$cwnd = ssthresh \quad (1)$$

次に、ハンドオーバー後に帯域が狭くなった場合には、式(2)を使ってウィンドウサイズを決定する。式(2)では、取得した有効帯域情報(Bandwidth)とハンドオーバー前から保持しているRTTを利用して、ウィンドウサイズとスロースタート閾値を設定する。TCPのウィンドウサイズは経路上の帯域と遅延の乗算値である帯域遅延積である。

$$cwnd = ssthresh = Bandwidth * \frac{RTT}{2} \quad (2)$$

このように下位層の帯域情報を用いることで、TCPの通信効率の低下を回避することができる。

## 5. 実装

提案方式はモビリティプロトコルLIN6の上に実装し、オペレーティングシステムとしてはFreeBSD5.4-RELEASEを使用した。提案手法にはハンドオーバー後の有効帯域情報の取得、TCPレイヤへの有効帯域情報の通知、有効帯域情報を利用したTCPの再開処理の三つの機能を実装する必要がある。ハンドオーバー後の有効帯域情報の取得は近隣ネットワーク情報の取得とハンドオーバー前後の有効帯域の計算に分けられる。近隣ネットワーク情報の取得はユーザ空間におけるRA送信デモンの拡張により実装し、ハンドオーバー前後の有効帯域の計算はMNのカーネルにおけるRA受信部分の拡張により実装した。TCPレイヤへの帯域情報の通知はLIESを拡張し、MNのカーネル内に実装した。また、帯域情報を利用したTCPの再開処理はMNとCNのカーネル内のTCPのプロトコルスタックを改良して実装した。近隣ネットワーク情報の取得とハンドオーバー前後の有効帯域の計算についての詳細は次節以降で説明する。

#### 5.1 近隣ネットワーク情報の取得

提案方式では、RA(Router Advertisement)メッセージを拡張し、近隣ネットワーク情報オプションを新たに定義することで、MNがハンドオーバー前に近隣ネットワーク情報を取得することを可能とした。RAのオプションは、図3に示すような3つの情報を含む。これらの情報は事前にARごとに設定される。MNはARの下流IPアドレスの値を調べることで移動前

後のARが同一であるかを判断する。ARは定期的にRAを送信し、近隣ネットワーク情報をMNに知らせる。MNは移動前後のARが同一であることを確認するために移動前後でARの下流のIPアドレスの値を検査し、同じであればARが同一場合であるとする。またMNが接続するARの上流のIPアドレスを検査し、ネットワークプレフィックスが同じであればARの1つ上流のルータが共通であると判断する。ネットワークプレフィックスが異なる場合は経路が全く異なる場合と判断する。CNからMN間の最小の帯域がハンドオーバー後のウィンドウサイズとスロースタート閾値を決定するために利用される。

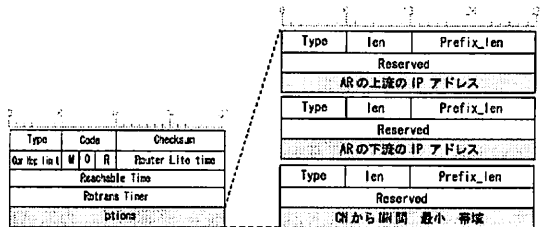


図3 拡張したRAのオプションのメッセージフォーマット

#### 5.2 ハンドオーバー前後の有効帯域の計算

従来のTCPではハンドオーバーに対する特別な処理を行わず、再送タイムアウトが発生すると一律にスロースタートフェーズからデータ転送を開始する。このように従来のTCPはハンドオーバー時にスロースタートを繰り返しながら通信を行う。一方、提案方式では、近隣の帯域情報を取得することによって、ハンドオーバー後のTCPの挙動を最適化することが可能である。図4はハンドオーバー後の輻輳制御方法を決定するアルゴリズムを示す。

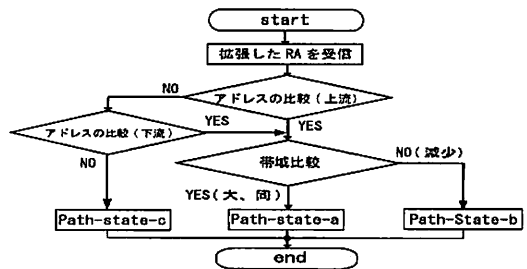


図4 ハンドオーバー後の輻輳制御方法の決定アルゴリズム

MNの移動前後でARが同一の場合またはARの1つ上流のルータが共通の場合は移動先の有効帯域によって動作が異なる。MNは移動先の帯域がハンドオーバー前の帯域と比べて大きいまたは等しい場合はハンドオーバー前のスロースタート閾値を利用して通信を行う。ハンドオーバー前に設定されたスロースタート閾値は、ハンドオーバー前のEnd-to-Endの帯域に最適な値にセットされている。そのために移動先の帯域が同じである場合、輻輳回避フェーズで通信を継続することが可能である。輻輳フェーズで通信を行うことによって、すでに存在していたTCPのフローとの公平性も確保できる。移動先の帯域がハン

ドオーバ前より小さい場合はハンドオーバ後の帯域の 1/2 に設定する。ハンドオーバ後の帯域の 1/2 以下に設定する場合は輻射を悪化させる可能性が高いためである。次に、MN の移動後にネットワーク経路が異なる場合は有効帯域を正確に推測し、最適なウィンドウサイズを決定することが困難である。この場合は通常の TCP と同じくスロースタートフェーズから輻射回避フェーズへ移行する。

### 5.3 無線環境エミュレータ

ハンドオーバ時の TCP の性能を測定するため、無線環境エミュレータと呼ばれるソフトウェアを利用した。無線環境エミュレータは多数のネットワークインタフェース (NIC) をもつ PC 上で動作する。1 つの NIC には移動ノードを接続し、他の NIC には AR を接続する (図 5 参照)。無線環境エミュレータはシナリオにしたがって MN の接続先の AP を切り替えたり、無線リンクの電波強度を変化させたり、無線リンクにおけるパケットロス率を確率的に発生させたりすることができる。

## 6. ハンドオーバ時の TCP の性能評価

本章では、提案方式における、MN がハンドオーバした時のレイヤ間情報交換の効果、帯域情報取得の効果を評価する。そしてそれらを用いた提案方式によるスループットを評価する。

### 6.1 評価環境

ハンドオーバが頻繁に発生する環境において、TCP のスループットがどのように影響を受けるかを評価した。無線ネット

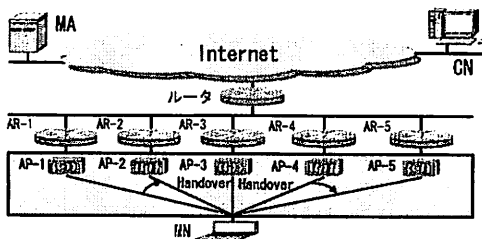


図 5 無線環境エミュレータを利用した実験環境

ワークでは雑音や電波の干渉など、様々な要因でネットワークの状況が変化してしまうため、実環境で評価を行うのは困難である。そのため本研究では無線環境エミュレータを用いて性能を測定した。無線環境エミュレータは前節で説明したように、AP の切り替えや帯域の設定などが可能である。図 5 に実験ネットワークを示す。MN が車で移動しながらファイルをダウ

パラメータ名	値
ファイルサイズ	30 Mbyte
RTT	20 / 50 / 100 msec
通信切断時間	0.1 / 0.5 / 1 / 2 sec
ハンドオーバ間隔	6 sec

表 1 パラメータ値

ンロードする場合を想定し、表 6.1 に示すパラメータを用いて評価した。車の移動の速度を約 60 km/h とする。さらに、無線通信方式としては無線 LAN を想定し AP の電波が届く範囲を直径 100m とする。この場合、MN が AP の通信可能範囲を

通過するために必要な時間は 6 秒となる。また、MN と CN 間の RTT とハンドオーバによる切断時間も変化させる。評価では 3 種類の実装における TCP の挙動を比較した。改良前の既存の TCP、LIES によるハンドオーバ検知のみの改良、本提案方式による改良の 3 種類である。

### 6.2 レイヤ間情報交換の効果

従来の TCP はハンドオーバ終了を自分自身で検知することができないため、再送タイムアウト時間を無駄に待たなければならない。この問題を、リンク接続完了通知およびネットワーク接続完了通知の下位レイヤ情報を用いて解消できることを確認する。図 6 は RTT が 10ms、ハンドオーバの切断時間が

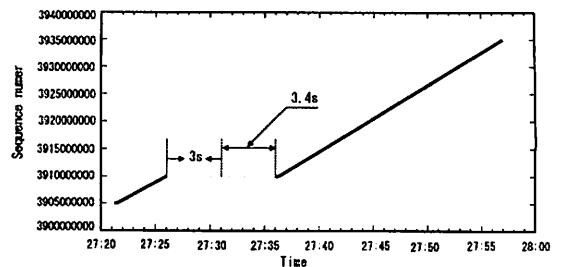


図 6 改良前の既存の TCP のシーケンス番号

3 秒のときの、既存の TCP によるシーケンス番号の変化を示している。縦軸は TCP のシーケンス番号、横軸は時間である。既存の TCP では 3 秒間の切断時間の後に送信再開まで 3.4 秒かかったことが分かる。図 7 は本提案方式のシーケンス番号の

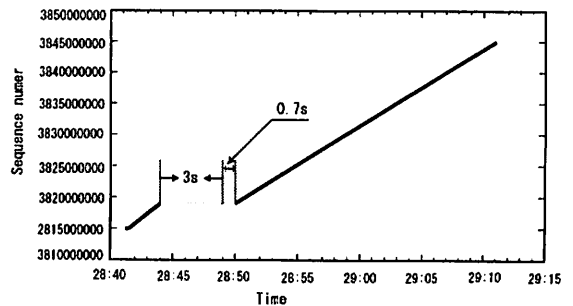


図 7 提案方式のシーケンス番号

変化である。本提案方式では 3 秒間の切断時間の後に送信再開までの時間が 0.7 秒で済んでいる。既存の TCP ではハンドオーバ完了後タイムアウトを待ってから通信を再開しているのに対し、本提案方式ではタイムアウトを待たずに通信を再開できたことが確認できる。

### 6.3 帯域情報取得の効果

モバイルノードの移動前後の有効帯域が変化しない場合の実験を行い、リンク層情報を取得した方式と既存の TCP の輻射ウィンドウの変化を図 8 に、リンク層から帯域を取得した方式の輻射ウィンドウの変化を図 9 に示す。図 8、図 9 の評価は RTT が 10ms、切断時間が 3sec 場合の結果である。図 8、図 9 の縦軸は Outstanding Data、横軸は時間である。既存の TCP

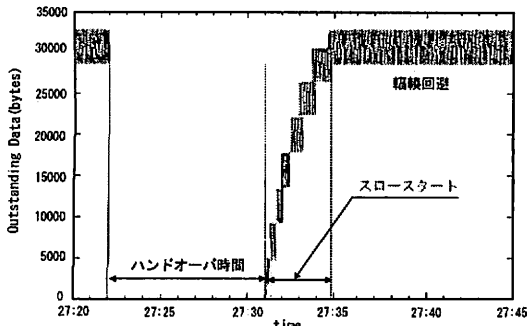


図 8 既存の TCP の Outstanding data の変化

ではハンドオーバー後スロースタートから通信を開始する。一方、リンク層から帯域を取得した方式では、スロースタートせずに移動前と同じ輻射回避フェーズで通信を再開する。したがって、効率的な通信が可能である。

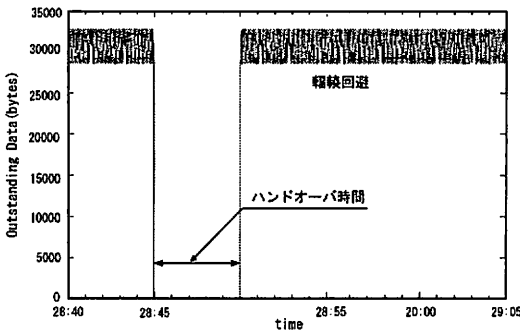


図 9 提案方式における Outstanding data の変化

#### 6.4 スループットの評価

図 10 は MN から CN へファイル転送した場合の、改良前の既存の TCP、LIES による改良、本提案方式のスループットの比較である。図 10 の縦軸はスループット、横軸は RTT と切断時間の組合せである。LIES による改良ではスループットが最大 48% 向上したことが分かった。

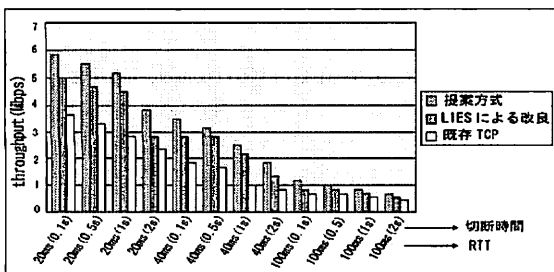


図 10 MN 側のスループットの比較

本提案方式ではスループットが最大 68% 向上したことが分かった。図 11 は、CN から MN へファイル転送した場合のスループットの比較である。MN の場合と同様に LIES による改良の

スループットが最大 50%、本提案方式による改良のスループットは最大 67% 向上したことが分かった。両方の場合とも RTT が 40ms のときに正常なハンドオーバーの回数が最も多く、スループットの向上が大きいことが分かった。

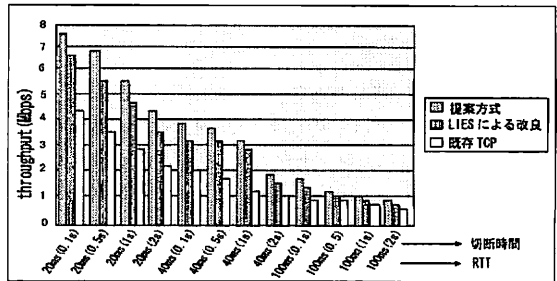


図 11 CN 側のスループットの比較

## 7. まとめ

本研究はハンドオーバーが TCP のスループットに与える影響を分析し、効率のよい輻射制御方式を提案した。また、ハンドオーバー時の通信再開の遅延解消について実験を行った。提案方式では、ハンドオーバー前後で MN と CN 間の経路がほぼ同一の場合、ハンドオーバー後は輻射制御から通信を再開する。また、下位層のハンドオーバーの完了を TCP に通知することにより、ハンドオーバー時に発生する通信再開の遅延解消を確認した。その結果、提案方式は既存の TCP より、高いスループットを示し、効率のよい通信が可能であることが分かった。

## 文 献

- [1] C. Perkins, D. Johnson, and J. Arkko. Mobility Support in IPv6. RFC 3775, IETF, June. 2004.
- [2] Masahiro Ishiyama, Mitsunobu Kunishi, Keisuke Uehara, Hiroshi Esaki, and Fumio Teraoka. LINA: A New Approach to Mobility Support in Wide Area Networks. *IEICE Transactions on Communication*, Vol. E84-B, No. 8, pp. 2076-2086, August 2001.
- [3] Mitsunobu Kunishi, Masahiro Ishiyama, Keisuke Uehara, Hiroshi Esaki, and Fumio Teraoka. LIN6: A New Approach to Mobility Support in IPv6. In *Proceedings of the Third International Symposium on Wireless Personal Multimedia Communications*, November 2000.
- [4] Tom Goff, James Moronski, Dhananjay S. Phatak, and Vipul Gupta. Freeze-TCP: A True End-to-End TCP Enhancement Mechanism for Mobile Environments. In *Proceedings of INFOCOM 2000*, pp. 1537-1545, Mar. 2000.
- [5] E. Hernandez M. Chinta, A. Helal. ILC-TCP: An Interlayer Collaboration Protocol for TCP. In *Proceedings of Third IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1004-1010, Mar. 2003.
- [6] Arjan Peddemors, Hans Zandbelt, and Mortaza Bargh. A mechanism for host mobility management supporting application awareness. In *MobiSys '04: Proceedings of the 2nd international conference on Mobile system, applications, and services*, pp. 231-244, Oct. 2004.
- [7] K. Gogo, R. Shibui, and F. Teraoka. An L3-Driven Fast Handover Mechanism in IPv6 Mobility. In *Proceedings of SAINT 2006 Workshops*, Jan. 2006.