

# TCP-J : 無線ネットワーク環境に適したトランスポートプロトコル

佐藤 徳彦<sup>†</sup>, 國司 光宣<sup>†</sup> 寺岡 文男<sup>†,††</sup>

本稿では、新しいトランスポートプロトコル TCP-J を提案する。TCP-J は、TCP の問題点である無線環境におけるスループットが低下を防止し、ネットワークの移動（ハンドオフ）によるネットワークリソース（バンド幅）の変化に適応した輻輳制御を行う。この結果、無線区間のビット誤り率が  $4 \times 10^{-6}$  のとき、NewReno を基準とすると 191%、TCP-westwood を基準とすると 35% のスループットの向上が認められ、移動先のネットワークの状況を考慮した輻輳制御が可能になった。

## TCP-J: New Transport Protocol for Wireless Network Environments

NORIHIKO SATO,<sup>†</sup> MITSUNOBU KUNISHI<sup>†</sup> and FUMIO TERAOKA<sup>†,††</sup>

This paper describes a new transport protocol called TCP-J to improve TCP (Transmission Control Protocol) performance over wireless links. TCP-J complies with the end-to-end TCP design principle, TCP-J does not require interception of TCP packets at intermediate nodes. Furthermore, at TCP-J, fairness between other TCP streams was improved when movement (hand-off) of mobile node occurs. Consequently, when the bit error rate was  $4 \times 10^{-6}$ , the improvement in throughput became 191% (compared with NewReno), and 35% (compared with TCP-westwood).

### 1. はじめに

TCP は有線回線上での利用を想定して設計されたプロトコルであり、フロー制御、輻輳制御を行うことにより通信の信頼性を高めている。TCP では告知ウィンドウ (advertised window) を用いてフロー制御を行い、輻輳ウィンドウ (congestion window) を用いて輻輳制御を行っている。有線回線は通信回線が安定しているので、TCP の情報伝送単位であるセグメントの損失が輻輳以外の原因で発生する可能性が低い。そのため、セグメントの損失はネットワークに過剰な負荷がかかっており、輻輳になっていると考えることができる。したがって、TCP はセグメントの損失を輻輳と見なし、輻輳ウィンドウを狭めることによってデータ転送量を抑え、輻輳崩壊を回避する。

一方、無線回線は有線回線と異なり、雑音や干渉等による高いビット誤り率によりセグメントの損失が生じる。このような無線回線上でのセグメントの損失は

有線回線と異なり、輻輳とは無関係に生じるため、輻輳制御を行う必要はない。ところが従来の TCP はセグメントの損失を輻輳と判断するため、一律に輻輳制御を行うことで、データの転送量を抑えてしまい、結果として無線ネットワーク環境ではスループットを大幅に下げってしまうという問題が生じる。

また、IP 層での移動透過を実現するために LIN6<sup>1)</sup> や MobileIPv6<sup>2)</sup> が提案されており、ホスト間で移動前に確立していたセッションを継続して利用することが可能になったが、これらのプロトコルを利用した際の移動時（ハンドオフ時）のセグメントの損失も、従来の TCP では輻輳と見なしてしまい一律にデータの転送量を下げってしまう。また新しいネットワークに移動するとネットワークリソース（バンド幅）に変化が生じると考えられるため、移動に際しては輻輳制御を変更する必要があると考えられる。しかし従来の TCP では移動によるネットワークリソースの変化に対して何の制御も行っていない。

本稿では新しいトランスポートプロトコル TCP-J を提案する。TCP-J は、ビット誤り率の高いネットワークにおいて有効である SACK<sup>3)</sup> に現在有効なバンド幅を見積もる ABE (Available Bandwidth Estimation) 機構を組み込んだものである。TCP-J はセグメント損失時に、ABE 機構により見積もられたバ

<sup>†</sup> 慶應義塾大学大学院理工学研究所  
Graduate School of Science and Technology, Keio University

<sup>††</sup> ソニーコンピュータサイエンス研究所  
Sony Computer Science Laboratories, Inc.  
現在、ソニー株式会社  
Presently with Sony, Ltd.

ンド幅を用いてスロースタート閾値 ( ssthresh ) および輻輳ウィンドウ ( cwnd ) を決定することで, 不当にデータの転送量を減少させることを防ぐ. また, 移動によるネットワークリソースの変化にも着目し, 輻輳制御に変更を加える.

## 2. 関連研究

無線ネットワーク環境における TCP の性能改善に関する研究のほとんどは, 以下に示す 3 つの方式に分類できる<sup>4)</sup>.

- Split connection schemes ( コネクション分割型 )
  - I-TCP<sup>5)</sup>
  - A TCP profile for W-CDMA ( 3G wireless packet service )<sup>6),7)</sup>
- Proxy schemes ( プロキシ型 )
  - Snoop<sup>8),9)</sup>
  - WTCP<sup>10)</sup>
- End-to-End schemes ( エンドツーエンド型 )
  - SACK<sup>3)</sup>+ELN<sup>11)</sup>
  - TCP-probing<sup>12)</sup>
  - TCP-westwood<sup>13)</sup>

Split connection schemes は性質の異なる有線区間と無線区間で TCP コネクションを分割し, スループットを向上させようという方式である. また, Proxy schemes は無線基地局であるアクセスルータ ( AR ) においてセグメントをキャッシュしておき, 無線区間でセグメント損失が生じた場合には, キャッシュされたセグメントを再送することで, データの転送効率を向上させようとする方式である. Split connection schemes は一般的に実現が容易であるが, 有線区間と無線区間の速度差からスループットの向上が見込めない. また Proxy schemes はスループットの向上は見込まれるものの, 機構が非常に複雑であり, 実現が難しいという欠点がある. 加えて Split connection schemes, Proxy schemes はともに AR での TCP ヘッダの snoop を前提としているので, IP パケットのペイロードを暗号化する IPsec との併用が不可能であるという大きな問題をかかえている. 以上の考察から End-to-End schemes が最も好ましいといえる. なかでも, 従来 TCP ストリームとの公平性を保ち, スループットを向上させることのできる TCP-westwood が注目されている.

### 2.1 TCP-westwood

TCP-westwood では, 以下のようにしてバンド幅を見積もり, 高速リカバリフェーズ後の輻輳ウィンドウ ( cwnd ) とスロースタート閾値 ( ssthresh ) の値を

単純に半分にせずに, 求めたバンド幅から算出する. まず, 送信ホスト側で確認応答 ( ACK ) が受けとられたときの時刻を  $t_k$ , 送信したデータの大きさを  $d_k$  とすると, その時点でのバンド幅のサンプルは式 (1) となる.

$$b_k = \frac{d_k}{t_k - t_{k-1}} \quad (1)$$

$t_{k-1}$  は 1 つ前の ACK の到着時刻である. TCP-westwood では  $b_k$  をさらに式 (2) のように複雑な Tustin 近似を用いることで平滑化し, バンド幅  $\hat{b}_k$  を求めている.  $T$  はカットオフ周波数を表す.

$$\hat{b}_k = \frac{\frac{2T}{t_k - t_{k-1}} - 1}{\frac{2T}{t_k - t_{k-1}} + 1} \hat{b}_{k-1} + \frac{b_k + b_{k-1}}{\frac{2T}{t_k - t_{k-1}} + 1} \quad (2)$$

TCP-westwood では高速リカバリフェーズ後, 式 (2) で得られた  $\hat{b}_k$  と RTT の最小値 RTTmin の積をスロースタート閾値 ( ssthresh ) と輻輳ウィンドウの値として用いることで無線経路でのセグメントロスによる不必要なスループットの低下を抑制している. その結果, 無線環境において, 従来の TCP に比べスループットの向上を実現している. しかし, TCP-westwood には以下の問題点がある.

- Reno ベースである.
- バンド幅の予測機構が複雑である.
- 移動によって生じるネットワークリソース ( バンド幅 ) の変化を考慮していない.

## 3. TCP-J の設計

TCP-westwood の問題点をふまえ, 本稿で提案する TCP-J はビット誤り率の高いネットワークにおいて有効である SACK ( Selective ACKnowledgement ) をベースとする. そしてバンド幅の予測機構を再考し, シンプルな設計とすることで処理のオーバーヘッドについて配慮を行った ABE ( Available Bandwidth Estimation ) 機構を用いて cwnd と ssthresh を決定する. またハンドオフし, 新しいネットワークに移動することを考慮すると, 移動によりネットワークリソースに変化が生じるため, 移動に際しては輻輳制御を変更する必要があると考えられる. 本稿では, 移動によるネットワークリソース変化にも着目し輻輳制御を行うことで, モバイルコンピューティング環境に適応した総合的な TCP の設計を行う.

また TCP-J は処理のほとんどを End ホストで行うことにより, End-to-End のセマンティクス<sup>14)</sup>を保ち, IP ペイロードの暗号化を行う IPsec<sup>15)</sup>との併用も可能である. 以下で TCP-J の詳細について説明する.

### 3.1 従来 TCP の cwnd, ssthresh の制御

TCP-J は SACK をベースにしている．従来の TCP (SACK を含む) は ACK を受信するたびに輻輳ウィンドウを増加させ、セグメント損失が生じた後に cwnd および ssthresh を変化させる．ここで重要になるのがセグメント損失を生じた後の cwnd と ssthresh の制御である．セグメント損失は 3 つの重複 ACK もしくは、タイムアウトによって検出され、該当セグメントの再送が行われる．その後、従来の TCP では cwnd, ssthresh は式 (3), (4) のように設定される．

重複 ACK により生じるセグメント再送後の ssthresh と cwnd

$$\begin{aligned} \text{ssthresh} &= \text{cwnd} \times \frac{1}{2} \\ \text{cwnd} &= \text{ssthresh} \end{aligned} \quad (3)$$

タイムアウトによるセグメント損失検出後の ssthresh と cwnd

$$\begin{aligned} \text{ssthresh} &= \text{cwnd} \times \frac{1}{2} \\ \text{cwnd} &= 1 \end{aligned} \quad (4)$$

タイムアウトによるセグメント損失時には、ネットワークが重い輻輳状態にあると考えられるので、cwnd を 1 に設定することは正しいと考えられる．しかし、3 つの重複 ACK により生じるセグメント再送後に cwnd を半分に設定するのは、誤り率の高い無線ネットワーク上ではスループットを必要以上に低下させてしまう．

### 3.2 TCP-J の cwnd, ssthresh の制御

TCP-J では、誤り率の高い無線ネットワーク上でスループットの低下を防止するために cwnd, ssthresh を式 (5), (6), (7) に基づいて決定する．変数 BW は ABE (Available Bandwidth Estimation) 機構から求められるバンド幅であり、RTTmin はラウンドトリップタイム (RTT) の最小値を表す．ABE 機構については 3.3 節で説明する．

重複 ACK により生じるセグメント再送後の ssthresh と cwnd

$$\begin{aligned} & \text{if } (\text{ssthresh} \leq \text{cwnd}) \\ & \text{ssthresh} = \frac{(\text{BW} \times \text{RTTmin})}{\text{segment size}} \quad (5) \\ & \text{cwnd} = \text{ssthresh} \end{aligned}$$

$$\begin{aligned} & \text{elseif } (\text{ssthresh} > \text{cwnd}) \\ & \text{ssthresh} = \frac{(\text{BW} \times \text{RTTmin})}{\text{segment size}} \quad (6) \\ & \text{cwnd} \Rightarrow \text{keep recent value} \end{aligned}$$

タイムアウトによるセグメント損失検出後の ssthresh と cwnd

$$\begin{aligned} \text{ssthresh} &= \frac{(\text{BW} \times \text{RTTmin})}{\text{segment size}} \quad (7) \\ \text{cwnd} &= 1 \end{aligned}$$

式 (6), (7) に示すように、ABE 機構から求められた BW を用いて ssthresh, cwnd を決定することにより、セグメント損失が生じても不必要にスループットを低下させることを防止する．この TCP-J の輻輳制御では、有線区間のセグメント損失と無線区間のセグメント損失の区別を行わない．通常、無線区間のセグメント損失を検出し、無線区間のセグメント損失ならば cwnd を変化させないという手法が有効であると思われる．しかし、実際には通常の SACK 方式よりもスループットの向上が見られない．この原因としては、アクセスルータ (AR) でバッファオーバーフローが生じることが考えられる．つまり、無線区間でセグメントロスが生じた場合、データリンク層で再送が行われているが、データリンク層で再送を行ったにもかかわらず、セグメントが正常に到達しなかった場合、AR にバッファされるセグメントが増加する．さらに、送信側の TCP がセグメント再送後に輻輳ウィンドウを狭めず、AR のバッファがあふれるため、結果として通常の SACK 方式よりもスループットの向上が見られないものと考えられる．

したがって、本稿で提案する TCP-J では、セグメント損失を検出した場合には、軽い輻輳状態になっていると考え、AR への影響を考慮し、ABE 機構から求められた BW を用いて ssthresh, cwnd を決定する．

### 3.3 ABE (Available Bandwidth Estimation) 機構について

TCP-J では、以下のようにしてバンド幅を見積もり、高速リカバリフェーズ後の ssthresh の値を単純に半分にせずに、求めたバンド幅から ssthresh を算出する．まず、送信ホスト側が ACK を受信した時刻を  $t_k$ 、送信したデータの大きさを  $d_k$  とすると、その時点でのバンド幅のサンプルは式 (8) となる．

$$\text{BW}_{\text{sample}}(k) = \frac{d_k}{t_k - t_{k-1}} \quad (8)$$

$t_{k-1}$  は 1 つ前の ACK の到着時刻である．TCP では、いくつか ACK をまとめて返す遅延確認応答 (DelACK) の機能を有しており、BSD 系の実装では 200 msec 以内に受信した ACK を受信側ではまとめて送信側に返すことにより、帯域を節約している．そのため、ここで求めた  $\text{BW}_{\text{sample}}(k)$  にも揺らぎが生じる．TCP-J では得られた  $\text{BW}_{\text{sample}}(k)$  を式 (9) に

より平滑化することで有効なバンド幅  $BW_k (= BW)$  を算出する .

また  $\alpha$  は平滑化定数であり, 有線で TCP-J を利用した場合に際して, 従来 TCP との公平性を保つために Jacobson の平滑化式<sup>16)</sup> との比較から  $\alpha$  のおおよその値を求め, 設定値を変化させてシミュレーションを行った . その結果 0.9 が最も適切であったので以後,  $\alpha$  には 0.9 を適用する .

$$BW_k = \alpha \times BW_{k-1} + (1 - \alpha) \times \frac{BW_{sample}(k) + BW_{sample}(k - 1)}{2} \quad (9)$$

TCP-J では TCP-westwood の平滑化に用いている Tustin 近似式 (2) のように複雑な計算を行わないことで, 処理のオーバーヘッドに配慮している .

TCP-J では高速リカバリフェーズ後, ここで得られた  $BW_k (= BW)$  と RTT の最小値  $RTT_{min}$  の積を  $cwnd$  と  $ssthresh$  の値として用いることで無線経路でのセグメントロスによる不必要なスループットの低下を抑制している . その結果, 無線環境において, 従来の TCP に比べスループットの向上を実現している . ABE 機構の有効性は 4 章で示す .

### 3.4 ハンドオフ時の輻輳制御

無線環境では, 新しいネットワークに移動する場合も考慮する必要がある . IP 層での効率的ハンドオフ手法では, ハンドオフのコストを抑える工夫をすることで, セグメントの損失を最小限にとどめようと試みている . TCP-J ではこれらのハンドオフに起因するセグメント損失が生じた場合でも, ABE 機構により, 不必要に  $cwnd$  を狭めることはない . しかし, 移動し新しいネットワークに接続すると, ネットワークリソース (バンド幅) に変化が生じると考えられるため, 移動に際しては輻輳制御を変更する必要がある本節では TCP-J のハンドオフ時の制御について説明する .

IP 層での効率的ハンドオフ手法には様々な方式がある . HMIP<sup>17)</sup> のような IP 層での効率的ハンドオフ手法の利用を考えた場合, 移動ノード (MN) と固定ノード (CN) が通信を行っている際の典型的なトポロジは図 1 のようになる . HMIP に代表される Hierarchical 方式はドメインという範囲内の移動の管理を Regional Route Router (RR) に集約することで, ドメイン内の移動コストを削減している . そのため, ドメイン間の移動のコストは通常 MIP とほぼ同じであるが, ドメイン内の移動については通常 MIP よりも低いコストで移動することができる . ここでネットワークリソースの変化に着目した場合, 移動の種類は, 図 1 の 3 つ (A, B, C) に分類できる . また, 移動

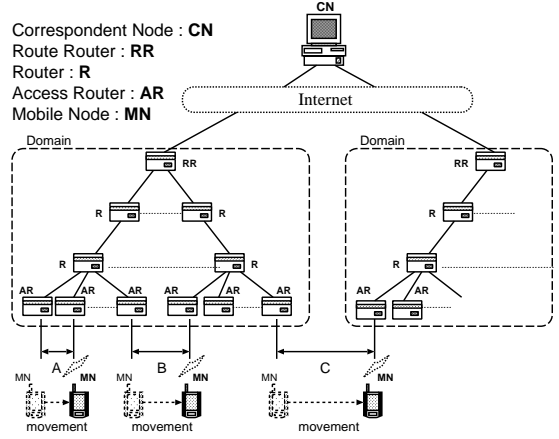


図 1 ハンドオフの種類

Fig. 1 Classification of handoff.

前後で上流ルータが共通であるか共通でないかによってさらにこれらは local movement (A) と non-local movement (B, C) に分類できる .

#### 3.4.1 Local movement

Local movement (移動 A) すなわち, 上流ルータが共通の場合, 有線区間のネットワークリソースも共通であると考えられる . したがって, local movement に起因するセグメント損失, もしくはタイムアウトが生じた場合,  $cwnd$  および  $ssthresh$  は式 (5), (6), (7) のようになると述べたが, 移動先の有線区間のネットワークリソースが共通であっても, 無線区間がボトルネックになることが考えられる . このため, 移動後, 新しいアクセスルータ (AR) で無線の帯域を MN に通知し, MN が通信ノードに無線区間の帯域を通知する . 通信ノードでは, 現在有効なバンド幅と MN から通知された無線区間の帯域を比較し, 無線区間の帯域の方が小さければそちらを現在のバンド幅とすることで  $cwnd$  と  $ssthresh$  を決定する . この無線区間の帯域通知により, 移動による無線区間のリソースの変化にも対応が可能となる . 移動端末から送信側に無線帯域を通知する方法としては IP 層での移動通知 (binding update) に無線帯域を付加する手法等が考えられる . また, この手法を実現するためには, AR でデータの送受信量を監視している必要がある .

#### 3.4.2 Non-local movement

Non-local movement (移動 B, 移動 C) すなわち, 上流ルータが共通でない場合, 移動によって経路が大きく変化するので, ネットワークリソースも大きく変化すると考えられる . したがって non-local movement が生じた場合は,  $ssthresh$  を広告ウィンドウサイズ (advertised window) とし明示的にスロースター

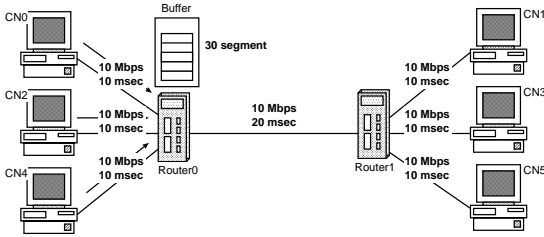


図2 トポロジ  
Fig.2 Topology.

トを行うと同時に、ABE 機構をリセットする。non-local movement の場合にも、無線区間のリソースが変化することが考えられるが、non-local movement が生じた際には明示的にスロースタートを行うため、無線区間の帯域通知により cwnd と ssthresh を決定する必要はない。

4. 評価

本章では TCP-J の評価を行う。無線環境における様々な問題を実世界で再現するのは困難であるため、本稿では TCP-J の評価を NS2 (ns-2.1b8a)<sup>[8]</sup> を用いたシミュレーションで行った。

本章では最初に従来 TCP ストリームと TCP-J の公平性の評価を行い、続 ABE 機構の評価を行う。そして無線環境下における TCP-J の評価を行い、最後にハンドオフの際の評価を行う。

4.1 従来 TCP との公平性の評価

従来 TCP との公平性の評価を行うために図2のようなトポロジのネットワークを想定しシミュレーションを行った。CN0 から CN1, CN2 から CN3, CN4 から CN5 へ TCP コネクションをそれぞれ確立し、FTP で連続してデータ転送を行うというシナリオで、それぞれの TCP コネクションを監視する。このシミュレーションでは、Router0 から Router1 の区間を3つの TCP コネクションで共有し、公平に帯域を利用できていることを確認する。

最新オペレーティングシステムでの TCP の実装は NewReno であり、SACK オプションもデフォルトで有効になっているので、まず CN0 から CN1 が NewReno, CN2 から CN3 は SACK, CN4 から CN5 が TCP-J の場合について評価を行う。このときの各パラメータを表1に示す。

セグメントサイズ、イニシャルウィンドウサイズおよび遅延確認応答の値は、BSD 系の実装のデフォルト値を使用した。また、ルータのバッファの大きさは製品によって様々である。本シミュレーションではバッ

表1 各パラメータ  
Table 1 Parameters.

| パラメータ名             | 値           |
|--------------------|-------------|
| セグメントサイズ           | 1,460 byte  |
| イニシャルウィンドウサイズ      | 65,535 byte |
| 遅延確認応答             | 200 msec    |
| Router のバッファサイズ    | 30 segment  |
| 平滑化定数 ( $\alpha$ ) | 0.9         |

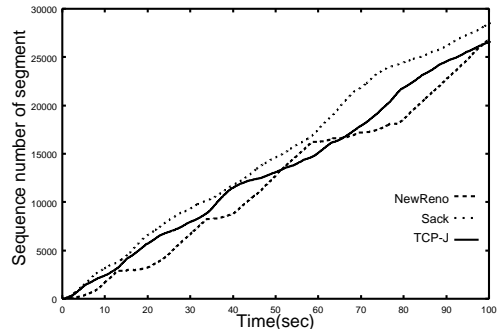


図3 NewReno, SACK, TCP-J 間の公平性について (セグメントのシーケンス番号)  
Fig.3 Fairness between NewReno and SACK and TCP-J.

ファオーバフローによる輻輳を条件として与えるため、収容するコネクションが3つであることを考慮し、Router0 のバッファをやや小さめのバッファサイズ (30 segment) に設定し、DropTail 型とした。

このときの受信セグメントのシーケンス番号を示したものが図3である。x 軸が時間、y 軸がセグメントのシーケンス番号を表す。図3から到着セグメントのシーケンス番号の増加の割合は、NewReno, SACK, TCP-J でほぼ同じであることが分かる。したがって有線環境では NewReno, SACK, TCP-J が混在する環境下で帯域が公平に使用されていることが分かる。

以上の検討から、TCP-J と従来 TCP ストリーム間では帯域を公平に使用することが示され、有線環境で TCP-J を利用した通信を行った際、従来 TCP ストリームには影響を与えないことが分かった。このため、TCP-J に移行する際に従来システムに対する特別な配慮が必要ないことが分かる。TCP の標準化の際には、従来 TCP との共存を重視し、スループットを向上させるだけの輻輳制御は認めない傾向にあるので、従来 TCP との間で公平性が保たれる TCP-J は標準化の際に大きな利点があると考えられる。

次に TCP-J の重要な機能である ABE 機構の評価を行う。

4.2 ABE 機構の評価

TCP-J では式 (8) により求められた  $BW_{sample}$  を式

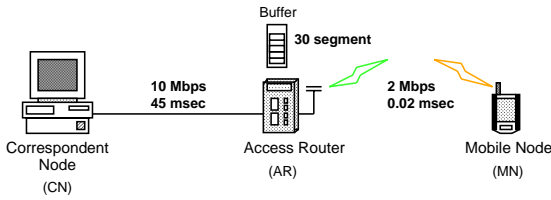


図4 トポロジ  
Fig. 4 Topology.

(9) で平滑化し、現在有効なバンド幅を求めている。一方、TCP-westwood では複雑な Tustin 近似 (式 (1)) を用いた平滑化を行うことにより、有効なバンド幅を求めている。本項では、スループットの実測値と ABE 機構により推測された、現在有効なバンド幅を比較することで、ABE 機構によるバンド幅の推測が有効であることを示す。また、本稿で提案する平滑化式 (8)、(9) を用いたバンド幅の推測結果と、Tustin 近似を用いたバンド幅の推測結果を比較することで、本稿で提案する平滑化方式は、複雑な Tustin 近似と比べ、ロコストであるにもかかわらず、ほぼ同程度の精度の推測が可能であることを示す。

まず、シミュレーションに用いるネットワークのトポロジを図 4 に示す。ここで用いたトポロジは無線と有線が混在するシンプルなたポロジである。ネットワークの各パラメータは実測値を用いて設定した。イニシャルウィンドウサイズ、遅延確認応答値は BSD 系実装のデフォルト値を用いた。

ビット誤り率 (BER: Bit Error Rate) は、比較的状态の良い  $10^{-8}$  と転送レートが極端に落ち始める  $10^{-6}$  についてシミュレーションを行い、スループットの実測値と ABE 機構によるバンド幅の推測値を比較した。また、無線の状況が突然悪化した場合を再現するために、50 sec から 0.01 sec の間、100 sec から 0.05 sec の間、150 sec から 0.1 sec の間それぞれ無線経路において通信が不通となっているものとする。また、ビット誤りはランダムに発生するものとする。

BER =  $10^{-8}$  におけるスループットの実測値と提案手法を用いた推測値の比較を図 5 に示し、BER =  $10^{-8}$  におけるスループットの実測値と Tustin 近似を用いた推測値の比較を図 6 に示す。図 5、6 から提案手法によるバンド幅の推測も Tustin 近似による平滑化を用いたバンド幅の推測手法もスループットの実測値とほとんど同じ結果となっていることが分かる。相違点としては、50 sec、100 sec、150 sec に通信不通により、タイムアウトが生じスループットが下がったときに、提案手法による推測値は Tustin 近似による平滑化を用いた推測値よりも高い値を示している、これ

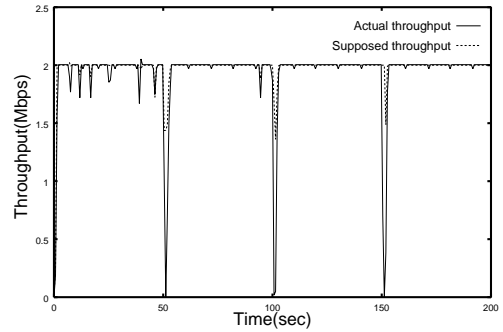


図5 BER =  $10^{-8}$  におけるバンド幅の実測値と推測値の比較 (提案手法)

Fig. 5 Comparison actual throughput with supposed throughput (BER =  $10^{-8}$ : Supposed technique).

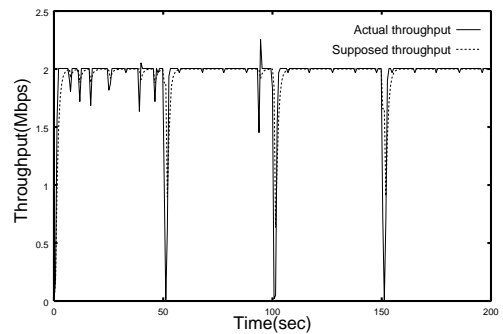


図6 BER =  $10^{-8}$  におけるバンド幅の実測値と推測値の比較 (Tustin 近似)

Fig. 6 Comparison actual throughput with supposed throughput (BER =  $10^{-8}$ : Tustin).

により、提案手法では  $sssthresh$  を多少高く設定するため、タイムアウトからのリカバリが Tustin 近似による平滑化を用いた手法よりも迅速に行われることが予想される。

次に転送レートが大きく変化する BER =  $10^{-6}$  におけるスループットの実測値と提案手法を用いた推測値の比較を図 7 に示し、BER =  $10^{-6}$  におけるスループットの実測値と Tustin 近似を用いた推測値の比較を図 8 に示す。図 7、8 の場合は BER が  $10^{-6}$  と高くなっており、スループットの変動が激しいことが分かる。この場合、提案手法によるバンド幅の推測と Tustin 近似による平滑化を用いたバンド幅の推測手法に多少違いが生じる。提案手法は、スループットの実測値を近似した値となっており、Tustin 近似による平滑化を用いた手法は、スループットを提案手法よりも忠実に推測している。しかし、スループットの極端な落ち込みは無線区間に起因するものであるため、忠実に再現する必要はなく、 $cwnd$ 、 $sssthresh$  を決定

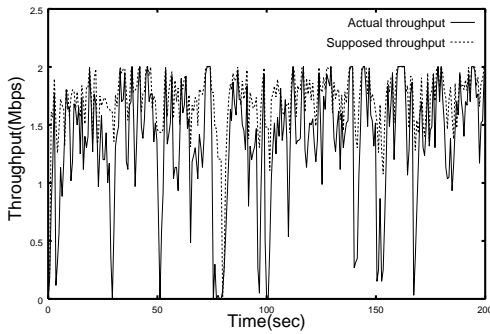


図7 BER =  $10^{-6}$  におけるバンド幅の実測値と推測値の比較 (提案手法)

Fig. 7 Comparison actual throughput with supposed throughput (BER =  $10^{-6}$ : Supposed technique).

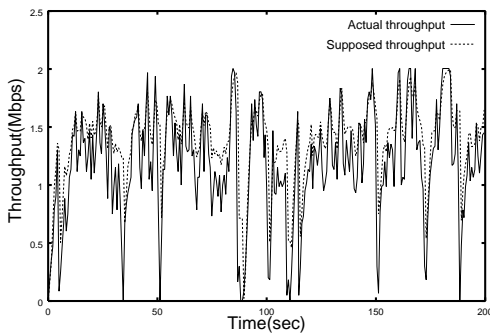


図8 BER =  $10^{-6}$  におけるバンド幅の実測値と推測値の比較 (Tustin 近似)

Fig. 8 Comparison actual throughput with supposed throughput (BER =  $10^{-6}$ : Tustin).

するうえでは、ある程度近似が行われ、変動を抑えられている方が好ましいといえる。以上から、本提案手法による ABE 機構はシンプルであるにもかかわらず cwnd, ssthresh を決定するのに十分な精度を持っていることが分かる。また BER =  $10^{-6}$  の例から、本提案手法は BER が大きくなりスループットの変動が大きくなった場合にも有効であることが分かる。

#### 4.3 無線環境における従来 TCP との比較

本節では無線環境において従来 TCP (NewReno, SACK, TCP-westwood) と TCP-J の性能比較を行う。まず、無線区間のランダムなビット誤りにより、セグメント損失がランダムに発生する場合の各 TCP の評価について述べ、次に、一定時間無線区間が不通になり、パースト的なセグメント損失が生じた場合の各 TCP の評価について述べる。

##### 4.3.1 ランダムなセグメント損失に対する評価

無線区間において、ランダムなビット誤りが生じる場合に従来 TCP と TCP-J の性能比較を行うために、

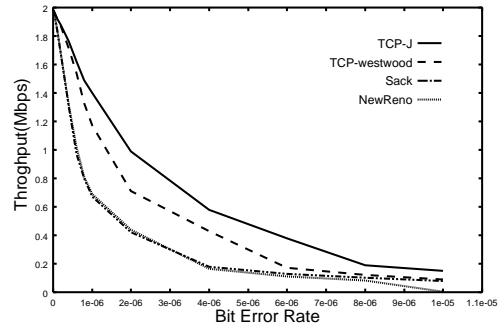


図9 各 TCP の BER に対するスループット  
Fig. 9 Throughput (random loss).

図 4 に示すシンプルなたポロジを構成しシミュレーションを行った。CN と AR 間は有線区間、AR と MN 間は無線区間という想定である。

まず AR と MN 間の無線の BER を  $10^{-8} \sim 10^{-5}$  の間で変化させたときの NewReno, SACK, TCP-westwood, TCP-J の各スループットを図 9 に示す。スループットは各 BER において 200 sec の間、FTP で連続してデータ転送を行っていた場合の測定値である。また、ビット誤りはランダムに発生するものとする。

図 9 の x 軸は BER で y 軸はスループットである。この図から NewReno, SACK はほぼ同程度のスループットを示していることが分かる。また、TCP-J はすべての BER の値において TCP-westwood を上回る高いスループットを示していることが分かる。NewReno を基準値とした場合の、各 BER におけるスループットの向上率は表 2 となる。

また、表 2 から、無線環境に TCP-J を適用した場合、NewReno を基準として、最大 191% のスループットの向上が認められた。

##### 4.3.2 パースト的なセグメント損失に対する評価

本節では、無線の状況が悪化し一定時間 (0.01 sec ~ 1 sec) 不通になり、パースト的なセグメント損失が生じた場合について、各 TCP のスループットを測定する。トポロジおよび各パラメータは、ランダムなビット誤りが生じる場合の TCP の性能比較に用いたもの同一とする (図 4, 表 1)。スループットは、10 sec の間、FTP でデータの連続転送を行っている最中に、一定時間 (0.05 sec ~ 1 sec) 無線区間が不通になるという想定で不通になる時間を変化させて測定する。また無線区間のビット誤り率は、各 TCP 間でスループットの差が少ないという理由から  $10^{-8}$  とした。結果を図 10 に示す。

図 10 から、TCP-J は一定時間 (0.01 ~ 1 sec) 無線

表2 スループット向上率  
Table 2 Improvement of throughput.

|          | ビット誤り率 (BER) |                    |                    |                    |           |                    |                    |                    |
|----------|--------------|--------------------|--------------------|--------------------|-----------|--------------------|--------------------|--------------------|
|          | $10^{-7}$    | $2 \times 10^{-7}$ | $4 \times 10^{-7}$ | $8 \times 10^{-7}$ | $10^{-6}$ | $2 \times 10^{-6}$ | $4 \times 10^{-6}$ | $8 \times 10^{-6}$ |
| SACK     | 0%           | 0%                 | 0.77%              | 0.98%              | 6.2%      | 4.5%               | 10%                | 20%                |
| Westwood | 4.3%         | 11%                | 32%                | 64%                | 41%       | 61%                | 15%                | 20%                |
| TCP-J    | 4.3%         | 12%                | 35%                | 84%                | 103%      | 125%               | 191%               | 90%                |

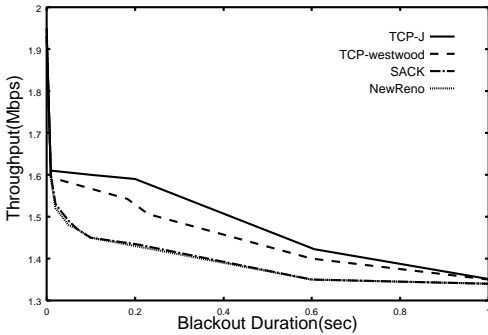


図10 パースト的なセグメント損失が生じた際のスループット  
Fig. 10 Throughput (burst loss).

区間が不通になった状況からのリカバリが迅速であり、高いスループットが得られていることが分かる。また、この結果からハンドオフの際に一定時間通信が中断した場合にも高いスループットを得られることが分かる。ところでパラメータの1つである  $\alpha$  値によりバンド幅の推定値は大きく変動すると考えられるが、 $\alpha$  値は従来 TCP との公平性を主眼に設定した値であり、 $\alpha$  値により積極的にスループットを向上させているのではない。スループットの不要な落ち込みを帯域の推定値に反映していないことが Tustin 近似と本提案手法 (ABE 機構) の大きな違いであり、スループットを高く保つことのできる要因となっている。

本項では、無線区間のランダムなビット誤りにより、セグメント損失がランダムに発生する場合の各 TCP の評価について述べ、次に、一定時間、無線区間が不通になり、パースト的なセグメント損失が生じる場合の各 TCP の評価について述べた。本項では、無線環境において従来 TCP と TCP-J の性能比較を行い、TCP-J が高いスループットを得ることができ、有効であることが分かった。

4.4 ハンドオフの評価

本節では local movement と non-local movement に分けてハンドオフの評価を行う。

TCP-J のハンドオフの際のスループットに関しては、ハンドオフにより、どの程度の時間、通信が不通になるかによって変化する。通信が一定時間、不通になったときのスループットの比較に関しては 4.3.2 項

表3 ハンドオフにかかるコスト  
Table 3 Cost of handoff.

|       | コスト (msec)        |
|-------|-------------------|
| ドメイン内 | 50+30+RR までの RTT  |
| ドメイン間 | 50+30+通信相手までの RTT |

で示したとおりであり、TCP-J は従来の TCP よりも高い値を得られることが分かった。本節ではハンドオフの際のスループットの評価は除き、ハンドオフにより、新しいネットワークに移動した際に与える影響について評価する。

ハンドオフの評価で用いる共通のパラメータを表1に示す。

セグメントサイズ、イニシャルウィンドウサイズおよび遅延確認応答の値はこれまでの評価と同様に、BSD 系の実装のデフォルト値を使用した。またアクセラータのバッファも、これまでの評価と同様、大きさを 30 セグメントに設定し、DropTail 型とした。まず、最初に local movement について評価を行い、最後に non-local movement について評価を行う。

また、ハンドオフにかかるコスト (時間) は、データリンク層とネットワーク層に依存する。本稿では、データリンク層は 802.11b、ネットワーク層は HMIP 等のマイクロモビリティプロトコルの利用を想定する。802.11b (Wave-LAN) のハンドオフにかかるコストは、ビーコンの間隔から算出できる。802.11b におけるビーコン間隔のデフォルト値は 100 msec<sup>19)</sup> であるので 802.11b のハンドオフにかかるコストは期待値から 50 msec となる。また、ネットワーク層のハンドオフのコストはマイクロモビリティの研究により、次のように設定した。通信ノード間の RTT を  $X$  msec、Regional Route Router (RR) と MN 間の RTT を  $Y$  msec とした場合、ドメイン内移動のコストを  $X + 30$  msec、ドメイン間移動のコストを  $Y + 30$  msec とした。したがって、ハンドオフにかかるコストは表3のようになると考えられる。

本稿では、表3の計算結果から求められたハンドオフのコストをシミュレーションのパラメータとし評価を行う。



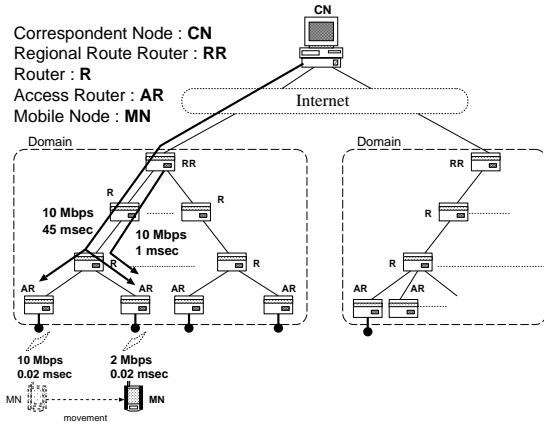


図 11 Local movement のトポロジ  
Fig. 11 Topology of local movement.

4.4.1 Local movement の評価

Local movement は 4 章で述べたように、上流ルータが共通の場合のハンドオフを指す。つまり local movement の場合、有線区間のネットワークリソース(バンド幅)は共通であると考えられる。しかし、無線区間のネットワークリソースがボトルネックになることも予想される。したがってここでは、無線区間のネットワークリソースが異なり、ボトルネックになる場合の local movement の評価を行う。トポロジおよびネットワークパラメータを図 11 に示す。

図 12 の x 軸は時間であり、y 軸は cwnd の値である。図 12 により、100 sec 時にハンドオフし、バンド幅の狭いネットワークに移動したため、cwnd の値が小さい値に落ち着いているのが分かる。また、同じ条件で TCP-westwood を使用した際の cwnd の変化を図 13 に示す。図 13 から、TCP-westwood はハンドオフ(100 sec 時)により移動した後、無線区間がボトルネックになっていることを開知できないため、cwnd を増加させていることが分かる。この cwnd の増加は、他 TCP ストリームに悪影響を与え、輻輳を生じる危険性がある。一方、図 12 より TCP-J は、AR から無線区間のバンド幅の通知を受けているので、cwnd を不必要に増加させていないことが分かる。この動作の違いは、スループットには直接現れないが、他 TCP ストリームとの公平性という点で TCP-J が優れていることを示している。

4.4.2 Non-local movement の評価

Non-local movement は 4 章で述べたように、上流ルータが共通でない場合のハンドオフを指す。つまり、non-local movement の場合、有線区間、無線区間ともに、ネットワークリソース(バンド幅)が変化

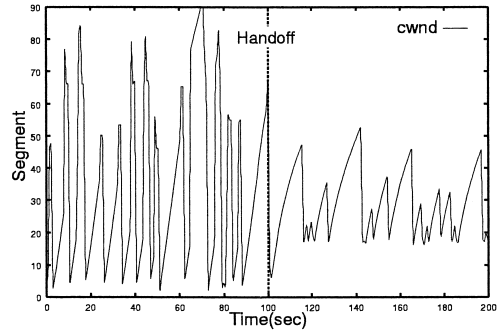


図 12 Local movement (TCP-J: 輻輳ウインドウの変化)  
Fig. 12 Local movement (TCP-J: cwnd).

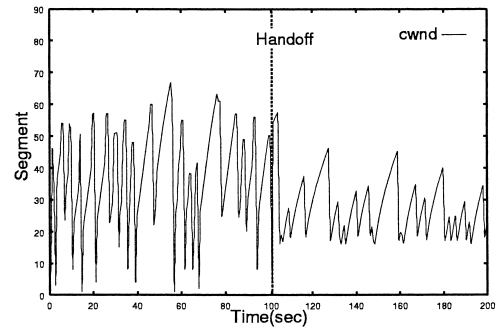


図 13 Local movement (TCP-westwood: 輻輳ウインドウの変化)  
Fig. 13 Local movement (TCP-westwood: cwnd).

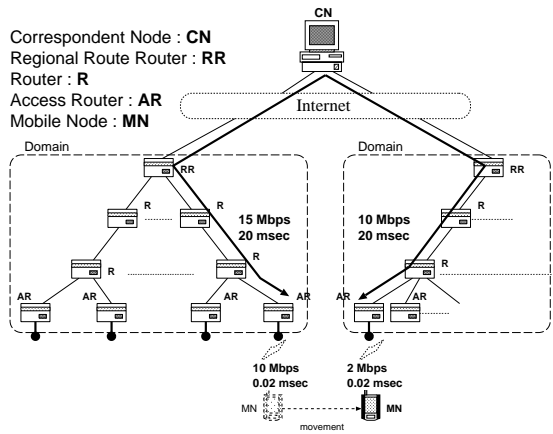


図 14 Non-local movement のトポロジ  
Fig. 14 Topology of non-local movement.

すると考えられる。本項では、ドメイン間の non-local movement について評価を行う。

ここでは、移動前のネットワークリソースにくらべ、移動後のネットワークリソースが小さい場合の non-local movement (ドメイン間) について評価を行う。トポロジおよびネットワークパラメータを図 14 に示

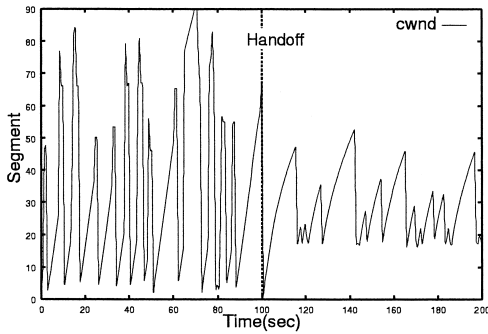


図 15 Non-local movement (TCP-J: 輻輳ウィンドウの変化)  
Fig. 15 Non-local movement (TCP-J: cwnd).

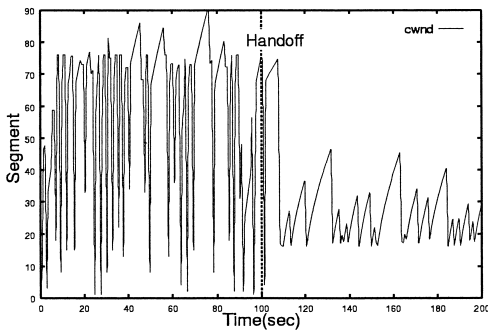


図 16 Non-local movement (TCP-westwood: 輻輳ウィンドウの変化)  
Fig. 16 Non-local movement (TCP-westwood: cwnd).

す。また、ハンドオフにかかるコストはドメイン間であるので、表 3、図 14 から 120 msec と求められる。また、無線区間は状態が良いものとして、ビット誤り率を  $10^{-7}$  とした。また FTP で 200 sec 間データを連続転送し、100 sec 時にハンドオフを行う。このときの TCP-J の cwnd の変化を図 15、TCP-westwood の cwnd の変化を図 16 に示す。図 15、16 の比較により、ハンドオフ直後 TCP-westwood は cwnd を大きく増加させており、他 TCP ストリームに対して悪影響を及ぼす可能性がある。一方、TCP-J は ABE 機構をリセットしているため、このような誤作動は生じていない。この結果からも、他 TCP ストリームとの公平性という点で TCP-J が優れていることが分かった。

#### 4.4.3 ハンドオフの帯域通知が存在しない場合について

上述のようにハンドオフ時に移動先の帯域通知が可能な場合には、移動先のネットワークリソースを考慮したハンドオフが可能になる。しかし移動先の AR が帯域通知に対応していない場合、特に移動先のネットワークの帯域が狭い場合にスループットを不必要に増加させてしまい移動先のネットワークを圧迫する可能

性がある。この危険性を回避するためには、ハンドオフの際、移動先の AR が帯域通知に対応していない場合に ABE 機構をリセットし明示的にスロースタートを行うという手法が考えられる。この場合、移動先の AR が帯域通知に対応していないとすべてのハンドオフ時に明示的にスロースタートを行うので local movement 時、non-local movement 時はそれぞれ同様に図 15 と同じ cwnd 変化となる。しかしこのように移動先の AR が帯域通知に対応していない場合にも TCP-J は 4 章でも述べたように通常の TCP よりもスループットを高く保つことができる。

## 5. 結 論

本稿では TCP-J を NS2 で評価し、無線環境のようにビット誤り率の高いネットワークにおいてスループットが向上することを示した。また、TCP-J では従来研究では触れていないネットワークの移動(ハンドオフ)によるネットワークリソースの変化にも着目し、輻輳制御に変更を加えた。この結果、ネットワークの移動(ハンドオフ)の際に、他の TCP データストリームに影響を与えることなく、ネットワークリソースの変化に適応可能となった。さらに本稿では、TCP の標準化の際に最も問題とされる、従来 TCP データストリームとの公平性についても評価を行い、TCP-J は従来 TCP ストリームと公平性が保てることを示した。

## 参 考 文 献

- 1) Ishiyama, M., Kunishi, M., Uehara, K., Esaki, H. and Teraoka, F.: LINA: A New Approach to Mobility Support in Wide Area Networks, *IE-ICE Trans. Comm.*, Vol.E84-B, No.8, pp.2076–2086 (2001).
- 2) Perkins, C. and David, J.: Mobility Support in IPv6, *Proc. MobiCom'96*, ACM (1996).
- 3) Mathis, M., Floyd, S. and Romanow, A.: TCP Selective Acknowledgment Options, *IETF*, RFC2018 (1996).
- 4) 西田佳史: 無線ネットワークにおける TCP の改善に関する考察, 情報処理学会研究報告 (2000).
- 5) Barkre, A. and Badrinath, B.R.: I-TCP: Indirect TCP for Mobile Hosts, *Proc. 15th ICDCS* (1995).
- 6) Inamura, H. and Ishikawa, T.: A TCP profile for W-CDMA: 3G wireless packet service, *IETF, draft* (2000).
- 7) Inamura, H.: TCP over 2.5G and 3G Wireless Networks, *IETF, draft* (2001).
- 8) Balakrishnan, H., Seshan, S., Amir, E. and Katz, R.: Improving TCP/IP Performance over

Wireless Networks, *Proc. 1st ACM Conf. on Mobile Computing and Networking* (1995).

- 9) Balakrishnan, H., Seshan, S., Amir, E. and Katz, R.: Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks, *Proc. ACM Conf. on Mobile Computing and Networking* (1995).
- 10) Ratnam, K. and Matta, I.: WTCP: An Efficient Mechanism for Improving TCP Performance over Wireless Links, *Proc. 3rd IEEE Symposium on Computer and Communications (ISCC)* (1998).
- 11) Balakrishnan, H. and Katz, R.H.: Explicit loss notification and wireless web performance, *IEEE Globecom Internet Mini-Conference, Sydney* (1998).
- 12) Tsaoussidis, V. and Badr, H.: TCP-Probing: Towards an Error Control Schema with Energy and Throughput Performance Gains, *Proc. 8th IEEE Conference on Network Protocols, Japan* (2000).
- 13) Mascolo, S., Casetti, C., Gerla, M., Lee, S.S. and Sanadini, M.: TCP Westwood: Bandwidth Estimation for Enhanced Transport over Wireless Links, *7th annual international conference on Mobile computing and networking* (2001).
- 14) Saltzer, J.H., Reed, D.P. and Clark, D.D.: End-To-End Arguments In System Design, *ACM Trans. Comput. Syst.*, Vol.214, pp.277-288 (1984).
- 15) Kent, S. and Atkinson, R.: Security Architecture for the Internet Protocol, *IETF, RFC2401* (1998).
- 16) Stevens, R.W.: TCP/IP Illustrated Volume 1 (1994).
- 17) Soliman, H., Castelluccia, C., Malki, K.E. and Bellier, L.: Hierarchical MIPv6 mobility management, Internet draft, *IETF* (2001). draft-ietf-mobileip-hmipv6-04.txt
- 18) NS2. <http://www.isi.edu/nsnam/ns/>
- 19) 802.11b. <http://grouper.ieee.org/groups/802/11/>

(平成 14 年 3 月 25 日受付)

(平成 14 年 10 月 7 日採録)



佐藤 徳彦 (学生会員)

1977 年生。2002 年慶應義塾大学大学院工学研究科開放環境科学専攻修士課程修了。主にトランスポート層のプロトコルに興味を持ち、TCP の改良に取り組む。現在、ソニー

(株)勤務。



國司 光宣

1976 年生。2001 年慶應義塾大学大学院理工学研究科前期博士課程修了。現在、同大学院同研究科後期博士課程に在学中。ネットワーク層でのモビリティサポートの研究、IPv6 の改良に興味を持ち、IPv6 のプロトコル設計、実装にたずさわる。



寺岡 文男 (正会員)

慶應義塾大学理工学部情報工学科教授。1959 年生。1984 年慶應義塾大学大学院工学研究科電気工学専攻修士課程修了。同年キャノン株式会社入社。1988 年株式会社ソニーコンピュータサイエンス研究所入社。2001 年 4 月から現職。博士 (工学)。1991 年日本ソフトウェア科学会高橋奨励賞受賞。1993 年元岡記念賞受賞。2001 年情報処理学会平成 12 年度論文賞受賞。コンピュータネットワーク、オペレーティングシステム、分散システム等の研究に従事。特に移動透過性を提供するプロトコル VIP (Virtual IP) の開発を通して IETF の Mobile IP 分科会の活動に貢献。2000 年 5 月から 2002 年 5 月まで情報処理学会理事。著書に「ワイヤレス LAN アーキテクチャ」(共著, 共立出版)。監訳に「詳解 Mobile IP」(共監訳, プレンティスホール出版)。ACM, IEEE, 日本ソフトウェア科学会, 電子情報通信学会各会員。