

SURE: Shizuoka University REpository

<http://ir.lib.shizuoka.ac.jp/>

Title	利用経路を動的に制御する複数経路集約通信方式の評価
Author(s)	川島, 佑毅; 峰野, 博史; 石原, 進; 水野, 忠則
Citation	情報処理学会論文誌. 48(2), p. 880-891
Issue Date	2007-02-15
URL	http://hdl.handle.net/10297/1918
Rights	(c)社団法人情報処理学会

This document is downloaded at: 2009-11-23T03:16:42Z

推薦論文

利用経路を動的に制御する複数経路集約通信方式の評価

川島 佑 毅[†], 峰野 博 史^{††}
石原 進^{†††} 水野 忠 則^{††}

我々は複数ネットワークインタフェースを持つ端末を想定し、モバイル環境で複数経路を同時に利用し高速高信頼な通信を実現する Mobile IP SHAKE を開発・評価してきた。その結果、トラフィックをパケットごとに各経路へ振り分けた場合、各経路の可用帯域や遅延、遅延揺らぎの違いによって、通信相手にパケットの到着順序の入れ替わりが頻発し、経路集約型通信の効果が得られない状況が発生することが分かった。本稿では、パケットの到着順序入れ替わりの許容範囲について検討し、各経路の遅延と遅延揺らぎを測定することで、到着順序の入れ替わりが許容範囲を超える可能性のある経路へパケットを振り分けない制御方式を検討した。検討方式の実装評価を行い、従来方式よりもパケット到着順序逆転を防ぐことができ、複数経路間でジッタや通信速度に差がある環境でも、安定したスループットが得られることを確認した。

Evaluation of Multiple Paths Distribution Method Based on Delay-jitter

YUKI KAWASHIMA,[†] HIROSHI MINENO,^{††} SUSUMU ISHIHARA^{†††}
and TADANORI MIZUNO^{††}

At present, mobile nodes have multiple network interfaces. In response to evolving system requirements, we have developed and evaluated the Mobile IP SHAKE, designed to use multiple Internet-linked mobile nodes simultaneously in a mobile ad-hoc network. It disperses traffic to multiple paths between mobile nodes and a correspondent node in the Internet. An earlier evaluation of Mobile IP SHAKE revealed that the TCP throughput decreases due to the out-of-order packets on the receiver when the delay-jitters of multiple paths are extremely different. This paper proposes a new traffic distribution method that avoids the out-of-order packets caused by the delay-jitter by not using paths with large delay-jitter. Results of experiments show that the proposed method avoids the throughput degradation when the delay-jitters of multiple paths are extremely different.

1. はじめに

多種多様な無線通信インフラの普及と、低コストでの常時無線接続環境の整備、MEMS (Micro Electro Mechanical System) 技術の発展により、異種混在のヘテロジニアスなネットワーク環境を意識しないで利用可能なモバイルコミュニケーション環境の到来が予測される。昨今では、有線 LAN、無線 LAN、モバイ

ルデータ通信用 PC カードといった複数のネットワークインタフェース (NetIF) を利用する移動端末が一般的となり、これら複数 NetIF を活用し複数の経路を同時利用した通信方式 (以降、経路集約通信) が注目を集めている。様々なプロトコル階層での経路集約型通信の実現方法が研究されているが、以下に実現レイヤによって異なる利点欠点についてまとめる。

データリンク層アプローチは、上位層プロトコルへいっさいの影響を与えない透過性と高効率な経路集約を実現することができる^{1)~5)}。ただし、専用のハードウェア機器間でしか経路集約型通信の恩恵を受けられないといった特徴があり、ルータ間で同一メディアを用いた WAN 回線を集約するのに使用されるのが

[†] 静岡大学大学院情報学研究科

Graduate School of Informatics, Shizuoka University

^{††} 静岡大学情報学部

Faculty of Informatics, Shizuoka University

^{†††} 静岡大学工学部

Faculty of Engineering, Shizuoka University

現在、三菱電機株式会社

Presently with Information Technology R&D Center,
Mitsubishi Electric Corporation

本稿の内容は 2005 年 3 月のモバイルコンピューティングとユビキタス通信研究会にて報告され、MBL 研究会主催により情報処理学会論文誌への掲載が推薦された論文である。

一般的である。ネットワーク層アプローチは、Mobile IPv4 や Mobile IPv6 と IP-in-IP カプセル化を用い、単一フローを形成する IP パケットをホームエージェント (Home Agent: HA) で複数の経路へソースルーティングさせることで経路集約型通信を実現する^{6),7)}。これらの Mobile IP ベースの方式は、通信相手 (Correspondent Node: CN) が経路集約型通信機能をサポートしている必要がなく、上位層プロトコルへも透過性を提供することができる。ただし、HA でのトラフィック分配機能の性能が End-to-End のスループットへ大きな影響を与え、複数経路間の遅延差や帯域差、変動の大きい場合に帯域集約によって高スループットを得るための効率良いトラフィック分配方法が重要となる。トランスポート層アプローチでは、TCP の輻輳制御や再送制御機能を拡張することで、各経路の状態に応じた効率的なトラフィック分配や再送を行うことができる⁸⁾⁻¹⁴⁾。他レイヤでの経路集約型通信で利用されているような、単純な測定ベースのトラフィック分配方式では、遅延や帯域変動の激しい場合や経路間の遅延差や帯域差の変動が大きい場合に適切なトラフィック分配を行うことは難しい。そのため、容易な作業ではないが、TCP の輻輳制御や再送制御機能を経路集約通信に対応するように拡張することで、高速高信頼な経路集約通信を実現することができる。ただし、より複雑な制御、End-to-End でのサポートが必要となり、一般に広く普及するのは困難である。一方、アプリケーション層アプローチは実装が容易であり、複数の TCP コネクションを同時に利用することでスループットと信頼性向上に対する効果が示されている¹⁷⁾⁻¹⁹⁾。ただし、特定のアプリケーションやプロトコルを想定した経路集約型通信しか実現できない、End-to-End のスループットがトラフィック分配機能の性能に大きく依存するといった特徴を持つ。

以上の分析結果から、筆者らは経路集約型通信の普及が現実的であるネットワーク層アプローチの Mobile IP SHAKE を提案し評価を行ってきた⁷⁾。Mobile IP SHAKE では、近隣の移動端末と無線 LAN や Bluetooth 等の比較的高速な短距離無線通信技術を利用して一時的な協力関係 (アライアンス) を構築し、アライアンス内の移動端末を HA へ登録する。HA で複数の経路へソースルーティングさせることで経路集約型通信をネットワーク層で実現する。これまでに、HA でのトラフィック分配機能を改良し、パケットごとの分配とフローごとの分配の 2 種類の分配方式を選択可能とする制御機構を開発し評価してきた²¹⁾。その結果、パケットごとの分配で、より高く安定したスループット

を得るためには、利用可能なすべての経路を使用するのではなく、品質の悪い経路は使用を停止した方がよいことが分かった。本稿では、Mobile IP SHAKE において、遅延揺らぎ (ジッタ) を考慮したトラフィック分配を行うことで、品質の悪い経路の含まれる経路集約通信環境でも高いスループットを実現する仕組みについて検討する。

以下、本稿の構成を記す。2 章では、Mobile IP SHAKE の動作、パケット分配方法、従来の問題点について説明する。3 章では、パケットの順序逆転の許容範囲、ジッタを考慮した分配方式を提案し、経路品質を決定するパラメータと測定法について検討する。4 章では、エミュレーション環境において、従来方式と本稿で提案するジッタを考慮した分配方式を比較し評価結果を示す。5 章で本稿を総括し、今後の課題を示す。

2. Mobile IP SHAKE

2.1 Mobile IP SHAKE 概要

以下に Mobile IP SHAKE の概要を示す。Mobile IP SHAKE では、移動端末 (Mobile Node: MN) がアライアンス外部にいる任意の CN と通信を実現するために、Mobile IPv4 で経路最適化を行わない場合に CN と MN 間の通信は必ず HA を中継するという特性を利用している。つまり、HA にトラフィック分配機構を持たせることで、経路集約型通信をネットワーク層で実現している。HA でトラフィックを分配することで、CN には分配のための特別な機構を持たせる必要がなく、MN は複数経路を利用した通信が可能となる。

一方、当研究グループでは、Mobile IPv6 による経路最適化を行った場合の SHAKE についても提案・実装を行っている²⁰⁾。経路最適化を行う場合は、CN が SHAKE に対応したトラフィック分配機構を持つ必要がある。以降本稿では Mobile IPv4 で経路最適化を行わない SHAKE について述べる。

Mobile IP SHAKE の動作概要を図 1 に示す。移動端末 (MN1) は Mobile IP と同様に自身の気付けアドレス (MN1 CoA) を HA に登録する。もし、近隣の移動端末 (MN2) が MN1 のアライアンスに参加しているなら、MN1 は MN2 の気付けアドレス (MN2 CoA) も HA に登録する。これにより、MN1 の HA は、MN1 宛のトラフィックを MN1 CoA と MN2 CoA へ送信できるようになる。ここで、SHAKE を利用して通信を行う端末 MN1 をアライアンスリーダー (Alliance Leader: AL)、トラフィックを中継する端末 MN2 を

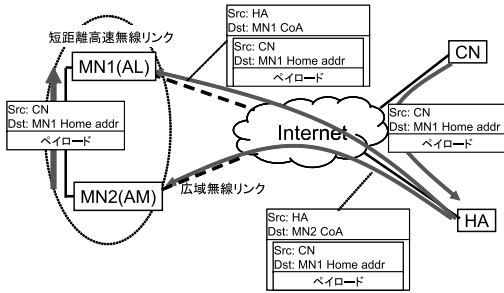


図 1 Mobile IP SHAKE 動作概要
Fig.1 Operation image of Mobile IPSHAKE.

アライアンスメンバ (Alliance Member: AM) と呼ぶ。また、HA と AL はユーザの設定したポリシーに従い、通信速度幅や遅延を考慮してトラフィックをパケットごともしくはフローごとに分配するトラフィック分配機構を持つ。そのため、AL と AM の各端末は理論通信速度、パケットコスト等の端末固定の情報 (Terminal Profile) をトラフィック分配機構に登録することができる。

CN から AL への下りトラフィックを例に、トラフィック分配機構の動作を説明する。HA は、CN から AL 宛の IP パケットを受信するため、AL 宛の IP パケットを AL だけでなく、AM 宛にもカプセル化して転送する。AM はトラフィックを受信するとカプセル化を解除して AL へ転送する。このとき、分配手法がパケットごとの場合は、Terminal Profile に基づき各経路の通信速度と遅延からパケット到着時間を予想し、パケットが宛先へ順序どおりに到着するように送信する DQ (Delay Queuing) 方式を採用している。フローごとの分配の場合は、あらかじめフローとその扱いをポリシーとして対応付けておくことで、通信速度重視やパケットコスト重視の経路を選択する。

2.2 DQ (Delay Queueing) 方式

経路集約通信で TCP トラフィックをパケットごとに分配する場合、各経路の通信速度や遅延が異なるため、ラウンドロビンのような単純な分配方法ではパケット到着順の逆転が頻発し、TCP の送信制御が働いてスループットが低下してしまう。

経路集約通信のパケットごと分配では、各経路の通信速度差と遅延時間差を考慮しパケットが順序どおりに到着するように分配する手法を用いている^{7),14)~16)}。Mobile IP SHAKE では、各経路のパケット転送処理を仮想的な送信キュー (仮想送信キュー) (図 2) と見なす DQ 方式を採用している。DQ 方式では、キュー送信待ち時間 (キューサイズ/通信速度) と伝送遅延時間の合計時間が最短になる経路へ送信パケットを振

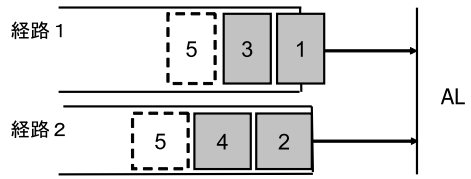


図 2 仮想送信キュー
Fig. 2 Example of virtual queue.

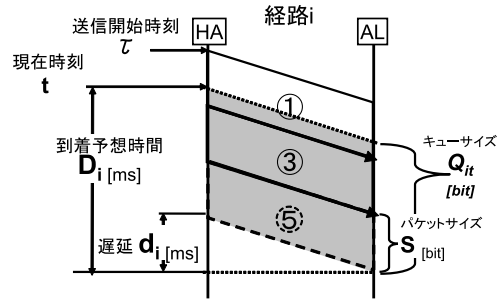


図 3 パケット到着予想時間
Fig.3 Forecast of packet arrival time.

り分ける。

DQ 方式におけるパケットの到着時間は、現在の仮想送信キューにストアされているデータを送出し終わるまでに要する時間とパケットの伝送遅延時間の合計で求められる。図 3 に仮想送信キューのシーケンスを示す。図 3 では、パケット 1 と 3 がストアされており、新たにパケット 5 がストアされた状態である。経路 i でパケット 1 が送出開始された時刻を τ 、パケット 5 がキューにストアされた現在時刻を t とすると、現在のキューサイズ Q_{it} [bit] は、 τ 時点でのキューサイズ $Q_{i\tau}$ [bit] と経路 i の通信速度 B_i [kbps] で時刻 t までに送出したデータの差にパケット 5 のサイズ S [bit] を加えることで求められる (式 (1))。

$$Q_{it} = Q_{i\tau} - B_i(t - \tau) + S \tag{1}$$

そして、この Q_{it} [bit] を B_i [kbps] で送出し終わるまでに要する時間に伝送遅延時間 d_i [ms] を加えることで、パケット 5 の到着予想時間 D_i [ms] を算出する (式 (2))。

$$D_i = \frac{Q_{it}}{B_i} \cdot 1000 + d_i \tag{2}$$

ここで、伝送遅延 d_i の測定方法については、3.4 節で説明する。

2.3 経路状況によるスループット低下

SHAKE のような経路集約通信では、各経路の品質や状態が変化するため、前節の到着順序予測をしてもパケット到着順序逆転が発生してしまう場合がある²¹⁾。経路品質や状態の変化への対応として、文献 16) で

は、経路の片道遅延時間と可用転送速度を測定することで実際の通信速度に応じた分配を行っている。しかし、この方式では遅延揺らぎは考慮されていない。文献 15) では、パケット分配のためのプロキシが ack をキャプチャすることで、重複 ack の原因が順序逆転なのかパケットロスなのか判断し ack 返送を行っている。しかしながら、これらの方式では、いかなる場合もパケットを複数経路へ送信してしまうため、品質の悪い経路を使用し続けてしまう。

SHAKE では、複数の移動端末を使用することを想定している。そのため、ある端末の経路だけ品質が低下する可能性がある。品質の低下した経路を使用し続けるよりも、品質の良い経路だけに分配した方がパケット到着順序逆転の可能性は低くなる。つまり、利用可能な複数経路の状況に応じて、利用する経路を切り替える仕組みがあれば、複数経路をより効果的に活用した経路集約通信が実現できると考える。

以降では、まず、経路集約型通信が影響を受ける経路の品質を表すパラメータとその測定方法について検討する。その後、測定したパラメータから、高いスループットを得られる経路を動的に選択する仕組みについて検討する。

3. ジッタを考慮した分配

3.1 経路品質を表すパラメータ

HA 上のトラフィック分配機構で、経路の状況に応じて使用経路を切り替える機能を実現するために、経路の品質を表すパラメータについて検討する。

経路の品質を決定するパラメータとして、可用通信速度、伝送遅延、ジッタ、パケットロス率の 4 つがあげられる。実際の通信では、これらのパラメータがそれぞれ変動し経路品質に影響を与えている。しかし、本稿では問題を単純化するために、通信速度は Terminal Profile の交換により既知であり、通信中に変化することはないとする。また、パケットロス率も、長距離無線通信ではデータリンク層で再送を行うため、非移動時のパケットロス率は十分小さいとして、パケットロスがない状態を想定する。このような想定環境のもと、Mobile IP SHAKE を用いて TCP 通信を使用した際に、伝送遅延やジッタの変化が End-to-End のスループットに与える影響を検討する。

まず遅延について検討する。経路の遅延が増大すると、TCP の基本性能である遅延帯域積により、スループットが低下してしまう。そこで、あらかじめ TCP の受信ウィンドウサイズを十分確保しておくことで対応する。

次にジッタについて考える。ジッタとは伝送遅延のバラツキ具合を表すものである。ジッタが大きくなると、DQ 方式で予想したパケット到着時間と実際の到着時間に差が生じ、パケット到着順序の逆転が生じてしまう。加えて、使用可能な経路数が多くなるほど、各経路のジッタの違いによってパケット到着順序の逆転が発生しやすくなる。パケット到着順序の逆転が頻発すると、重複 ack の影響で TCP は送信量の制限を行うため End-to-End のスループットは抑制されてしまう。

DQ 方式では各経路の遅延を測定しているため、測定した遅延から経路のジッタを算出することが可能である。そのため、得られたジッタを用いて、パケットの振分けに使用する経路と使用しない経路を選択することでパケット到着順序逆転の発生を抑えられる方式を検討する。

3.2 パケット到着順序逆転の許容範囲

パケットの到着順序逆転によるスループット低下が発生する条件について検討を行う。TCP 通信においてパケット到着順序の逆転が生じると受信ホストは重複 ack を送信する。一般的な TCP (TCP Tahoe, TCP Reno, TCP newReno) は高速再送アルゴリズムにより、送信ホストが重複 ack を 3 回連続で受信するとパケット再送を行い、送信ウィンドウを減少させてしまう。パケットの到着順序逆転が生じた場合でも、重複 ack が 2 つまでなら再送は発生しない。つまり、パケットの到着順序逆転の許容範囲は 3 パケット未満ということになる。よって、各経路の仮想送信キューに溜まっているトラフィック量とジッタの大きさから、パケットの到着順序逆転が 3 つ未満に収まるような経路利用判定を行う方法が考えられる。

ただし、TCP Vegas は 3 回の重複 ack を待たずに、高速再送アルゴリズムによるパケット再送を行うため、本提案方法は適用不可である。しかし、TCP Vegas は他の TCP との親和性が悪いことから広く普及しているプロトコルではないため、本稿ではパケット到着順序逆転の許容範囲を 3 とする。

3.3 経路利用判定 (DQJ) 方式

DQ 方式で用いた仮想送信キューヘジッタ (Delay-Jitter) を考慮することで経路利用判定を行う DQJ (DQ-Jitter) 方式を提案する。

DQJ 方式では、仮想送信キュー内のパケットがジッタによって入れ替わる数を経路ごとにカウントする。その結果、全経路での入れ替わり数が 3 未満になる経路の組合せのみへトラフィックを振り分けることができる。たとえば、経路 i の通信速度を B_i [kbps]、送

信パケットサイズを S [byte] とすると, t [ms] 間に送出するパケットの数は, 式 (3) で表すことができる.

$$P_i(t) = \left\lfloor \frac{B_i \cdot t}{8 \cdot S} \right\rfloor \quad (3)$$

経路 i が経路 j のジッタ J_j [ms] の間に送信するパケット数は $P_i(J_j)$ となる. そのため, 使用可能な経路が n 本で, 経路 j のジッタ J_j [ms] 間に, すべての経路から送出されたパケットが 3 パケット未満になる経路利用のジッタの閾値は, 式 (4) で表すことができる.

$$\sum_{i=1}^n P_i(J_j) < 3 \quad (4)$$

DQJ 方式では, トラフィック分配機構がパケット分配を行うたびに, この条件判定を行い, この条件を満たす経路へ DQ 方式を用いてパケットを振り分ける.

式 (3) を用いて, 経路 j のジッタ J_j が閾値を超えている間, HA は経路 j へのパケット振分けを停止する. しかし, ジッタが大きな経路でも単独での使用はスループットへの影響が少ないことが分かっている²²⁾. そのため, 使用可能な経路の合計通信速度とジッタの大きな経路の通信速度を比較し, より大きな通信速度を得られる経路を選択することとする.

図 4 に DQJ 方式を使用した場合の各経路の仮想送信キューとジッタの状態を示す. 図 4 において, 経路 1 で送信中のパケットがジッタ J_1 によって, 到着順序逆転を引き起こす可能性のあるパケットは, 番号 1, 2, 3, 4 の 4 パケットとなる. パケット到着順序逆転は 3 パケットまでが望ましいため, 経路 1 はジッタが大きくスループット低下を招く恐れがあると判断できる. 一方, 経路 2 のジッタ J_2 は, パケット送出時間に対して十分小さいため, スループット低下を引き起こす原因にはならないと判断できる.

3.4 経路品質測定

3.3 節で述べた経路品質を用いて利用経路を判定す

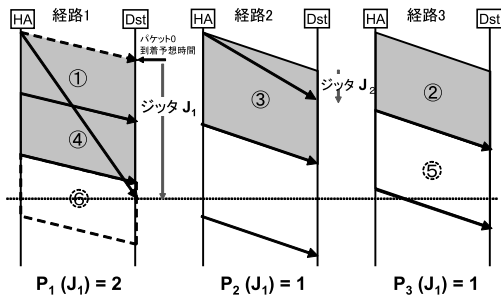


図 4 各経路の仮想送信キューとジッタ
Fig. 4 State of each virtual queue.

るために, HA-AL 間, HA-AM 間の遅延, ジッタを測定する方法を検討する. これらの測定手法には, 送信するトラフィックから求めるパッシブ測定と測定用のパケットを使用するアクティブ測定がある.

パッシブ測定は, 余計なトラフィックを増やさず測定が可能であるが, トラフィックのない経路での測定はできない. 一方, アクティブ測定では, 利用中の送信経路以外の経路についても測定が可能である. しかし, 新たにパケットを送信するため, 測定用パケット自体が経路に負荷をかけてしまわないよう注意する必要がある. 以上のように, 利用中の経路しか測定できないパッシブ測定より, すべての利用可能な経路の状態を把握できる測定用パケットによるアクティブ測定が適すると考える.

図 5 にアクティブ測定に用いるパケットフォーマット, 図 6 に遅延とジッタの測定方法を示す. HA-AL 間, HA-AM 間の各経路で, n 番目の測定用発信パケットは, 経路識別子, シーケンスナンバー n , HA 送信タイムスタンプ S_n , HA のアドレス, AM アドレスを含む 60 byte の UDP パケットであり, 応答パケットは, 経路識別子, シーケンスナンバー n , HA 送信タイムスタンプ S_n , AM アドレス, AM での処理時間 P_n , ジッタの測定結果 J_n を含む 64 byte の UDP パケットである.

AL と AM は測定用パケットを受信し, ジッタを算

0	16	31
経路識別子		シーケンスナンバー n
HA送信タイムスタンプ S_n		
HAアドレス		
AMアドレス		

(a) 送信フォーマット

0	16	31
経路識別子		シーケンスナンバー n
HA送信タイムスタンプ S_n		
AMアドレス		
AM処理時間 P_n		
ジッタ測定結果 J_n		

(b) 応答フォーマット

図 5 測定用パケットフォーマット
Fig. 5 Format of probe packet.

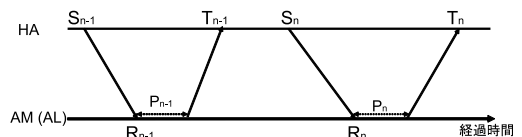


図 6 遅延, ジッタ測定手法

Fig. 6 Measurement method of delay and delay-jitter.

出する。ジッタ算出には、RTP (RFC1889) の手法を用いた。RTP はネットワークの時間の情報を掌握し、データ送信量調節するリアルタイムデータ転送プロトコルである。

まず、測定用パケットから得られる送信タイムスタンプ S と受信タイムスタンプ R から連続するパケット $n-1, n$ の伝送遅延時間の差 (片道遅延の差) G を下式で算出する (式 (5))。

$$G_{n-1,n} = (R_{n-1} - S_{n-1}) - (R_n - S_n) \quad (5)$$

そして、ジッタ J は、パケット n と1つ前のパケット $n-1$ の伝送遅延の時間差 G と前回計測のジッタ J を用い、下式のように算出できる (式 (6))。

$$J_n = J_{n-1} + \frac{(|G_{n-1,n}| - J_{n-1})}{16} \quad (J_0 = 0) \quad (6)$$

RTP では、突発的なジッタの影響を少なくするために、平滑化のパラメータに 16 を用いている。

AL と AM は式 (6) の結果と端末での処理時間 p_n をパケットに記録し HA へ返信する。HA は AL, AM からのパケットを受信すると、受信時のタイムスタンプ T_n から片路伝送遅延 d_n を計測する (式 (7))。

$$d_n = \frac{(T_n - S_n) - p_n}{2} \quad (7)$$

また、突発的な遅延増大の影響を小さくするため、これまでに測定した遅延 d_{old} を用いて測定結果の平滑化を行う (式 (8))。

$$d = \alpha \cdot d_{old} + (1 - \alpha) \cdot d_n \quad (0 < \alpha < 1) \quad (8)$$

なお、本稿における α の値については 4.1 節で後述する。

式 (6) と式 (8) を用いて各経路において求めたジッタ J 、遅延 d を式 (2) と式 (3) で利用する。

4. 性能評価

4.1 評価環境

ヘルシンキ工科大学で実装された Linux 版 Mobile IP の実装である Dynamics²⁴⁾ をベースに開発された Mobile IP SHAKE を拡張し、経路利用判定の機能を追加した。本実装を利用し、図 7 に示すように CN と HA を設置し、3 台の移動端末でアライアンスを構築した。ここで、本評価は DQJ 方式の基本特性の評

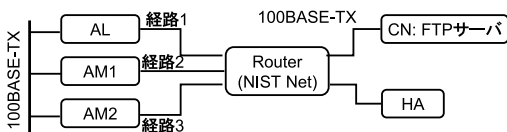


図 7 評価環境
Fig. 7 Experimental environment.

価が目的であるため、無線通信の干渉の影響を考慮する必要がないよう、すべての端末を有線で接続した。Router 上でネットワークエミュレータ NIST Net を動作させることで各経路の基本品質を第 3 世代携帯電話を模擬し、下り 384 kbps, 上り 64 kbps, 片道遅延 200 ms, ジッタ ± 50 ms に設定した。ジッタの分布は、インターネットのジッタ分布をモデル化するのが困難なため NIST Net の標準である一様分布に従う。

現在の実装では、ack 返送路は AL の持つ外部リンクを使用する。また、測定用パケットの送信間隔を 1 秒とした。測定用パケットのサイズが 60 byte と 64 byte と小さいため、1 秒間隔程度ならば他のトラフィックへの影響は小さいと思われる。さらに、遅延の平滑化係数 α は TCP の RTT 平滑化係数を参考に 0.8 とした。

なお以降の実験では、アライアンスが構築され、遅延・ジッタ測定を行っている状態での提案方式の評価を行う。

4.2 実験 1: 静的ジッタに対する評価

評価環境 (図 7) において、AL が FTP サーバ (CN) から約 2 Mbytes のデータをダウンロードする際のスループットを 5 回計測し、その平均値を分析した。

経路 3 のみジッタを $\pm 0 \sim 200$ ms で変化させることで、ジッタの変動が DQ 方式のスループットに与える影響と、DQJ 方式がジッタの大きい経路を判定し、利用を停止することで高いスループットを保つことができるかを検証する。また、単一経路利用時の平均スループットも DQ, DQJ 方式との比較のために示す。

4.3 実験 1 の評価結果

図 8 に実験結果を示す。経路 3 本使用時の平均スループットに関して、DQ 方式、DQJ 方式ともジッタのない ± 0 ms という理想的な環境では、単一経路のみを利用した場合に比べ、およそ 3 倍弱のスループットが得られた。ジッタが増加すると、スループットは

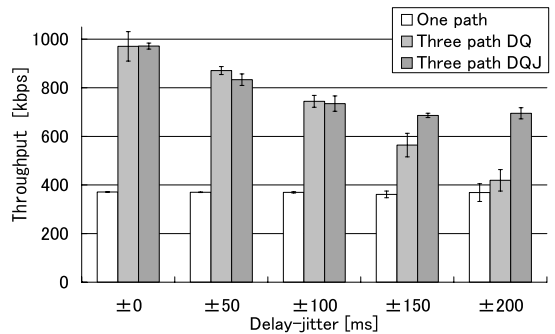


図 8 ジッタによるスループットの変化 (経路 3 本)
Fig. 8 Throughput against delay-jitter.

低下していくが、±50 ms, ±100 ms までは DQ 方式, DQJ 方式ともに大きな差は見られない。しかし、ジッタが ±150 ms を超えると、DQ 方式のスループットが低下したのに対して、DQJ 方式のスループットはおよそ 700 kbps で安定しており、経路 2 本使用時のスループットと同等である。これは DQJ 方式の経路利用判定によってジッタの大きな経路 3 への送信が停止し、ジッタの小さな経路 1, 2 の 2 本で通信を行ったためである。そのため、ジッタの大きな経路 3 を利用することで発生する重複 ack の頻発によるスループットの低下が抑えられたことが分かる。

ここで、経路 3 本でジッタ ±200 ms のときの DQ 方式, DQJ 方式における各経路へのトラフィック送信量 (rate of traffic) と AL の受信ウィンドウサイズ (Rwin), ack ナンバ (#ack) について詳細に分析する。DQ 方式の結果を図 9 に、DQJ 方式の結果を図 10 に示す。

図 9 より、DQ 方式ではジッタの大きな経路 3 へもトラフィックを送信することが分かる。その結果、パケット到着順序の逆転が頻発し、重複 ack が頻繁に送信される。また、TCP はすでに到着しているパケットを保持しておかなくてはならず、受信ウィンドウサイズが圧迫されてしまう。その結果、AL は広告ウィンドウサイズを頻繁に下げため、全体の送信量 (total) が低下した。

図 10 より、DQJ 方式では、ジッタ測定値と各仮想

送信キューの大きさによっては、経路 3 へトラフィックを送信してしまい重複 ack が発生し、一時的にスループットが低下するときがある。しかし、経路利用判定によって経路 3 への送信が抑制された後は、ジッタの小さな経路 1, 2 のみの利用に切り替わり、安定した通信を行うことが確認できる。

ここで、各方式の 3 パケット以上の重複 ack 発生率は、DQ 方式が 10.71%であったのに対して、DQJ 方式は 0.02%であった。つまり、DQJ 方式の経路利用判定が有効に動作し、パケット到着順序の逆転による重複 ack の発生を抑えられることが確認できた。

4.4 実験 2: 動的ジッタに対する評価

次に、移動環境を想定し、ジッタが動的に変化する設定で DQJ 方式の有効性を評価した。評価環境 (図 7) で、経路 1, 2 (path1, path2) のジッタを ±50 ms とし、経路 3 のジッタのみを表 1 のように変化させた。AL が FTP サーバ (CN) から約 5 Mbytes のデータをダウンロードする際の DQ 方式と DQJ 方式の各経路へのトラフィック送信量と AL の受信ウィンドウサイズ, ack ナンバの推移を分析する。

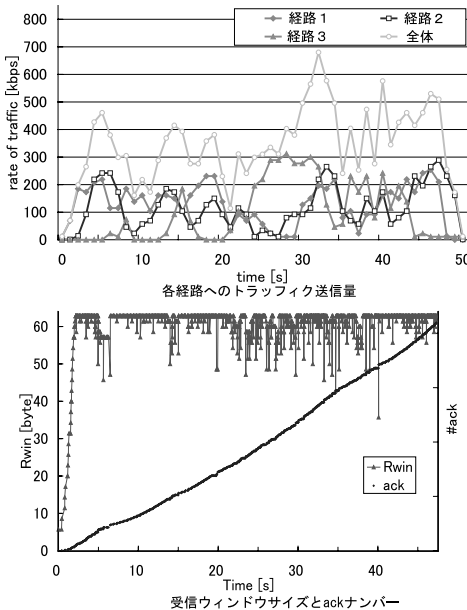


図 9 DQ 方式 : 384 kbps × 3 本, ジッタ ±200 [ms]
Fig. 9 DQ in case of 384 kbps × 3 and delay-jitter 200 ms.

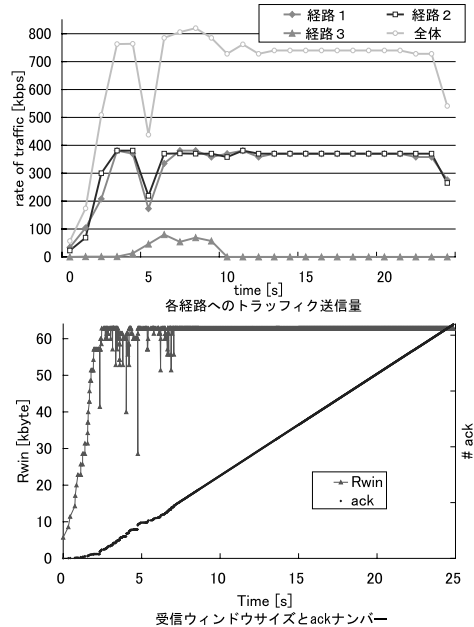


図 10 DQJ 方式 : 384 kbps × 3 本, ジッタ ±200 [ms]
Fig. 10 DQJ in case of 384 kbps × 3 and delay-jitter 200 ms.

表 1 経路 3 のジッタ
Table 1 Status of delay-jitter in path 3.

経過時間 [s]	0 ~	10 ~	20 ~	30 ~	40 ~
ジッタ [ms]	±50	±100	±150	±200	±50

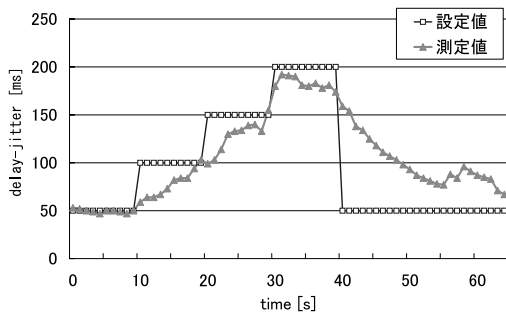


図 11 ジッタの測定結果

Fig. 11 Measurement result of delay-jitter.

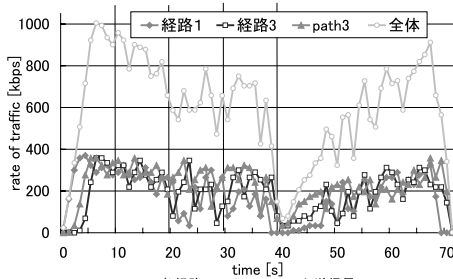


図 12 動的ジッタに対する DQ 方式の評価

Fig. 12 DQ in case of dynamic delay-jitter.

4.5 実験 2 の評価結果

図 11 にジッタの設定値と測定結果のズレを示し、図 12 に DQ 方式の結果を、図 13 に DQJ 方式の結果を示す。

ジッタ測定の結果（図 11）において、通信開始 10 秒までは、通信開始前と同様にジッタ ± 50 ms であるため、設定値と測定値がほぼ一致する。しかし、20 秒以降、設定値を ± 100 ms, 150 ms, 200 ms, 50 ms と変動させると、設定値と測定値にズレが生じ、設定値が測定値に反映されるまでに 10 秒程度かかる。これは、ジッタの算出（式 (6)）において突発的な変動の影響を少なくするために前回の測定結果を重み付けし平滑化を行っているためである。

DQ 方式の結果（図 12）において、通信開始後 20 秒までは、経路 3 (path3) のジッタが ± 50 ms, ± 100 ms

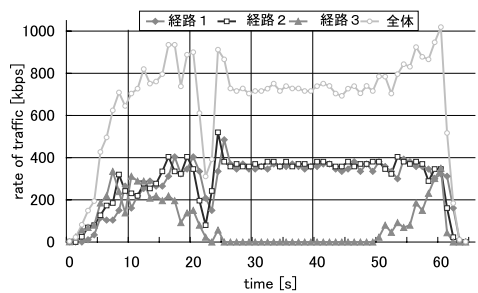


図 13 動的ジッタに対する DQJ 方式の評価

Fig. 13 DQJ in case of dynamic delay-jitter.

と小さいため各経路へ安定してパケットを分配しており、全体のトラフィック送信量は $800 \sim 1,000$ kbps と多い。しかし 20 秒経過後、経路 3 のジッタが ± 150 ms になると、重複 ack の数が増加し、受信ウィンドウサイズは小さくなり、全体のトラフィック送信量は低下する。さらに、ジッタが ± 200 ms になってしばらくすると、急激に全体のトラフィック送信量が低下し、一時的にほとんどトラフィックが送信されない状態になる。これは、パケット到着順序の逆転が頻発したことによって、CN の送信ウィンドウが大きく減少したためと考えられる。40 秒経過後にジッタが ± 50 ms に戻ると、徐々に全体のトラフィック量が増加する。これは、ジッタが ± 200 ms の設定時に、下がりきってしまった送信ウィンドウサイズがスロースタートによって徐々に大きくなるためだと考えられる。つまり、ジッタの大きな環境では DQ 方式のスループットは大きく低下してしまう。

次に、DQJ 方式の結果（図 11, 図 13）について分析する。通信開始後は、ジッタが小さいため時間の経過とともに全体への送信量が着々と増加する。

10 秒経過後、ジッタの設定値が ± 100 ms になっても、しばらくは全経路にトラフィックを送信する。しかし、ジッタの測定値が設定値の 100 ms に近づくと経路 3 への送信量は低下しする。20 秒経過後、ジッタの設定値が ± 150 ms に増加すると、経路 1, 経路 2 への送信量が一時的に低下する。これは経路 3 へ送

信されたパケットの到着順序の逆転によって送信量の抑制が発生し、全体のトラフィック送信量が低下すると考えられる。ジッタの測定値が 150 ms に近づくと、経路 3 へのトラフィック送信は完全に停止され、ジッタの小さな経路 1, 2 のみを使用して安定した通信をする。

その後、40 秒経過後にジッタの設定値が ± 50 ms に戻っても、ジッタの測定値が追従できるまでは経路 3 への送信を再開していない。50 秒経過後の時点で、ジッタの測定値が利用判定を満たし、経路 3 へのトラフィック送信が再開され全体の送信量が増加する。

以上より、DQ 方式はジッタの影響でスループットが大きく低下してしまうのに対し、DQJ 方式はジッタの大きな経路への送信を停止することで、大幅なスループット低下を防ぐことができる。ただし、今回のジッタ測定は測定パケットの送信間隔を 1 秒とし、前回結果を積み付けし測定結果に反映する。そのため、ジッタの測定値が設定値近づき経路利用判定に反映されるまで 10 秒程度要し、その間は適切な判定が行えていない。提案方式は評価実験のような一定以上継続して通信がなされ、ジッタの変動が頻繁でない場合に有効であるといえる。

ジッタが頻繁に変化する状況に対応するためには、測定間隔を短くし、前回の測定結果の重みを小さくすることでジッタの変動に素早く追従することが必要である。また、測定用パケットとあわせて、HA でのカプセル化したヘッダのタイムスタンプを利用してジッタ測定をする方法も考えられる。

4.6 実験 3: 通信速度が異なる経路での性能評価

評価環境 (図 7) において、経路 1 (path1)、経路 2 (path2) の下り通信速度を 128 kbps、経路 3 の下り通信速度を 384 kbps、AL が FTP サーバ (CN) から約 2 Mbytes のデータをダウンロードする際のスループットを 5 回計測し、その平均値を求めた。利用可能な複数経路の通信速度が異なる環境で、経路 3 (path3) のジッタのみを $\pm 0 \sim 200$ ms で変化させ、DQJ 方式によってジッタの大きい経路の判定を行い、より高いスループットを得られる経路の組合せを選択できるか確認した。経路 3 単独での平均スループットも比較のために示す。

4.7 実験 3 の評価結果

図 14 に結果を示す。DQ 方式、DQJ 方式ともジッタが ± 150 ms までは、3 経路の合計通信速度 (640 kbps) に近いスループットが得られる。これは、経路 1、経路 2 は、経路 3 に比べ通信速度が狭くパケット送信に時間がかかるため、経路 1、経路 2 へ振り分けるトラ

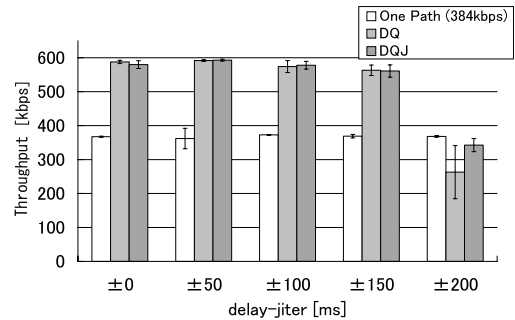


図 14 通信速度差がある場合の平均スループットの変化
Fig. 14 Throughput against delay-jitter in case of difference in quality.

フィックは経路 3 より少なく、経路 3 のジッタが多少変動しても全体としてのパケット到着順序の逆転はそれほど発生しないからであると考えられる。

一方、経路 3 のジッタが ± 200 ms に達すると、DQ 方式では 384 kbps の経路 3 のみを単独で使用したときよりも平均スループットが大きく低下する。DQJ 方式では、経路 3 単独のスループットよりやや劣るものの、ほぼ 384 kbps の経路 3 のみを単独で使用した場合と同等のスループットが得られる。これは、DQ 方式では経路間のジッタの違いが大きくてもすべての経路を利用するため、ジッタの大きな経路の影響でパケット到着順序の逆転が頻発しスループットが低下するのに対し、DQJ 方式では経路利用判定によって、より大きな通信速度を得られる経路 3 のみで通信を行うからである。また、DQ 方式の 95%信頼区間が大きいのは、5 回試行中、1 度だけジッタの影響を受けずに通信できた場合があったため、このことから、DQ 方式は経路間のジッタに大きな違いがあるときには安定したスループットが得られないことが分かる。

次に、経路 3 本でジッタ ± 200 ms に設定時の DQ 方式、DQJ 方式における各経路へのトラフィック送信量と AL の受信ウィンドウサイズ、ack ナンバについて詳細分析する。図 15 に DQ 方式の結果を、図 16 に DQJ 方式の結果を示す。

図 15 より、DQ 方式では、通信開始直後に各経路へトラフィックを送信することが分かる。その結果、受信側でパケット到着順序の逆転による重複 ack が頻発し、すぐに全体の送信量が減少する。その後、トラフィックは経路 3 のみに送信されるが、これはトラフィックの送信量が減少したため、DQ 方式で用いる各経路の仮想送信キューにパケットが貯まらず、つねに通信速度の大きな経路 3 を選択するからだと考えられる。経路 3 への送信量が増加し、経路 1, 2 へもトラフィックの送信が再開されると、パケット到着順序

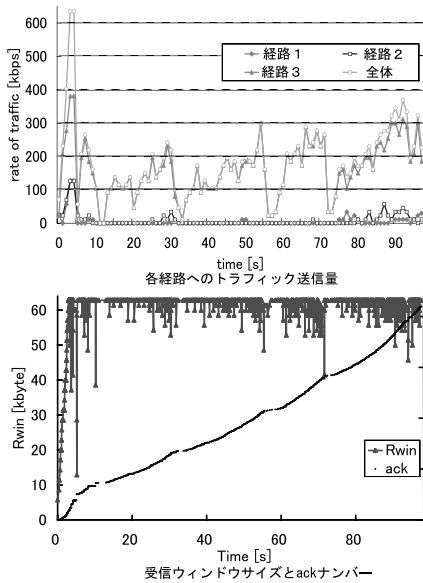


図 15 DQ 方式 : 128 kbps*2 + 384 kbps, ジッタ ± 200 [ms] での評価

Fig. 15 DQ in case of 128 kbps*2 + 384 kbps and delay-jitter 200 ms.

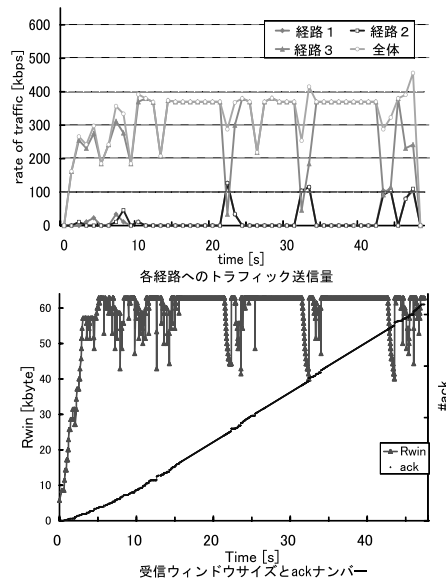


図 16 DQJ 方式 : 128 kbps*2 + 384 kbps, ジッタ ± 200 [ms] での評価

Fig. 16 DQJ in case of 128 kbps*2 + 384 kbps and delay-jitter 200 ms.

の逆転が再発するため、再び全体の送信量が減少する。以上の動作が繰り返されるため、DQ 方式では全体としてスループットが増加しない。

図 16 より、DQJ 方式では、通信開始直後から経路 3 しか利用していないことが分かる。これは、FTP 開始前までのジッタの測定値と経路利用判定により 3 本

の経路を同時に使用するとジッタによるパケット到着順序の逆転が頻発すると判断し、ジッタは大きい通信速度の大きな経路 3 を送信経路に選択したからである。ジッタが大きくても複数の経路を利用しなければパケット到着順序の逆転は発生しにくいいため、DQ 方式よりも安定した通信を行うことが分かる。DQJ 方式でも、一時的に経路 1 や経路 2 へトラフィックを送信することもあるが、全体の送信量が大きく低下するほどのパケット到着順序の逆転は発生していない。

以上より、経路の通信速度が異なる場合でも、DQJ 方式の経路利用判定が有効に動作し、トラフィック送信量の低下を防ぐことで安定したスループットを得られることが確認できた。

5. まとめと今後の課題

本稿では、Mobile IP SHAKE で遅延揺らぎ (ジッタ) を考慮した経路利用判定方式 (DQJ 方式) を提案し、実装評価した。仮想的な送信キューを用いて、遅延揺らぎによってパケット到着順の逆転を引き起こす経路を判断し利用経路から除外することで、複数経路間の遅延揺らぎの違いによって生じるパケット到着順の逆転を抑制し TCP 通信時の重複 ack による送信量の制限を緩和できることを確認した。

性能評価より、提案方式が有効となる条件として、複数経路中にジッタの大きな経路が混在し、その経路のジッタが頻繁に変動しないことがあげられる。この条件において、提案方式はその他の経路の合計通信速度に近いスループットを得ることができる。

また、本提案方式は、遅延揺らぎの大きな経路を一時的に利用しないよう制御した。さらなる性能向上のために、閾値を超えた経路を利用停止するのではなく、同一の複製パケットを送信し冗長性を持たせる副経路として利用することで、移動端末の接続性とスループットを向上させられると考える。この場合、先に到着したパケットは受信され ack を返信するが、後に到着したパケットは受信側で破棄し ack も返信しないため、重複 ack を引き起こすことはないと思われる。

今後、以上のような改善策を検討し経路集約型通信における接続性・スループットの向上を目指すとともに、経路品質測定において、上りと下りで遅延が異なる場合や揺らぎ変動の追従、ジッタ以外のパラメータに関しても検討が必要である。

参考文献

- 1) Sklower, K., Lloyd, B., McGregor, G., Carr, D. and Coradetti, T.: The PPP Multilink Pro-

- ocol (MP), RFC1990 (1996).
- 2) IEEE P802.3ad Link Aggregation Task Force. <http://grouper.ieee.org/groups/802/3/ad/>
 - 3) Traw, B. and Smith, J.: Striping within the Network Subsystem, *IEEE Network*, Vol.9, pp.22–32 (1995).
 - 4) Snoeren, A.: Adaptive Inverse Multiplexing for Wide-Area Wireless Networks, *Proc. IEEE GLOBECOM'99*, pp.1665–1672 (1999).
 - 5) Kamio, Y., Kojima, F. and Fujise, M.: Implementation and Performance Evaluation of 384 kbps-PHS Experimental System, *IEICE Trans.*, Vol.E83-B, No.8, pp.1844–1853 (2000).
 - 6) Wakikawa, R., Uehara, K. and Murai, J.: Multiple Network Interfaces Support by Policy-Based Routing on Mobile IPv6, *Proc. 2002 International Conference of Wireless Networks (ICWN'02)* (2002).
 - 7) Koyama, K., Ito, Y., Mineno, H. and Ishihara, S.: Performance Evaluation of TCP on Mobile IP SHAKE, *IPSJ Journal*, Vol.45, No.10, pp.2270–2278 (2004).
 - 8) Iyengar, J., Shah, K., Amer, P. and Stewart, R.: Concurrent Multipath Transfer Using SCTP Multihoming, *Proc. International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS'04)* (2004).
 - 9) Hsieh, H., Kim, K. and Sivakumar, R.: An End-to-End Approach for Transparent Mobility Across Heterogeneous Wireless Networks, *ACM Mobile Networks and Applications*, Vol.9, pp.363–378 (2004).
 - 10) Zhang, M., Lai, J., Krishnamurthy, A., Peterson, L. and Wang, R.: A Transport Layer Approach for Improving End-to-End Performance and Robustness Using Redundant Paths, *Proc. USENIX 2004*, pp.99–112 (2004).
 - 11) Rojviboonchai, K., Osuga, T. and Aida, H.: A Study on Rate-Based Multi-Path Transmission Control Protocol (R-M/TCP) Using Packet Scheduling Algorithm, *IEICE Trans.*, Vol.E89-D, No.1, pp.124–131 (2006).
 - 12) Goff, T. and Phatak, D.S.: Unified Transport Layer Support for Data Striping and Host Mobility, *IEEE Journal on Selected Areas in Communications*, Vol.22, No.4, pp.737–746 (2004).
 - 13) Phatak, D.S. and Goff, T.: A novel mechanism for data streaming across multiple IP links for improving throughput and reliability in mobile environments, *Proc. IEEE INFOCOM 2002*, Vol.2, pp.773–781 (2002).
 - 14) 林 孝典, 山崎真一郎, 森田直人, 相田 仁, 武市正人, 土居範久: インターネットを用いた複数経路データ伝送方式の性能評価, 信学論 B, Vol.J84-B, No.3, pp.514–522 (2001).
 - 15) Kim, K. and Shin, K.G.: Improving TCP performance over wireless networks with collaborative multi-homed mobile hosts, *Proc. 3rd international conference on Mobile systems, applications, and services (MobiSys'05)*, pp.107–120 (2005).
 - 16) Moriya, T., Ohnishi, H., Ogawa, T. and Ito, T.: Method of improving bandwidth and connectivity with multiple links in nomadic network environment, *Proc. Information and Telecommunication Technologies, Asia-Pacific Symposium (APSITT 2005)*, pp.41–46 (2005).
 - 17) Mineno, H., Ishihara, S., Ohta, K., Aono, M., Ideguchi, T. and Mizuno, T.: Multiple Paths Protocol for a Cluster Type Network, *International Journal of Communication Systems*, Vol.12, pp.391–403 (1999).
 - 18) Sivakumar, H., Bailey, S. and Grossman, R.L.: Pockets: The Case for Application-level Network Striping for Data Intensive Applications using High Speed Wide Area Networks, *Proc. ACM/IEEE conference on Supercomputing* (2000).
 - 19) Hacker, T. and Athey, B.: The End-to-End Performance Effects of Parallel TCP Sockets on a Lossy Wide-Area Network, *Proc. 16th IEEE-CS/ACM International Parallel and Distributed Processing Symposium* (2002).
 - 20) 舩田知広, 大木一将, 峰野博史, 石原 進: Mobile IPv6 を用いた通信回線共有方式における経路最適化の評価, 情報処理学会研究報告, モバイルコンピューティングとユビキタス通信, 2005-MBL-32, Vol.2005, No.28, pp.23–28 (2005).
 - 21) 川島佑毅, 峰野博史, 石原 進, 水野忠則: 経路集約通信におけるポリシーベースの分散手法切替型トラフィック制御の実装, マルチメディア・分散・協調とモバイル (DICOMO2004) シンポジウム論文集, pp.571–574 (2004).
 - 22) 川島佑毅, 峰野博史, 石原 進, 水野忠則: 経路集約通信における動的トラフィック制御の実装, 情報処理学会研究報告, モバイルコンピューティングとユビキタス通信, 2004-MBL-31, Vol.2004, No.114, pp.171–176 (2004).
 - 23) Schulzrinne, H., Casner, S., Frederick, R. and Jacobson, V.: RTP: A transport protocol for real-time applications, RFC 1889, IETF (Jan. 1996).
 - 24) Dynamics HUT Mobile IP. <http://dynamics.sourceforge.net/>

(平成 18 年 4 月 18 日受付)

(平成 18 年 11 月 2 日採録)

推薦文

本稿は、著者らの提案する複数経路を同時に利用する通信回線共有方式 (SHAKE) において、遅延揺らぎを考慮したトラフィック分配を行うことで高いスループットを実現する手法を提案している。モバイル環境において高品質な通信環境を提供するためには様々なネットワーク手段を状況に応じて効率的に利用することが求められるが、本研究の遅延揺らぎにより定義される経路品質を利用したトラフィック分配は実用的であり、MobileIP 上に実装した研究の成果は今後ますますの発展が期待される。

(モバイルコンピューティングとユビキタス通信研究会主査 渡辺 尚)



川島 佑毅 (正会員)

1982 年生。2004 年静岡大学情報学部情報科学科卒業。2006 年同大学大学院情報学研究科情報学専攻博士前期課程修了。同年三菱電機 (株) 入社。モバイルコンピューティング、ユビキタスネットワークに関心を持つ。



峰野 博史 (正会員)

1975 年生。1997 年静岡大学工学部情報知識工学科卒業。1999 年同大学大学院理工学研究科計算機工学専攻修士課程修了。同年日本電信電話 (株) 入社。NTT サービスインテグレーション基盤研究所配属。2002 年 10 月より静岡大学情報学部助手。2006 年九州大学大学院システム情報科学府博士 (乙) 取得。博士 (工学)。1999 年電子情報通信学会東海支部学生研究奨励賞、2001 年 NTT サービスインテグレーション基盤研究所長表彰受賞、2005 年、2006 年情報処理学会 DICO MO シンポジウムベストカンパニオン賞受賞。モバイルコンピューティング、ユビキタスコンピューティングに関する研究に従事。電子情報通信学会、IEEE、ACM 各会員。



石原 進 (正会員)

1972 年生。1994 年名古屋大学工学部電気学科卒業。1999 年同大学大学院工学研究科博士後期課程修了。1998 年日本学術振興会特別研究員。1999 年静岡大学情報学部助手。2001 年同大学工学部助教授、2006 年より同大学創造科学技術大学院助教授。博士 (工学)。1997 年電気通信普及財団テレコムシステム技術学生賞受賞。モバイルコンピューティング、無線環境用 TCP/IP、アドホックネットワークに関する研究に従事。電子情報通信学会、IEEE、ACM 各会員。



水野 忠則 (フェロー)

1945 年生。1968 年名古屋工業大学経営工学科卒業。同年三菱電機 (株) 入社。1993 年静岡大学工学部情報知識工学科教授、1996 年同大学情報学部情報科学科教授、2006 年より同大学創造科学技術大学院総長、教授、工学博士。情報ネットワーク、モバイルコンピューティング、放送コンピューティングに関する研究に従事。著訳書としては『コンピュータネットワーク概論』(日経 BP)、『モダンオペレーティングシステム』(ピアソン・エディケーション) 等がある。電子情報通信学会、IEEE、ACM 各会員。当会フェロー、幹事。