

## 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<sup>†</sup> 峰野 博史<sup>††</sup>  
石原 進<sup>‡</sup> 水野 忠則<sup>††</sup>

我々は複数ネットワークインター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毎に各経路へ振り分けた場合、各経路の実行スループットや遅延、遅延揺らぎの違いによって、通信相手でパケットの到着順序入替わりが頻発し、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し、到着順序入替わりが許容範囲を超える可能性のある経路へ振り分けない制御方式を検討した。また、検討方式を実装し評価を行い、従来方式のよりもパケット到着順序逆転を防ぐことができることを確認した。

### Evaluation of multiple paths distribution method based on delay-jitter

YUKI KAWASHIMA,<sup>†</sup> HIROSHI MINENO,<sup>††</sup> SUSUMU ISHIHARA<sup>‡</sup> and TADANORI MIZUNO<sup>††</sup>

This paper describes implementation and evaluation of multiple paths communication traffic distribution method based on delay-jitter (DQ). Multiple paths communication intended to the speed and reliability of wireless communication by using multiple Internet link mobile hosts simultaneously and dispersing the traffic between mobile hosts and a correspondent host. The performance of SHAKE decreases by out-of-order packets caused by delay-jitter. We implement the mechanism evading the performance decrease by not using path of large delay-jitter (DQJ). Performance comparison of DQJ with DQ using test-bed environment shows that DQJ is able to achieve a remarkable improvement in the end-to-end delay-jitter.

#### 1. はじめに

近年、端末の小型化、高機能化により、有線 LAN、無線 LAN、Bluetooth といった複数ネットワークインターフェース (NetIF) を搭載した移動端末も登場してきている。これら複数 NetIF を活用しネットワークの負荷を分散、接続性向上、通信速度向上を図る複数経路通信が注目を集めている<sup>1)2)3)</sup>。

筆者らは、複数経路を同時に利用する通信回線共有方式 SHAKE (SHARING multiple paths procedure for cluster network Environment) を提案している。SHAKE では近隣の移動端末を複数使用し、無線 LAN や Bluetooth 等の比較的高速な無線で相互接続して一時的な協力関係 (アライアンス) を構築する。そして、通信時に自分の持つ低速な長距離無線リンクだけでなく、アライアンス内の他端末の持つ外部リンクを利用する。

これまで SHAKE 実現の一手法として、Mobile IPv4 を応用した Mobile IP SHAKE を提案し評価を行ってきた<sup>4)</sup>。

Mobile IP SHAKE はネットワークレイヤで複数経路通信を実現することで、任意のアプリケーションで複数経路通信を可能にしている。トラフィックを複数経路に分散させる方法として、パケット毎の分配とフロー毎の分配の 2 種類の分配方式を可能とする制御機構を開発し評価した<sup>5)</sup>。その結果、パケット毎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、利用可能な全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する方がよいことがわかった。

本稿では Mobile IP 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仕組みについて検討する。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、遅延揺らぎを考慮した分配方式、経路品質を決定するパラメータと測定法について説明する。4 章では、エミュレーション環境において、従来方式と対案方式を比較評価した結果を示す。5 章では、本稿を総括し、今後の課題を示す。

<sup>†</sup> 静岡大学大学院情報学研究所  
Graduate School of Informatics, Shizuoka University

<sup>††</sup> 静岡大学情報学部  
Faculty of Informatics, Shizuoka University

<sup>‡</sup> 静岡大学工学部  
Faculty of Engineering, Shizuoka University

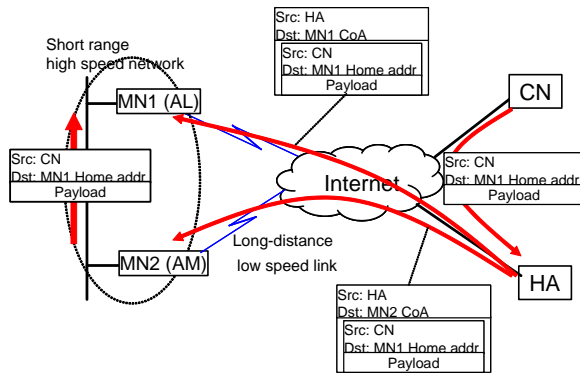


図1 Mobile IP SHAKE 動作概要

## 2. Mobile IP SHAKE

### 2.1 Mobile IP SHAKE 概要

以下に Mobile IP SHAKE の概要を示す。移動端末 (Mobile Node : MN) がアライアンス外部にいる任意の通信相手 (Correspondent Node: CN) と IP 層で SHAKE による通信を実現するために、Mobile IPv4 の経路最適化を行わない限り CN と MN 間の通信は必ずホームエージェント (Home Agent: HA) を通過する特性を利用し、HA にトラフィック分配機構を持たせた。HA でトラフィック分配をすることで、CN は分配のための特別な機構を持たずに複数経路を利用した通信が可能となる。

Mobile IP SHAKE の動作概要を 図 1 に示す。移動端末 (MN1) は自身の気付けアドレス (MN1 CoA) を HA に登録する。MN1 は近隣の移動端末 (MN2) とアライアンスを組み、MN2 の気付けアドレス (MN2 CoA) を HA に登録する。ここで、SHAKE を利用して通信を行う端末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を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帯域幅や遅延を考慮しながらトラフィックをパケット毎もしくはフロー毎に分配するトラフィック分配機構を持つ。AL と AM の各端末は論理帯域、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。

CN から AL への下りトラフィックでは、HA が通信ホスト (CN) から AL 宛のトラフィックを受信し、トラフィック分配機構が AL だけでなくカプセル化し AM にも転送する。AM はトラフィックを受信するとカプセル化を解除し AL に転送する。

このとき、分配手法がパケット毎の場合は、Terminal Profile に基づき各経路の帯域と遅延からパケット到着時間を予想し、パケットが順序通り到着するように送信す

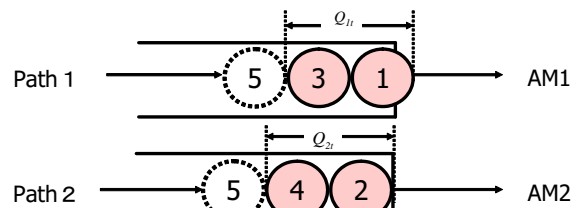


図2 仮想的な送信キュー

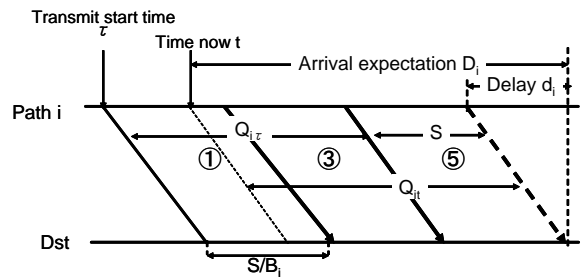


図3 パケット到着予想時間

る DQ(Delay Queuing) 方式<sup>4)</sup>を用いている。フロー毎の分配の場合は、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、帯域重視やパケットコスト重視の経路を選択する。

### 2.2 Delay Queueing(DQ) 方式

SHAKE による通信で TCP トラフィックをパケット毎に分配する場合、受信側へのパケット到着順を考慮しないとパケット到着順の逆転が頻発し、TCP の送信制御が働いて性能が低下してしまう。現在実装されているパケット毎の分配方式では、各経路の仮想的な送信キュー (図 2) の待ち時間と伝送遅延時間の合計時間が最短になる経路へ送信する DQ(Delay-Queueing) 方式を採用している。パケット到着時間予想時の仮想キューの状態を図 3 に示す。送信しようとしているパケットのパケットサイズを  $S$  [bit]、経路  $i$  の論理帯域を  $B_i$  [kbps]、送信中パケットの送信開始時刻  $\tau$ 、現在時刻  $t$  とすると、仮想キューに溜まっているキューのサイズ  $Q_i$  [bit] は式 (1) で表すことができる。さらに  $Q_i$  [bit] を送出するまでの時間に伝送遅延  $d_i$  [ms] を加えることで、パケット到着予想時間  $D_i$  [ms] を算出する (式 (2))。

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

$$D_i = \frac{Q_i}{B_i} \cdot 1000 + d_i \quad (2)$$

伝送遅延  $d_i$  の測定方法については、3.4 章にて説明する。

### 2.3 経路状況によるスループット低下

複数経路通信では、各経路の品質や状態が変化するため、ユーザがあらかじめ設定した静的なトラフィック制

御方式では期待したスループットが得られない場合もある<sup>5)</sup>。SHAKE では外部リンクに長距離無線の使用を想定している。長距離無線リンクでは、低レイヤでの誤り回復により遅延揺らぎが大きくなる。複数経路へパケット毎にトラフィックを分配する場合、遅延揺らぎが大きくなると、AL へのパケット到着順の逆転が頻発し、TCP の輻輳制御によりスループットが低下してしまう。フロー毎の分配では、経路の遅延揺らぎが増加しても全てのパケットが同一経路を流れるため到着順序の逆転が生じにくく、スループットはほとんど変化しないことがわかっている。ただし、複数経路を利用しないため経路の持つ論理帯域以上のスループットは得られない。

つまり、使用する経路状況に応じて動的にパケット分配、フロー分配で利用する経路を切り替えることができると複数経路を有効に活用した通信が実現できると考える。以降では、まず、複数経路通信が影響を受ける経路パラメータとその測定方法について検討する。次に、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. 遅延揺らぎを考慮した分配

#### 3.1 経路品質を表すパラメータ

HA で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、可用帯域、伝送遅延、遅延揺らぎ、パケットロス率の 4 つが挙げられる。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、帯域は 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ロスがない状態を想定する。このような想定環境のもと、TCP 通信を使用した Mobile IP SHAKE の遅延、遅延揺らぎが経路品質に与える影響を検討する。

まず遅延について検討する。Mobile IP SHAKE では DQ 方式によりパケット到着時間を推測している。そのため、各経路の遅延差によるパケット到着順序逆転の重複 ACK は発生しにくいと考えられる。ただし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。

次に遅延揺らぎについて考える。遅延揺らぎとは伝送

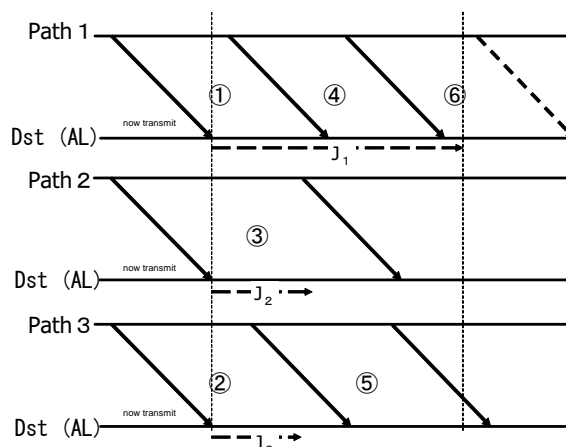


図 4 各経路のキューと遅延揺らぎ

遅延のバラツキ具合を表すものである。遅延揺らぎが大きくなると、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と、それだけ遅延揺らぎによるパケット到着順序逆転が発生しやすくなる。パケット到着順序逆転が頻発すると TCP を用いた通信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遅延揺らぎを算出することが可能である。そこで、得られた遅延揺らぎから使用する経路・使用しない経路を選択する方法を考えらるることができる。

#### 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。TCP 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 パケット以内ということになる。よって、各経路のキューに溜まっているトラフィック量と遅延揺らぎの大きさによりパケットの到着順序逆転が 3 つ以内に収まるような経路利用判定を行う。

#### 3.3 経路利用判定 (DQJ) 方式

DQ 方式で用いた仮想キューへ遅延揺らぎ (Delay-Jitter) を考慮することで経路利用判定を行う DQJ (DQ-Jitter) 方式について説明する。図 4 に DQJ 方式を使用した場合の各経路の送信キューと遅延揺らぎの状態を示す。

図 4 において、経路 1 で送信中のパケットが遅延揺らぎ  $J_1$  によって、到着順序逆転を起こす可能性のあるパ

ケットは、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は遅延揺らぎが大きく、スループット低下を招く恐れがあると判断できる。経路2の遅延揺らぎ $J_2$ 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。

ここで経路 $i$ が $t$ [ms]間に送出するパケットの数を $P_i(t)$ とすると、経路 $i$ が経路 $j$ の遅延揺らぎ $J_j$ [ms]間に送信するパケット数は $P_i(J_j)$ となる。経路 $j$ の遅延揺らぎ $J_j$ [ms]間にすべての経路から送出されたパケットが3パケット以下になる経路利用の遅延揺らぎの閾値は、式(3)で表すことができる。

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

式(3)を用いて、経路 $j$ の遅延揺らぎ $J_j$ が閾値を超えている間、HAは経路 $j$ へのトラフィック送信を停止する。

### 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るために、HA-AL間、HA-AM間の遅延、遅延揺らぎ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<sup>6)</sup>。

パッシブ測定は、余計なトラフィックを増やさず測定可能である。ただし、トラフィックの無い経路での測定が出来ない。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。利用中の経路しか測定できないパッシブ測定より全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していると考えられる。

図5にアクティブ測定を用いた遅延と遅延揺らぎの測定方法を示す。HA-AL間、HA-AM間の各経路間で $n$ 番目の測定用パケットには、送信者であるHAのアドレス、送信先のアドレス、シーケンスナンバー $n$ 、タイムスタンプ $S_n$ が含まれる。

AL, AMは測定用パケットを受信すると、RTP(RFC1889)<sup>7)</sup>の遅延揺らぎ測定方法に基づき、経路の片道遅延を算出する。受信パケットの送信時タイムスタンプ $S_n$ と到着時のタイムスタンプ $R_n$ から、2つのパケット $n-1$ と $n$ のパケットに関して、パケット転送遅延時間の差(片道遅延の差) $G$ は以下のように表される(式(4))。

$$G_{n-1,n} = (R_{n-1} - S_{n-1}) - (R_n - S_n) \quad (4)$$

遅延揺らぎ $J$ は、パケット $n$ と1つ前のパケット $n-1$

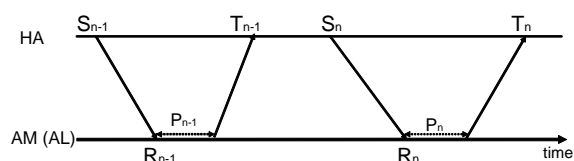


図5 遅延、遅延揺らぎ測定手法

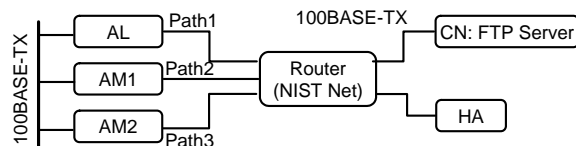


図6 実験環境

の $D$ を用いて下式のように表される(式(5))。

$$J_n = J_{n-1} + \frac{(|G_{n-1,n}| - J_{n-1})}{16} \quad (5)$$

AL, AMは式(5)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 $T_n$ から片道伝送遅延 $d_n$ を計測する(式(6))。

$$d_n = \frac{(T_n - S_n) - p_n}{2} \quad (6)$$

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(7))。

$$d = \alpha \cdot d_{old} + (1 - \alpha) \cdot d_n \quad (0 < \alpha < 1) \quad (7)$$

式(5), (7)で各経路において求めた遅延揺らぎ $J$ 、遅延 $d$ を式(2), (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Dynamics<sup>8)</sup>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。本稿では本実装を利用し、図6に示すように3台の端末でアライアンスを構築した。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384kbps、上り64kbps、片道遅延200ms、遅延揺らぎ $\pm 50$ msの一様分布で発生するように設定した。

現在の実装では、ack返送路はALの持つ外部リンクを使用している。また、測定用パケットの送信間隔は1秒とし、TCPのRTT平滑化係数を参考に遅延の平滑化係数 $\alpha$ を0.8とした。

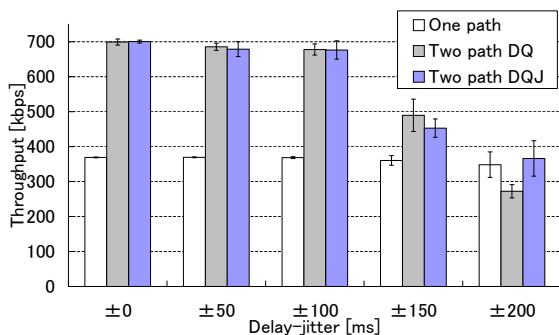


図7 遅延揺らぎによるスループットの変化 (経路 2 本)

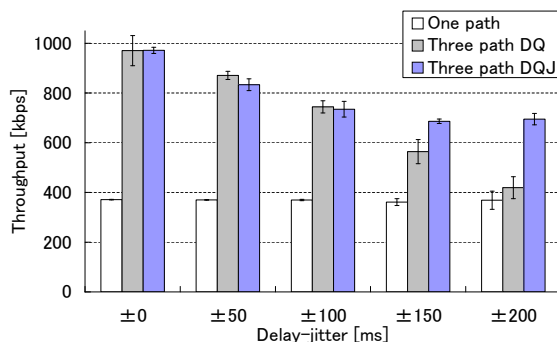


図8 遅延揺らぎによるスループットの変化 (経路 3 本)

#### 4.2 実験 1

評価環境(図6)において,FTPサーバ(CN)からALへ約2Mbytesのデータ転送したときのスループットを5回計測し,その平均を求めた.各経路の遅延揺らぎを以下のように設定することで,遅延揺らぎがDQ方式のスループットに与える影響と,DQJ方式が遅延揺らぎ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出来るかを確認する.比較のために単一経路での平均スループットを示す.

設定1: 経路1, 経路2を使用, 経路2のみ遅延揺らぎを±0~200msで変化.

設定2: 経路1, 経路2, 経路3を使用, 経路3のみ遅延揺らぎを±0~200msで変化.

#### 4.3 実験1の評価結果

経路2本使用時の平均スループットを図7に, 経路3本使用時の平均スループットを図8に示す. 図7において, DQ方式, DQJ方式とも理想的な環境である遅延揺らぎの無い±0msでは単一経路のときのおよそ2倍のスループットが得られている.そして, ±50ms, ±100msまではDQ方式, DQJ方式の両方式において遅延揺らぎによるスループット低下はほとんど見られない.

遅延揺らぎ±150msではDQ方式, DQJ方式ともに大幅なスループットの低下が生じている.今回の環境では, 遅延揺らぎ±100~150msの間に性能変化の境目があり, ±150msでは経路2の遅延揺らぎにより大幅なスループットの低下が発生するが, 完全に単一経路へ切り替えるよりも高いスループットが得られている. DQ方式とDQJ方式にスループットの差が見られるのは, DQJ方式では遅延揺らぎによるスループット低下に加えて, 経路利用判定により遅延揺らぎの大きな経路2への送信を行わないためであると考えられる.

遅延揺らぎ±200msでは, DQ方式は単一経路を使用するよりもスループットが低下している.これに対し, DQJ

方式では判定により遅延揺らぎの少ない経路へのみ送信することで, 単一経路を使用するよりもスループットの低下を抑えられていることが分かる. DQJ方式が単一経路よりもスループットが理由は, 遅延揺らぎの状況によって, 2本の経路を利用しているためである.

次に, 図8で経路3本使用時の平均スループットについて考察する. DQ方式, DQJ方式とも理想的な環境である遅延揺らぎの無い±0msでは単一経路のときのおよそ3倍弱のスループットが得られている. ±50ms, ±100msまではDQ方式, DQJ方式の両方式において遅延揺らぎによるスループット低下が起こっているが, 方式による差は見られない.

遅延揺らぎが±150msを超えると, DQ方式のスループットが低下しているのに対して, DQJ方式のスループットはおよそ700kbps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方式の判定が遅延揺らぎの大きな経路への送信を抑え, 遅延揺らぎの小さな経路を2本で通信を行っているためである.そのため, 遅延揺らぎの大きな経路3を利用せず重複ackの頻発によるスループットの低下を抑えられていることがわかる.

ここで, 経路3本, 遅延揺らぎ±200msでのDQ方式, DQJ方式における各経路へのトラフィック送信量, ALの受信ウィンドウサイズとackナンバーについて詳細分析する. 図9, 10でDQ方式について示し, 図11, 12でDQJ方式について示す.

図9より, DQ方式では遅延揺らぎの大きな経路3へもトラフィックを送信している.その結果, パケット到着順序逆転が頻発し, 図10のように重複ackが頻繁に送信されている.また, TCPは既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全体への送信量は低下している.

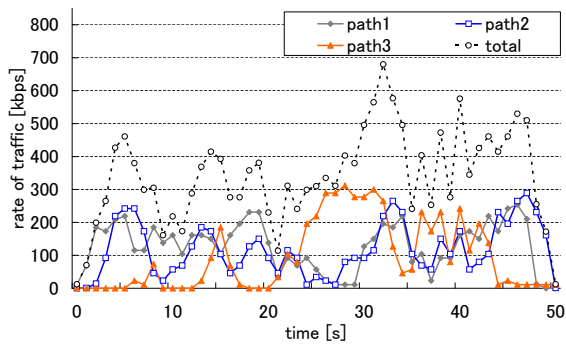


図 9 DQ 方式での各経路への送信量:±200[ms]

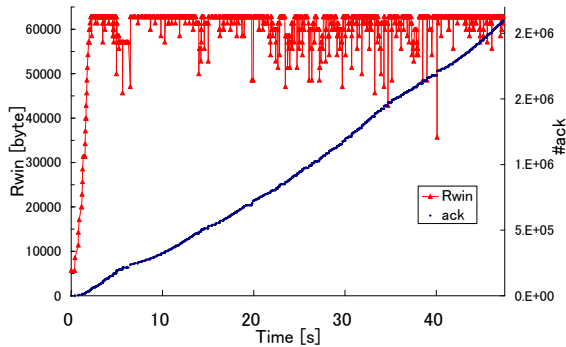


図 10 DQ 方式での受信ウィンドウサイズと ack ナンバー:±200[ms]

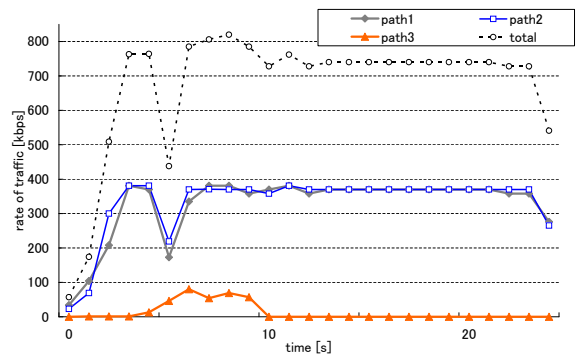


図 11 DQJ 方式での各経路への送信量:±200[ms]

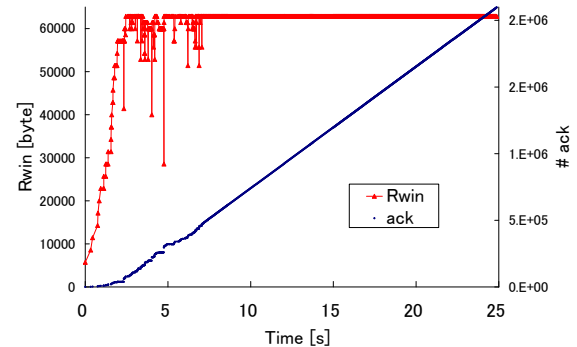


図 12 DQJ 方式での受信ウィンドウサイズと ack ナンバー:±200[ms]

表 1 経路 3 の遅延揺らぎ

time [s]	0	10	20	30	40 ~
jitter [ms]	50	100	150	200	50

図 11 より、経路 3 へトラフィックを送信時には重複 ack が発生してしまい一時的なスループット低下が見られるが、DQJ 方式は経路 3 への送信が抑制遅延揺らぎの小さな経路 1, 2 のみを利用し、図 12 より安定した通信を行っていることが確認できる。

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することで、パケット到着順序逆転による重複 ack 発生を抑えることが確認できた。

#### 4.4 実験 2

移動環境を想定し、遅延揺らぎが動的に変化する設定にて DQJ 方式の有効性を評価した。評価環境 (図 6) にて、経路 1, 経路 2 の遅延揺らぎを ±50ms とし、経路 3 遅延揺らぎのみを表 1 のように変化させ、FTP サーバ (CN) から AL へ約 5Mbytes のデータ転送する際の DQ 方式と DQJ 方式の各経路へのトラフィック量、ack ナンバーと受信ウィンドウサイズを推移を比較した。

#### 4.5 実験 2 の評価結果

図 13, 図 14 に DQ 方式の結果を図 15, 16 に DQJ 方式の結果を示す。また図 17 に遅延揺らぎの測定結果を示す。

図 13, 14 において、通信開始後 20 秒までは、経路 3 の遅延揺らぎが ±50ms, ±100ms と小さいため、各経路へ安定してパケットを分配しているため全体への送信量も 800 ~ 1000kbps と多い。しかし 20 秒経過後、経路 3 の遅延揺らぎが ±150ms になると重複 ack の数が多く、受信ウィンドウサイズも小さくなり、全体のトラフィック送信量が低下している。遅延揺らぎが ±200ms 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までなっている。これは、パケット到着順序逆転が頻発したことにより、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 秒経過後に遅延揺らぎが ±50ms に戻ると、徐々に全体のトラフィック量が増加している。これは、±200ms 時に下がりきってしまった送信ウィンドウがスロースタートより徐々にウィンドウサイズが大きくなっているためと考えられる。

図 15, 16, 17 において、通信開始後は、遅延揺らぎが小さいため経過時間毎に全体への送信量が増加している。しかし、遅延揺らぎの設定値が ±100ms になり、遅延揺



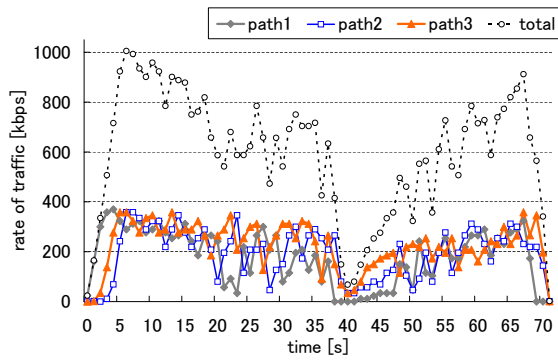


図 13 DQ 方式での各経路への送信量

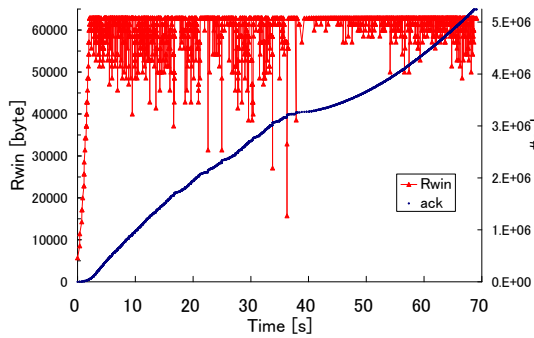


図 14 DQ 方式での受信ウィンドウサイズと ack ナンバー

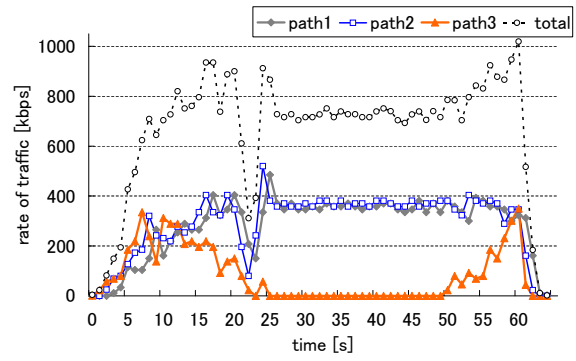


図 15 DQJ 方式での各経路への送信量

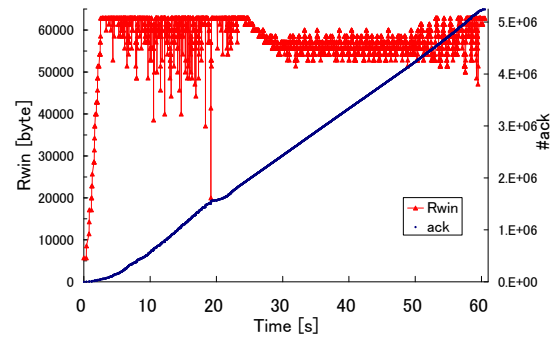


図 16 DQJ 方式での受信ウィンドウサイズと ack ナンバー

らぎの測定値が 100ms に近づくと経路 3 への送信量が低下している。

20 秒経過後、遅延揺らぎが  $\pm 150\text{ms}$  になると、経路 1、経路 2 への送信量も一時的に低下しその後安定した状態になっている。これはこの時点では経路 3 へのトラフィック送信は DQJ 判定により軽減されているが、送信されたパケットの到着順序逆転による送信量抑制が起こり全体のトラフィック送信量が低下したと考えられる。DQJ 判定で閾値を超えた場合に、送信を停止するのではなく、閾値を超えた経路に送信したパケットを別の経路でも送信し、冗長性を持たせることで、他の経路へのトラフィック送信量低下を抑えることができると考えられる。

遅延揺らぎの測定値が 150ms に近づくと、経路 3 へのトラフィック送信は停止され、揺らぎの小さな経路 1、2 のみを使用する安定した通信を行っている。40 秒経過後に遅延揺らぎが  $\pm 50\text{ms}$  に戻っても、遅延揺らぎの測定値が対応できていないためすぐには経路 3 への送信を再開しない。50 秒経過後の時点で経路 3 へのトラフィック送信を開始することで全体の送信量が増加している。

以上より、DQ 方式が遅延揺らぎの変化に対応できずに、スループットが低下してしまうのに対し、DQJ 方式は遅延揺らぎの大きな経路への送信をしないことで、大幅なスループット低下を防ぐことができている。ただし、

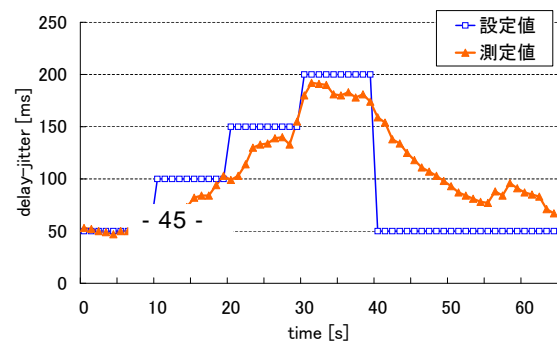


図 17 遅延揺らぎの測定結果

遅延揺らぎの測定に 10 秒程度要するため、その間は適切な判定が行えない。

## 5. まとめと今後の課題

本稿では Mobile IP SHAKE で遅延揺らぎを考慮した経路利用判定方式を提案、実装、評価した。仮想的な送信キューで遅延揺らぎによるパケット逆転を引き起こす経路を判断し、複数経路通信から除外する機能を追加し、極端な遅延揺らぎの発生している経路を利用しないことで性能低下が防止できることを確認した。

本提案方式では、遅延揺らぎの大きな経路を一時的に利用しないようにしているが、冗長パケットを送信する

ことで、一時的なトラフィックの低下を抑えることができる。今後、このような改善策を検討し、複数経路通信におけるスループットの向上を目指す。

#### 参 考 文 献

- 1) Hiroshi Mineno, Susumu Ishihara, Ken Ohta, Masahiro Aono, Tetsuo Ideguchi, Tadanori Mizuno. Multiple paths protocol for a cluster type network. *International Journal of Communication Systems*, Vol. 12, pp. 391–403, Dec 1999.
- 2) 長谷川洋平, 山口一郎, 浜崇之, 下西英之, 村瀬勉. End-to-End の通信品質を改善する複数経路通信方式の提案と無線環境における評価. 電子情報通信学会, CQ2004-104, MoMuC2004-78.
- 3) Hiroyuki Koga, Hiroaki Haraguchi, Katsuyoshi Iida, and Yuji Oie. A framework for network media optimization in multihomed qos networks. *DIN '05: Proceedings of the 1st ACM workshop on Dynamic interconnection of networks*, pp. 38–42, 2005.
- 4) Kenji Koyama, Yousuke Ito, Susumu Ishihara, Hiroshi Mineno. Performance evaluation of TCP on Mobile IP SHAKE. *Proc. of 1st International conference on Mobile Computing and Ubiquitous Networking (ICMU2004)*, pp. 8–13, Jan 2004.
- 5) 川島佑毅, 峰野博史, 石原進, 水野忠則. 複数経路通信におけるポリシーベースの分配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 (DICO2004) シンポジウム論文集, pp. 571–574, 2004.
- 6) 住本順一, 間瀬憲一, 横井忠寛. インターネットの品質・トラヒック管理〔II〕—IP 層の測定項目・測定手法を中心にして— 電子情報通信学会誌, Vol. 82, No. 12, pp. 1256–1263, 1999.
- 7) H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A transport protocol for real-time applications. RFC 1889, IETF, January 1996.
- 8) Dynamics HUT Mobile IP.  
<http://dynamics.sourceforge.net/>.