

Mobile IPv6 のハンドオフ性能評価

大石 巧 伊達 哲 山田 真理子 矢野 正

日立製作所 中央研究所 〒185-8601 国分寺市東恋ヶ窪 1-280

E-mail: {takumi-o, akira, m-yamada, m-yano}@crl.hitachi.co.jp

あらまし Mobile IPv6 は第 3 世代以降の移動体通信網, 公衆無線 LAN アクセス網などへの適用が考えられ, 応用範囲の広い技術である. アプリケーションから見ると Mobile IPv6 によって移動端末への接続性が保障される. しかし, 動画像配信などのストリーム系アプリケーションにとっては接続性だけでなく, ジッタ, 遅延時間等も重要である. なかでもハンドオフ時の通信途絶時間の長短はストリーム系アプリケーションにとって最重要である. そこで, アプリケーションから見た Mobile IPv6 のハンドオフ性能を評価するため, 東京都下と横浜市にある 2 つの実験室間を IPv6 で結び, 広域ハンドオフ実験を行った. この実験結果について報告する.

キーワード Mobile IPv6, handoff, latency

Performance Measurement of Mobile IPv6 Handoff

Takumi OISHI Akira DATE Mariko YAMADA and Masashi YANO

Central Research Laboratory, HITACHI Ltd. 1-280 Higashikoigakubo, kokubunji-shi, 185-8601 Japan

E-mail: {takumi-o, akira, m-yamada, m-yano}@crl.hitachi.co.jp

Abstract Mobile IPv6 is the widely applicable technique to the third or later generation mobile telecommunication networks, public wireless access (hotspot) networks and so on. Mobile IPv6 guarantee applications connectivity with mobile nodes. Not only connectivity but also jitter and latency, however, are important for stream data applications such as video stream delivery. Especially, it is most important for stream data applications how long the communication have been cut off at handoff. We therefore measured performance (latency at handoff) of Mobile IPv6 from viewpoint of applications, and report the result. We linked two our laboratories located at Tokyo and Yokohama with IPv6, and executed handoff.

Keyword Mobile IPv6, handoff, latency

1. はじめに

近年 IJ[1]や NTT コミュニケーションズ[2]等が IPv6 の商用サービスを開始し, IPv6 のネットワーク基盤は整いはじめている. また, 端末やサーバ装置の OS に関しても Windows XP をはじめ, Linux, BSD 系 OS 等 IPv6 が標準で使用できるようになってきている. したがって, IPv6 対応アプリケーションさえあればユーザはすぐにも IPv6 で通信できる環境が簡単に手に入るようになりつつある.

IPv6 の特徴の 1 つに Mobile IP 導入の簡便さが挙げられる. これは, 移動先における端末の一時的アドレス(気付けアドレス, 以下 CoA と呼ぶ)の付与に特別なネットワーク機器を必要とせず, 通常の IPv6 対応ルータで十分なことが大きな理由である. これにより, 移動端末(以下 MN と呼ぶ)のほかにホームエージェント(以下 HA と呼ぶ)という MN のアドレス管理を行う装置

を用意するだけで Mobile IPv6 が利用できるようになった.

2003 年 1 月 13 日現在, IPv6 における Mobile IP(以下 MIP6 と呼ぶ)は IETF[3]において標準化途中であるが, 第 3 世代以降の移動体通信網[4][5], 公衆無線 LAN アクセス網[6]などへの適用が考えられ, 応用範囲の広い技術である.

この MIP6[7]は, アプリケーションに対して MN がネットワークどこにいても通信が可能となるように, 接続性(connectivity)を保障する. しかし, 動画像配信や VoIP(Voice over IP)など遅延に敏感なストリーム系アプリケーションにとっては接続性だけでは不十分である.

そこで, MIP6 のハンドオフ時の性能を測定し, ストリーム系アプリケーションと MIP6 の親和性を明らかにすることを考えた. ここでいう親和性とは MIP6 をア

アプリケーションが利用する場合に、アプリケーションにどのような制約が存在するか、という意味である。まったく制約がないことが理想であるが、現時点ではこれが可能かどうか未知数である。ただし、IPv6 はこれから一般に普及していくものであるから、過去のソフトウェア資産に対する互換性は IPv4 ほど必要ではない。したがって、MIP6 を利用する際にアプリケーションが守る必要がある制約条件を明らかにすることができれば、今後のアプリケーション開発に大変有用であると報告者らは考えている。

今報告では MIP6 の性能として特にハンドオフ処理に要する時間を取り上げる。これを本報告ではハンドオフ時の遅延時間と呼ぶことにする。ストリーム系アプリケーションは、ハンドオフ中にプログラムが停止しないようにする必要があるが、もし MIP6 が無瞬断でハンドオフできないならばアプリケーション自身で対策する必要がある。

2. アプリケーションからみたハンドオフに要する時間の測定

2.1. ネットワークトポロジーと使用機器

図1に本報告での実験ネットワークトポロジーを示す。図1にあるように、東京(国分寺)実験室と横浜(戸塚)実験室間を IPv6 で結んだ。東京-横浜間のホップ数は3である。

移動端末(MN)1台、通信相手(以下 CN と呼ぶ)1台、無線 LAN 基地局(以下 AP と呼ぶ)3個を東京(国分寺)の実験室に設置し、ホームエージェント(HA)を東京、横浜(戸塚)の実験室に各1台設置した。東京の HA(図1の HA-B)を用いて東京実験室内に閉じた実験を行い、横浜の HA(図1の HA-A)を用いて横浜実験室内の PC を東京実験室内に持ち込んだと想定して実験を行った。

MN の無線アクセス手段は最も手軽な市販の無線 LAN(IEEE802.11b 準拠)を利用した。

実験で使用した MN, CN, HA の各機器のスペックを表1に示す。また、各機器の使用 OS 等を表2に示す。表2のように MIP6 スタックとしては、端末やサーバに USAGI プロジェクト[8]、HA に KAME プロジェクト[9]のものを利用した。MIP6 のドラフトのバージョンは USAGI, KAME ともに 15 である。

また、遅延時間測定には netperf[10]を使用した。netperf のバージョンは 2.2p12 である。

CN にも MIP6 のスタックを導入した理由は、導入しない場合に CN と MN 間で netperf が動作しなかったためである。これは通常の linux カーネルが、IPv6 拡張ヘッダの1つである destination option header に存在する、MIP6 拡張オプション(home address option)を理解できないため、該当パケットの処理がうまくい

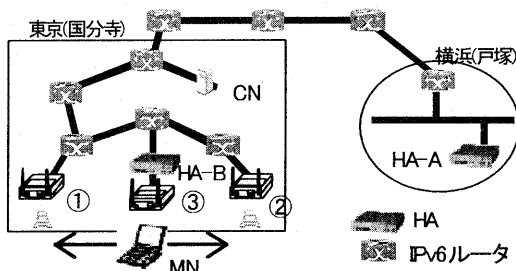


図1. 実験ネットワークトポロジー

表1. 実験機器一覧

機器	CPU	メモリ量	HD 容量
MN	PenIII 448MHz	128MB	6GB
CN	Xeon 1388MHz	512MB	35GB
HA-A	PenIII 448MHz	384MB	3.3GB
HA-B	PenIII 1GHz x 2	1GB	6.5GB

表2. 使用 OS と Mobile IPv6 スタック

機器	OS	Mobile IPv6 スタック
MN	Redhat7.2	USAGI 2002/5/24 snapshot
CN	Redhat7.2	USAGI 2002/5/24 snapshot
HA-A	FreeBSD4.5	KAME 2002/4/15 snapshot
HA-B	FreeBSD4.5	KAME 2002/4/15 snapshot

かないためであると考えられる。

CN にも MIP6 を導入したことで、CN と MN 間の通信経路を最適化(以下 R0 と呼ぶ)することが可能であるが、本実験においては R0 せず、常に HA 経由で通信することとした。R0 によって CN から MN へのパケットは HA を介さずに送られるため、本実験のネットワーク構成では横浜の HA を用いる意味がなくなってしまうためである。

図1のネットワークトポロジーでの RTT(Round Trip Time)は東京実験室内 CN-MN 間が 10 ミリ秒程度、横浜 HA 経由 CN-MN 間が 30 ミリ秒程度である。

2.2. ハンドオフ時の遅延時間の定義

本報告においては、ハンドオフ時の遅延時間は、ハンドオフ直前の最後のパケットを送信(再送分は除外)または受信してから、ハンドオフ直後に最初のパケットを受信するまでの経過時間と定義する。ここでいうパケットは netperf が送受信しているパケットであり、MIP6 や ICMP のパケットは除外する。この定義の具体例は 4.1 節の測定結果で示す。

遅延時間を上記のように定義した理由は、はじめに述べたように、アプリケーションからみた MIP6 のハンドオフによる遅延時間が知りたいためである。

また、実験の目的からすると本来ならば CN または MN で起動している netperf のプログラムにおいてタイムスタンプを表示させるべきであるが、実験の簡便さを考慮して、CN や MN において tcpdump でパケットをキャプチャし、代用することとした。

3. ハンドオフ時の遅延時間の測定方法

netperf はサーバ、クライアント型のアプリケーションである。クライアントからサーバに向かってパケットを送信して測定するため、MN をサーバとし、図 1 の CN をクライアントとした。

CN から MN へ向かって測定した理由は、一般的に移動端末(MN)へネットワークからデータをダウンロードする使用法が普通だろうと考え、それに合わせたためである。

netperf の IPv6 での測定の種類は STREAM タイプ(スループット測定に向く)と RR(Request-Response)タイプ(トランザクション測定に向く)がある。今回は TCP の request-response(RR)タイプを使用した。

また、測定のため送受信するデータ長は 1024 バイト、1回の測定は 30 秒間とし、各条件のもとで 5 回づつ試行した。この 30 秒の測定時間に 1 回ハンドオフさせ、上記で定義した遅延時間を測定した。

2.1 節で述べたように東京実験室内に AP を用意したため、狭い室内に複数の AP が存在する。したがって、MN の移動によるハンドオフが不可能なため、AP からの電波を遮断してハンドオフを擬似した。

3.1. 実験 1: 東京ローカルハンドオフ

実験 1 として東京実験室内で閉じた、ハンドオフ時の遅延時間を測定した。この測定時のパケット経路を図 2 に示す。図 2 は図 1 の東京(国分寺)実験室内部分を模式化したものである。図中、矢印の向きにパケットは流れる。

MN が 2 つの AP 間を行ったり来たりする場合に AP 間切り替え、すなわちハンドオフ時の遅延時間を測定した。

はじめ MN が AP①に接続している場合の詳細手順を以下に示す。

1. CN と MN で tcpdump 起動
2. CN から MN へ向けて netperf 起動

```
netperf -t TCPIPv6_RR -H "MN アドレス"
-1 30 -d -w 50 -- -r "データ長"
```
3. AP②のアンテナをとりつける
4. AP①のアンテナをはずし電波を遮断
5. netperf が終了したら tcpdump のログを回収

以下 AP①と②を入れ替えながら繰り返す。

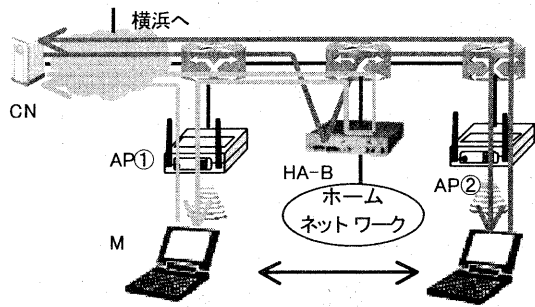


図 2. 東京実験室内ネットワーク

3.2. 実験 2: 東京—横浜広域ハンドオフ

実験 2 では、横浜実験室内の HA を使用し、ハンドオフ時の遅延時間を測定した。測定時のパケット経路を図 3 に示す。図 3 は図 1 を模式化したもので、東京実験室内部分は図 2 と同一である。図内の矢印は、パケットの流れる経路を示している。また、実験手順も実験 1 と同様である。

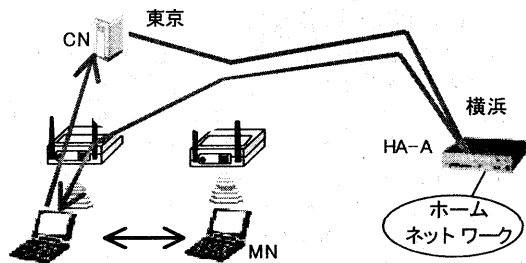


図 3. 東京—横浜ネットワーク

3.3. 実験 3: ルータ広告の MIPv6 拡張を使用した東京—横浜広域ハンドオフ

実験 1 と実験 2 では、IPv6 ルータとして市販のルータを利用したため、ルータ広告の広告周期に関して、MIPv6 拡張を利用できなかった。

そこで、FreeBSD4.5 と KAME スタックを用いて PC ルータを構成し、ルータ広告の周期を短くしてハンドオフ時の遅延時間を測定した。

MIPv6 において、ルータ広告は MN が接続先ネットワークの切り替えを知るトリガーとなる。よって、ルータ広告の間隔が短いほどすばやくハンドオフを検出できることになる。通常ルータ広告の最小間隔は 3 秒から 4 秒の範囲のランダム値 [11] である。MIPv6 拡張ではこれが、0.05 秒から 1.5 秒の範囲の値に小さくできる [7]。

本実験では、ルータ広告の間隔を 1 秒固定と 0.1 秒固定の 2 通りで測定した。

測定時のパケット経路は図3と同一である。ただし、東京実験室内部分は図2と異なっており、図4に示す。図2でHA-Bがあった位置に上記PCルータを置き、その先にAP③を設置した。実験手順は実験1と同様であるが、ハンドオフはAP①からAP③の方向だけで行い、遅延時間を測定した。

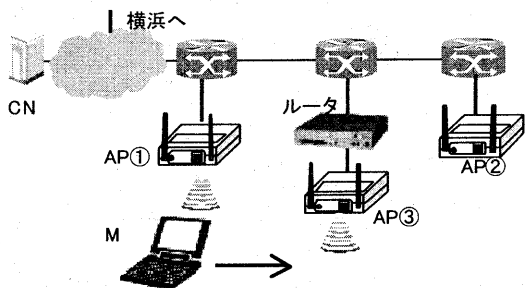


図4. 実験3での東京実験室内ネットワーク

4. ハンドオフ時の遅延時間測定結果

4.1. 遅延時間の計算方法

図5にCNでtcpdumpによりパケットをキャプチャした結果の例を示す。

netperfのTCPIPv6_RRタイプの測定では1トランザクションあたり3パケットやり取りされる(3ハンドシェイク)[12]。図中のNo. 449のパケットは3パケットのうち、最初のパケットであるが、これがNo. 450, 451, 452, 453, 455と5回再送されている。すなわち、No. 449のパケットが送信された時点ですでにハンドオフが始まっていたと考えられる。その1つ前のNo. 448のパケット送信の時点ですでにハンドオフが始まっていたとも考えられるが、誤差はせいぜい50ミリ秒弱である。

その後、No. 456のパケットが5回再送されたパケットの応答として受信されている。この時点でハンドオフが終了したと確認できる。実際にはNo. 454のパケッ

ト(MIPv6のBinding Updateメッセージ、以下BUと呼ぶ)が届いた時点でハンドオフがすでに終了していると確認できるが、アプリケーションから見ると、No. 456のパケットで初めて確認できる。結果として、ハンドオフに要した時間、すなわち遅延時間は、

$$29.015390 - 21.257981 \approx 7.8 \text{ (秒)}$$

となる。

4.2. 測定結果

前節にしたがって、パケットをキャプチャしたログから計算した遅延時間の値を図6から図8に示す。それぞれ、図6は実験1、図7は実験2、図8は実験3の結果である。

ホップ数が増えた影響と思われるわずかな遅延時間の増加はあるが、図7は図6と同様の傾向がうかがえる。だいたい8秒前後ハンドオフに要しており、長いときには14秒程度必要である。図6においては4回目に1.7秒という短い時間でハンドオフしているが、これについては次節で考察する。

図8を見ると、図6,7とくらべて概ね遅延時間が短くなっているが、一方で15秒程度必要な場合もある。これらバラツキの要因についても次節で考察する。

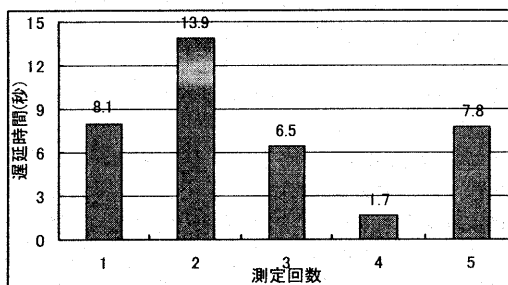


図6. 実験1の測定結果

No.	Time	Source	estination	rotoc	Info
446	21.187982	3ffe::3	3ffe::8	TCP	32878 > 1132 [PSH, ACK] Seq=2260456982 Ack=2245976524 win=3376
447	21.211389	3ffe::3	3ffe::8	TCP	1132 > 32878 [PSH, ACK] Seq=2245976524 Ack=2260458006 win=3376
448	21.211416	3ffe::3	3ffe::8	TCP	32878 > 1132 [ACK] Seq=2260458006 Ack=2245977548 win=3376
449	21.257981	3ffe::3	3ffe::8	TCP	32878 > 1132 [PSH, ACK] Seq=2260458006 Ack=2245977548 win=3376
450	21.507992	3ffe::3	3ffe::8	TCP	32878 > 1132 [PSH, ACK] Seq=2260458006 Ack=2245977548 win=3376
451	22.007973	3ffe::3	3ffe::8	TCP	32878 > 1132 [PSH, ACK] Seq=2260458006 Ack=2245977548 win=3376
452	23.007988	3ffe::3	3ffe::8	TCP	32878 > 1132 [PSH, ACK] Seq=2260458006 Ack=2245977548 win=3376
453	25.007994	3ffe::3	3ffe::8	TCP	32878 > 1132 [PSH, ACK] Seq=2260458006 Ack=2245977548 win=3376
454	27.071575	3ffe::3	3ffe::8	IPv6	Mobile IPv6 Destination Option
455	29.007993	3ffe::3	3ffe::8	TCP	32878 > 1132 [PSH, ACK] Seq=2260458006 Ack=2245977548 win=3376
456	29.015390	3ffe::3	3ffe::8	TCP	1132 > 32878 [PSH, ACK] Seq=2245977548 Ack=2260459030 win=3376
457	29.234106	3ffe::3	3ffe::8	TCP	1132 > 32878 [PSH, ACK] Seq=2245977548 Ack=2260459030 win=3376
458	29.234161	3ffe::3	3ffe::8	TCP	32878 > 1132 [ACK] Seq=2260459030 Ack=2245978572 win=3376

図6. tcpdumpのパケットキャプチャログ(ethereal[13]で表示)

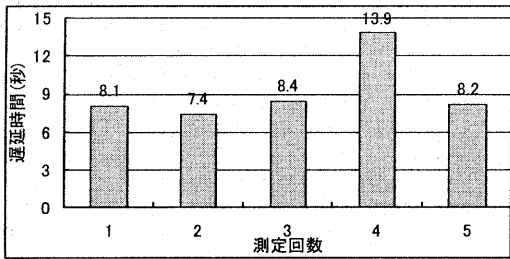


図7. 実験2の測定結果

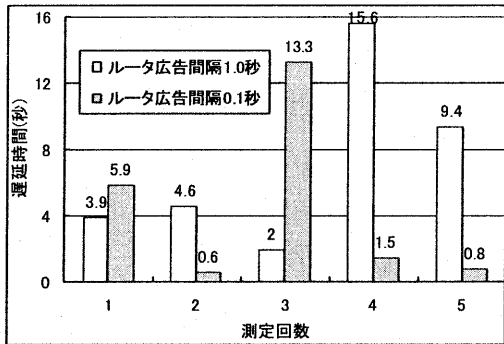


図8. 実験3の測定結果

5. 考察

図9に実験1で最短の1.7秒でハンドオフした時のMNでのパケットキャプチャ結果を示す。No. 421のパケットを受信し、No. 422のパケットを送信した後、No. 423のハンドオフ直前のルータ広告を受信している。そしてNo. 424がハンドオフ直後に受信したルータ広告である。No. 423とNo. 424の送信元アドレスを見ると、これはそれぞれAP①、AP②と接続されているルータのアドレスであるので、この間でハンドオフしたとわかる。

この場合はたまたまハンドオフ直後にルータ広告を受信したため、遅延時間が1.7秒という小さな値になったと考えられる。以下、詳細に検討する。

No. 424でハンドオフしたことを検出すると、No. 425

No.	Time	Source	Destination	Protocol	Info
421	18.249203	3ffe::83	3ffe::83	TCP	1130 > 32876 [PSH, ACK] Seq=2189527816
422	18.254924	3ffe::83	3ffe::83	TCP	32876 > 1130 [short Frame]
423	18.263316	fe80::3	ff02::1	ICMPv6	Router advertisement
424	19.758329	fe80::5	ff02::1	ICMPv6	Router advertisement
425	19.758620	3ffe::83	3ffe::83	IPv6	Mobile IPv6 Destination option
426	19.760565	3ffe::83	3ffe::83	IPv6	Mobile IPv6 Destination option
427	19.915567	3ffe::83	3ffe::83	TCP	1130 > 32876 [short Frame]
428	19.915567	3ffe::83	3ffe::83	TCP	1130 > 32876 [short Frame]
429	20.120002	3ffe::83	3ffe::83	TCP	1130 > 32876 [PSH, ACK] Seq=2189528840

図9. 実験1の4回目MNログ

のMIP6メッセージ、HAに登録してあるCoAを更新するためのBU、を送信する。No. 426がHAからの応答であり、HAは新CoAに向けてNo. 427以降のMNあてのパケットを転送してくる。No. 426とNo. 427の間に0.15秒ほど間があるのは、HAでの処理遅延だと考えられる。

No. 424のルータ広告を受信してからNo. 427のnetperfのデータを受信するまでが約0.16秒なので、ほかと比べると小さな値である遅延時間1.7秒のうちの大部分が、No. 424のルータ広告を受信するまでの時間である。すなわち、遅延時間を小さくするためには、すばやくルータ広告を受信することが必要である。これは図10からも読み取れる。

図10は実験3においてハンドオフに最長の15.6秒かかった時のMNにおけるパケットキャプチャの結果である。

No. 466のルータ広告を受けてNo. 467のBUを送信するまでに5回ルータ広告要請を行っている。ルータ広告要請はルータに対してルータ広告を要求するものであるが、5回要請を受けてからルータ広告を行ったように見える。ルータ広告要請を送信する間隔は指数バックオフされるので、この間だけで6秒程度経過している。ルータはルータ広告要請に対してすぐにルータ広告を行う必要はない[11]が、この場合毎秒1回の割合で定期的なルータ広告をしているので、無線リンクに問題があって通信できなかったとも考えられる。結局、ハンドオフ時の遅延時間を計算するもととなるNo. 459のパケットを送信してから9秒あまり経ってBU

No.	Time	Source	Destination	Protocol	Info
459	15.195719	3ffe::8	3ffe::83	TCP	32902 > 1156 [short Frame]
460	15.273185	3ffe::8	3ffe::83	TCP	32902 > 1156 [short Frame]
461	18.013181	fe80::	ff02::2	ICMPv6	Router solicitation
462	19.013174	fe80::	ff02::2	ICMPv6	Router solicitation
463	20.013181	fe80::	ff02::2	ICMPv6	Router solicitation
464	21.013166	fe80::	ff02::2	ICMPv6	Router solicitation
465	23.013179	fe80::	ff02::2	ICMPv6	Router solicitation
466	24.252342	fe80::	ff02::1	ICMPv6	Router advertisement
467	24.360112	3ffe::8	3ffe::83	IPv6	Mobile IPv6 Destination option

図10. 実験3の4回目MNログ
(ルータ広告間隔1.0秒)

No.	Time	Source	Destination	Protocol	Info
713	29.630745	3ffe::8	3ffe::8	TCP	32942 > 1053 [short Frame]
714	29.630808	3ffe::8	3ffe::8	TCP	1053 > 32942 [PSH, ACK] Seq=324
715	29.839985	3ffe::8	3ffe::8	TCP	1053 > 32942 [PSH, ACK] Seq=324
716	30.025130	fe80::	ff02::1	ICMPv6	Router advertisement
717	30.025130	3ffe::8	3ffe::8	IPv6	Mobile IPv6 Destination option
718	30.029986	3ffe::8	ff02::1	ICMPv6	Neighbor solicitation
719	30.032347	3ffe::8	3ffe::8	ICMPv6	Neighbor advertisement
720	30.032371	3ffe::8	3ffe::8	IPv6	Mobile IPv6 Destination option
721	30.034388	3ffe::8	3ffe::8	IPv6	Mobile IPv6 Destination option
722	30.125084	fe80::	ff02::1	ICMPv6	Router advertisement
723	30.225083	fe80::	ff02::1	ICMPv6	Router advertisement
724	30.257190	3ffe::8	3ffe::8	TCP	32942 > 1053 [short Frame]
725	30.257282	3ffe::8	3ffe::8	TCP	1053 > 32942 [short Frame]

図11. 実験3の2回目MNログ
(ルータ広告間隔0.1秒)

を送信している。これだけハンドオフに時間がかかってしまうと、TCP でも再送間隔が指数バックオフされるので、最終的に netperf のデータを受信するまでに 15.6 秒経過していた。

一方、図 11 は今回の実験を通してハンドオフ時の遅延時間が最短だった時(0.6 秒)の、MN におけるパケットキャプチャの結果である。

この場合では、0.1 秒ごとにルータ広告を行っていたため(No. 716, 722, 723), ハンドオフ後すぐに BU(No. 717)を送信でき、遅延時間は 0.6 秒で抑えられた(No. 714 から No. 724 まで)。

以上のように、ハンドオフ時の遅延時間はルータ広告をいつ受信するか大きく依存することが明らかになった。よってハンドオフが発生してからルータ広告を受信するまでの時間を縮めることが大切である。

そのための要素は大きく 2 つあり、1 つが無線リンク確立に要する時間、もうひとつがルータ広告の間隔である。前者は無線アクセス技術の種類に依存する問題であるから、IP 技術者にとって取り組むべきは後者の課題である。ルータ広告で貴重な無線帯域を消費しないよう、現状では効率的なルータ広告が MIPv6 を運用する上で必須であると考ええる。

6. まとめ

アプリケーションからみた Mobile IPv6 の性能、特にハンドオフ時の遅延を測定した。その結果、最短の場合には遅延時間が 1 秒未満であるが、多くの場合に 4~8 秒程度必要であることがわかった。

これらハンドオフ時の遅延時間のバラツキの要因として、移動端末がルータ広告を受信するタイミングがあげられる。

Mobile IPv6 を利用する場合のハンドオフに関しては、高速化技術が IETF においてさまざま議論されているが、それらが実用になるまで待てないことも考えられる。そういった場合に、特に遅延に敏感なストリーム系アプリケーションの安定した動作を保障するために、我々は少なくとも十数秒分のバッファをもつ必要があると考える。

今後は、さらに Mobile IPv6 に関する性能評価を継続していきたい。

文 献

- [1] IJ, <http://www.ij.ad.jp/IPv6/index.html>
- [2] NTT-C, <http://www.v6.ntt.net/ipbb/>
- [3] IETF mobileip WG, "<http://www.ietf.org/html.Charters/mobileip-charter.html>"
- [4] 3GPP, <http://www.3gpp.org/>
- [5] 3GPP2, <http://www.3gpp2.org/>
- [6] NTT-C, <http://www.hotspot.ne.jp/>

- [7] David B. Johnson, Charles E. Perkins, Jari Arkko, "Mobility Support in IPv6", <http://www.ietf.org/internet-drafts/draft-ietf-mobileip-ipv6-20.txt>
- [8] USAGI project, <http://www.linux-ipv6.org/>
- [9] KAME project, <http://www.kame.net/>
- [10] Netperf HP, <http://www.netperf.org/>
- [11] T. Narten, E. Nordmark, W. Simpson, "RFC2461 Neighbor Discovery for IP Version 6"
- [12] 竹下 隆史, 村山 公保, 荒井 透, 荻田 幸雄, "マスターリング TCP/IP 入門編 第 3 版", オーム社
- [13] Ethereal HP, <http://www.ethereal.com/>