

# 最適経路通信を行う階層ネットワークモビリティの実装と評価

島中翔 森廣勇人<sup>†</sup> 前田香織 相原玲二<sup>††</sup> 岸場清悟<sup>††</sup>

広島市立大学大学院情報科学研究科

<sup>†</sup>広島市立大学情報科学部

〒731-3194 広島市安佐南区大塚東 3-4-1

<sup>††</sup>広島大学情報メディア教育研究センター

〒739-8511 広島県東広島市鏡山 1-4-2

E-mail:hata@v6.ipc.hiroshima-cu.ac.jp

**概要** 携帯端末や無線インフラの普及により、ネットワーク層でモビリティサポートを実現するための技術開発が進んでいる。IETFで提案されているNetwork MObility Basic Support (NEMO)は、通信の際ホームエージェントを中継するため、通信遅延が増大する問題がある。また、最適経路通信が困難であり、モバイルネットワークが階層化された場合は、通信遅延がさらに増大してしまう。この問題を解決するため、本稿では、通信開始時から常に最適経路で通信を行うネットワークモビリティアーキテクチャ MAT について述べる。また、この度実装した、階層化されたモバイルネットワークについても通信を可能にする MAT デーモンの役割について詳細に述べる。最後に、NEMO および NEMO の改良アーキテクチャとの性能比較を行い、MAT の有効性を示す。

**キーワード** ネットワークモビリティ、ハンドオーバ、階層モバイルネットワーク、最適経路

## Implementation and Evaluation of Nested Network Mobility with Route Optimization

Sho HATANAKA Hayato MORIHIRO Kaori MAEDA

Reiji AIBARA<sup>††</sup> and Seigo KISHIBA<sup>††</sup>

Graduate School of Information Sciences, Hiroshima City University

<sup>†</sup>Information Sciences, Hiroshima City University

3-4-1 Ozuka-Higashi, Asaminami-ku, Hiroshima, 731-3194 Japan

<sup>††</sup>Information Media Center, Hiroshima University

1-4-2 Kagamiyama, Higashi-Hiroshima, Hiroshima, 739-8511 Japan

E-mail:hata@v6.ipc.hiroshima-cu.ac.jp

**Abstract** By deploying mobile clients and wireless network infrastructure, advanced technology to realize mobility support in a network (IP) layer is being developed. Though Network MObility Basic Support (NEMO) has been proposed by IETF, it has a serious problem. Since NEMO uses a home agent in all communications, communication delay occurs and route optimization is difficult. In the case of nested mobile networks, its delay is more serious. To solve these problems, we are developing a network mobility architecture, called as MAT (Mobile IP with Address Translation), which enables route optimization from the start of communications. In this paper, we describe the detail of implementation of a MAT daemon which realizes a interface selection function in handover and prefix and address delegation even in nested networks. Also, we show some performance evaluations compared with the other proposed architectures.

**Keyword** Network mobility, Handover, Nested mobile network, route optimization

# 1 はじめに

携帯端末やラップトップ PC などの小型通信端末が普及するとともに、WiFi[1], WiMAX[2], Bluetooth[3] や GSM[4]のような多様な通信メディアが出現している。これらの通信メディアが混在し、様々な形態のモバイルネットワークが構成されることが想定されている。例えば、WiMAX で構成されるネットワーク内を走行するバスの車内で個人が所有する PAN (Personal Area Network) が構成されるような場合も予想される。このような階層化されたネットワークに対してモビリティを提供するアーキテクチャとして、IETF で提案されている NEMO[5]や $\chi$ LIN6-NEMO[6]などが開発されている。これらのアーキテクチャでは、モバイルネットワーク内でシームレスなモビリティを実現することが要求される。しかし、NEMO では Mobile Router (MR) とホームエージェントとの双方向トンネリングが用いられ、 $\chi$ LIN6-NEMO では LIN6[7]のマッピングエージェントが通信を中継するように拡張されるため、いずれも通信経路が冗長になる問題をもつ。現在、携帯電話網、固定電話網、放送網やインターネットのように目的ごとに異なるネットワークが分断されて存在するが、次世代の基幹ネットワーク Next Generation Network (NGN) [8]による電話網の IP 化が推進されている。電話のような音声通信において遅延は通話品質を左右する非常に重要な指標であり、ネットワークモビリティを実現するアーキテクチャには最適経路通信が要求される。ここで、モバイルネットワークでの最適経路とは、通常のネットワーク層が採用する経路に一致する（またはより良い）経路を意味するものとする。

一方、筆者らは MAT (Mobility support Architecture and Technologies) を提案・実装している[9][10]。MAT はホストモビリティを MAT-HOST, ネットワークモビリティを MAT-MONET[11]で実現する。MAT ではホームエージェントのような通信を中継するサーバを用いないため、ホストモビリティだけでなくネットワークモビリティに対しても通信開始時から最適経路で通信を行う。MAT では複数インタフェースの使用を前提としており、通信していないインタフェースを用いてハンドオーバー先のネットワークにあらかじめ接続し、パケットロスが少ないシームレスハンドオーバーを実現する。ネットワークモビリティを実現するためには、アドレス割り当てや移動検知機能が必要になるが、これらの機能を MAT ではネットワークモビリティサポートプロセス (MAT デーモン) が担う。また、MAT デーモンはこれらに加え、複数インタフェースの選択機構や異なる通信メディア間をスムーズハンドオーバーする機能もサポートする。

先行研究[9][10][11]では MAT デーモンのプロトタイプが実装されたが、複数インタフェースの選択機構や階層化されたモバイルネットワークにおいて、[13]で示されたプレフィックス配布機能等が実装されていない。本研究の目的は MAT アーキテクチャに基づき、MAT デーモンのもつ全機能を実装、評価することにある。

本稿では、2 章で MAT-MONET について述べ、3 章では MAT デーモン機能の詳細について述べる。4 章で MAT,NEMO と NEMO の改良アーキテクチャの性能比較について述べ、5 章でまとめと今後の課題について述べる。

## 2 MAT-MONET

### 2.1 MAT-MONET の概要

MAT-MONET の構成例を図 1 に示す。モバイルルータ MR0 の配下となるモバイルネットワーク内にモバイルノード (MN), 従来ノード (LFN), 別のモバイルルータ MR1 が存在する。MR1 の配下のモバイルネットワークは MR0 のモバイルネットワークの入れ子となっている。モバイルルータとモバイルノードは同一機器に混載することも可能である。

MAT ではノードの識別子を表すアドレスをホームアドレス、ノードの位置情報を表すアドレスをモバイルアドレス (NEMO における Care of Address に相当) として定義している。同様に、ホームアドレスの上位 64 ビットをホームプレフィックス、モバイルアドレスの上位 64 ビットをモバイルプレフィックスと定義している。MN は通信を行う際、自身のホームアドレスをモバイルアドレスに変換する。MR は自身のアドレス変換に加えて、LFN のホームプレフィックスをモバイルプレフィックスへ変換することで LFN のネットワークモビリティをサポートする。

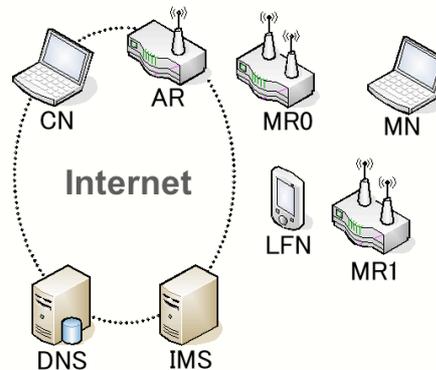


図 1 MAT-MONET の構成例

MATでは、通信相手ノード (CN) 側でも通信相手のホームアドレスをモバイルアドレスに変換しなければならないので、CNもMAT対応ノードである必要がある。このとき、CNはMNのモバイルアドレスを、IMS (IP Address Mapping Server) に問い合わせを行うことで取得する。IMSには、CNとその通信相手ノードのホームアドレスとモバイルアドレスの対応表である、IPアドレスマッピングテーブル (IMT) が管理されている。MRやMNは新しいネットワークに接続する度にMATのIMS Updateメッセージを送信し、IMSの持つIMTを更新する。IMSのアドレスは拡張を行ったドメインネームシステム (DNS) から取得できる。DNSの拡張についてはMRやMNのホームプレフィックスに属するゾーンを委譲されたDNSでだけ行われていれば良い[12]。

## 2.2 MAT-MONETのハンドオーバー

MAT-MONETのハンドオーバーの手順を図2に示す。

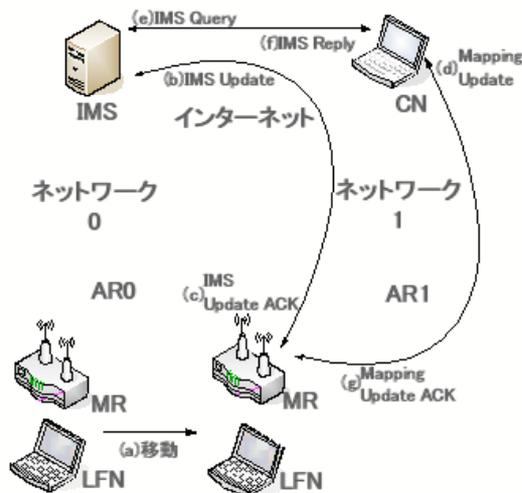


図2 ハンドオーバーの手順

MATでは複数インタフェースの使用を前提としており、図2(a)でMRとその配下のネットワークが移動し、ネットワーク1に対しても接続できるようになると、通信に使用していないインタフェースを用いてAR1に接続する。AR1からモバイルアドレスが割り当てられると、図2(b)のIMS UpdateメッセージでIMSのIMTを更新する。図2(c)のIMS Update ACKメッセージを受信することによりIMT更新の確認応答を得ると、そのIMTを基にアドレス変換を行う。また、図2(d)でCNに対して最新のIMTの参照を促すために、IPv6拡張ヘッダのDestinationオプションで定義したMapping Updateオプションを通信パケットに付与する。

Mapping Updateオプションが付与された通信パケットを受信したCNは図2(e)、(f)で最新のIMTを参照し、そのIMTに従ってアドレス変換を行い、図2(g)のMapping Update ACKを返信する。

Mapping Update ACKを受信したMRはMapping Updateオプションを通信パケットに付与するのを止める。MRがネットワーク1に接続したインタフェースを用いて通信すると判断した場合には、アドレス変換先が移動前と異なり、CNが図2(f)のIMS Replyを受信したときハンドオーバーが完了する。

## 2.3 Router Advertisement (ICMPv6)の変更

MATでは、ARやMR配下のMNやMRに対してはモバイルプレフィックスを、LFNに対してはホームプレフィックスをそれぞれ配布する。モバイルプレフィックス配布にはMATで定めたPrefix Advertisementメッセージを送信し、ホームプレフィックス配布にはRouter Advertisement (RA)メッセージを送信する。MRやMNが受信したRAメッセージがARとMRのどちらのノードから受信したメッセージなのかを判別するためにRAメッセージのAutoconfig flag内の最終ビットにMRフラグを追加する。MRフラグが追加されたRAメッセージのパケットフォーマットを図3に示す。

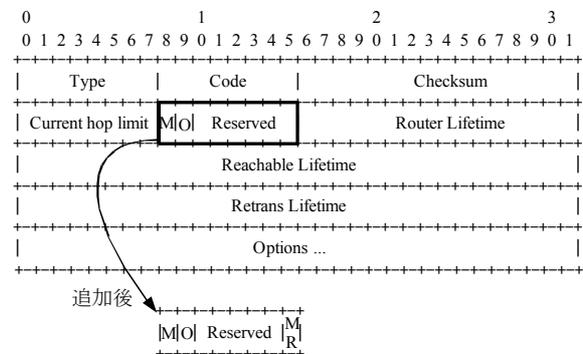


図3 MRフラグを追加したRAパケットフォーマット

MRがRAを送信する場合のみ、MRフラグの値は1となる。MRフラグが追加されたRAは、MRのみ送信するもので、MATで構成されたモバイルネットワーク内でのみ使用される。

## 3 MATデーモン

### 3.1 MATデーモンの概要

本研究で開発したMATデーモンとは、階層化されたモバイルネットワークにおいてもアドレス割り当て、移動検知やシームレスハンドオーバーを実現する機能を

持ち、MAT アーキテクチャを構成する一つのプロセスである。

MAT デーモンは MR がネットワークモビリティを実現するため以下の機能を提供する。

- (1) 配下に接続された LFN へのアドレス割当
- (2) 配下に接続された MN および MR へのモバイルアドレス割当
- (3) 上流の AR または MR からモバイルプレフィックスを取得し自ら使用 (1 または 2 プレフィックス)
- (4) 上流の AR または MR からモバイルプレフィックスを取得し配下の MR に割当
- (5) 複数インタフェースが使用可能な場合、使用するインタフェースの選択

MAT デーモンの構成を図 4 に示す。

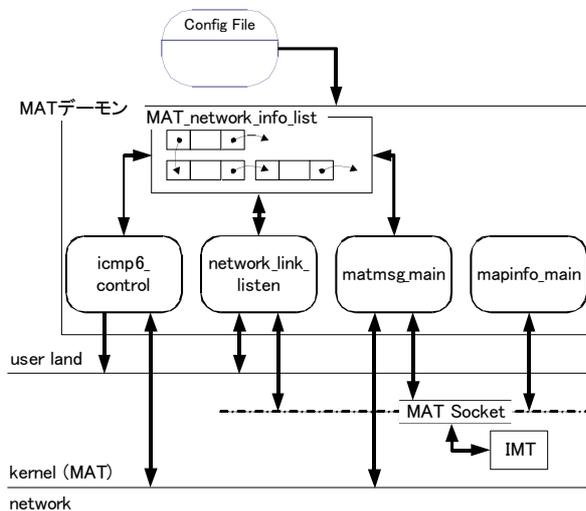


図 4 MAT デーモンの構成

MAT デーモンでは、MR フラグが追加された RA を処理するために、Linux カーネルで行われる RA の受信処理を無効にし、MAT デーモンで Router Solicitation (RS) と RA の処理を行う。また、近隣探索し、RA を受信することで取得したグローバルアドレスの重複アドレス検出も行う。これらの処理は図 4 の icmp6\_control で行われている。

network\_link\_listen では、インタフェースの電波強度などリンク状態が常に監視され、シームレスハンドオーバーのために、複数インタフェースのうちどのインタフェースを使用するかを選択処理が行われる。リンク状態は、個々の通信メディアに対応するデバイスドライバから得るのではなく、デバイスドライバの状態を反映するカーネルパラメータを取得することで実現している。これにより、様々な通信メディアを使用することができる。現時点では、有線 LAN、無線

LAN(IEEE802.11a/b/g)、PHS(6to4)での動作を確認している。

matmsg\_main では MAT で定められたプロトコルによるアドレスやプレフィックス配布を行う。プロトコルの詳細は文献[13]で述べられている。MAT では、カーネルとユーザーランド間のデータ通信のために、MAT で定めた MAT ソケットが使用され、IMT 更新処理を行う。matmsg\_main では上位 MR からのリンク状態のレポートを受信し、ハンドオーバーを行うと判断した場合には MAT ソケットを用いて IMT 更新処理を行う。

mapinfo\_main では通信相手ノードが使用する IMS のアドレスを DNS により取得し、その IMS に対して通信相手ノードのモバイルアドレスを問い合わせる処理を行う。問い合わせが成功した場合には、IMT 更新処理が MAT ソケットを通じて行われる。

icmp6\_control, network\_link\_listen, matmsg\_main では取得したアドレスやプレフィックスなど、モバイルネットワークの情報がリスト構造で管理される。

MAT アーキテクチャは Linux でプロトタイプの実装が行われている。

### 3.2 MAT デーモンによるアドレスとプレフィックス割り当て

MAT-MONET のアドレスとプレフィックス割り当てについて図 5 を用いて述べる。

AR のネットワークに接続した MR0 は、AR に対して RS(図 5 (a))を行い、AR による RA (図 5 (b))を受信することでアドレス割り当てを実現する。プレフィックス割り当ては Prefix Request (図 5 (a))を送信し、Prefix Delegation(図 5 (b))を受信することで実現する。

モバイルプレフィックスを受信した MR0 は Prefix Solicitation を送信した (図 5 (c), (e)) MN と MR1 に対して Prefix Advertisement (PA)を送信し、それぞれのアドレス割り当てを行う (図 5 (d),(f))。MN と MR1 は MR0 からの RA も受信するが、MR フラグが立っているため、その RA を無視する。LFN に対しては MR0 からの RA でホームプレフィックスの割り当てを行う。LFN のモバイルプレフィックスは MN や MR1 に割り当てたモバイルプレフィックス以外のモバイルプレフィックスを割り当てる。MR1 に対するモバイルプレフィックス割り当ては MR1 の Prefix Request (図 5 (e))、MR0 の Prefix Delegation (図 5 (f))で実現される。

図 6 にアドレスとプレフィックス割り当てのシーケンス図を示す。図 6 で MR0 は、アドレス割り当てにかかる時間を少なくするために、AR からモバイルプレフィックスを受信すると直ちに Prefix Advertisement メ

メッセージをリンクローカルマルチキャストで送信する。

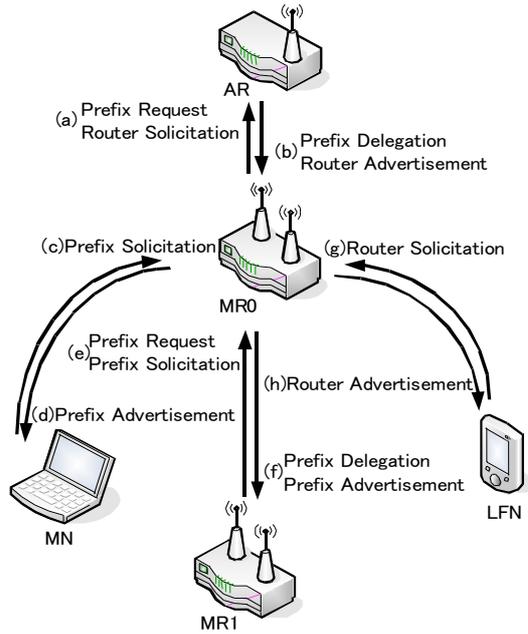


図5 アドレスとプレフィックス割り当て

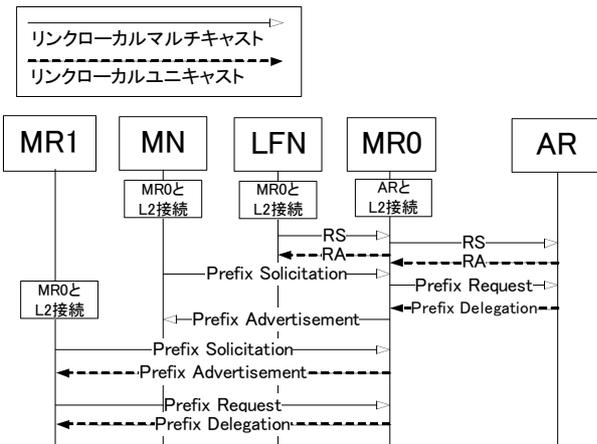


図6 アドレスとプレフィックス割り当てシーケンス

Prefix Delegation で配布するプレフィックスの割り当てブロックについては、Prefix Delegation メッセージ内の Address Block Information Option に含まれる。図7に Address Block Information Option のフォーマットを示す。

図7に示した Address Block Information Option の、各フィールドの説明を示す。

- Option Type = 9 : MATMSG\_OPT\_ABI
- Length : 8 オクテット単位のオプション長
- Lifetime : プレフィックスの有効時間
- Prefix Num : オプションに含まれるメッセージ数
- Prefix ID : プレフィックスを配布している AR の IP

アドレス

• Prefix  $n$  : プレフィックス

• Prefix Length : プレフィックス長

オプション長については、8 オクテットの整数倍になるようにオプションの最後にパディングを入れる。

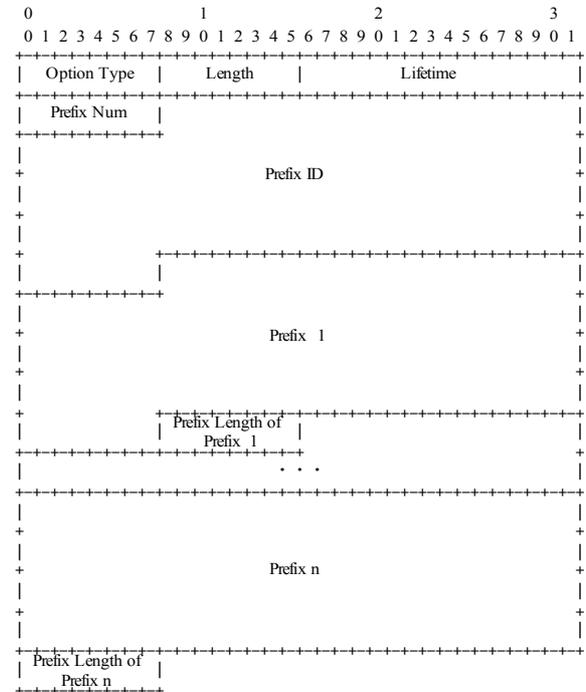


図7 Address Block Information Option フォーマット

### 3.3 通信インタフェース選択機構

#### 3.3.1 ハンドオーバーアルゴリズム

MAT では複数インタフェースを使用する通信をサポートする。通信していないインタフェースであらかじめハンドオーバー先のネットワークに接続し、ハンドオーバー前後の packet loss が発生しないようにしている。しかし、そのためには複数あるインタフェースを有効に選択することが必要である。ハンドオーバーの指標を電波強度とすると、複数あるインタフェースの電波強度が逆転する度にハンドオーバーが発生してしまうため、無駄なハンドオーバーが発生してしまう可能性がある。また、ハンドオーバー前後のネットワークの遅延差により、通信パケットの入れ替えや瞬断が発生してしまう可能性もあるため、安定した通信のためにはなるべくハンドオーバーの回数を減らすことが望ましい。

MAT デーモンに、ユーザが使用するインタフェースを選択させることと、ハンドオーバーによる無駄な通信経路の変更は避けるという目的で通信インタフェースを選択するハンドオーバーアルゴリズムを考え、実装し

た。ハンドオーバーアルゴリズムでは、図8のように各インタフェースの電波強度を3つの状態に分類する。通信インタフェースには、電波強度の高いStateのインタフェースが優先されて選択される。複数ある候補のインタフェースのStateが等しい場合には、なるべくハンドオーバーが発生しないインタフェースが選択される。

図9に、通信インタフェースが2つの場合における状態遷移図を示す。

本アルゴリズムを用いても複数のインタフェースのStateが同時かつ頻繁に変化する場合は、それに応じてハンドオーバーが頻発することは否めない。ただし、状態遷移の方向に応じて異なる閾値を設定することにより、いわゆるヒステリシス効果によりStateの頻繁な変更を抑制することができる。



図8 電波強度による状態割り当て

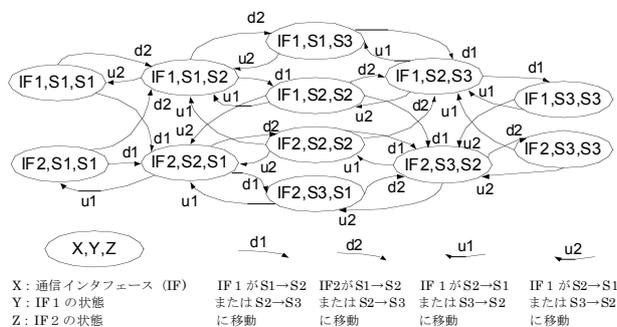


図9 状態遷移図

### 3.3.2 インタフェースの優先度について

使用できる通信メディアに対応したインタフェースが複数の場合には、ユーザが使用したいインタフェースを選択したい状況もある。そのため、ユーザが使用したいインタフェースの優先度を図4のConfig Fileで指定できるようにした。例えば、複数のインタフェースがState1にあるときには、ユーザが指定した優先度の高いインタフェースが通信インタフェースとして選択される。

## 4 性能評価

### 4.1 遅延比較

文献[14]では、NEMOを改良したアーキテクチャ(以降、Improved NEMOと呼ぶ)が提案されており、シミュレーションによりNEMOとの通信遅延の比較や、ハンドオーバーの際のパケットロスについてまとめられている。このときのシミュレーション環境を図10に示す。Improved NEMOは、最上位MRが下位のモバイルノードの双方向トンネリングの確立を行い、モバイルネットワークのネスト数に関わらず通信の際の双方向トンネリング数が1または2でよいという特徴を持つ。

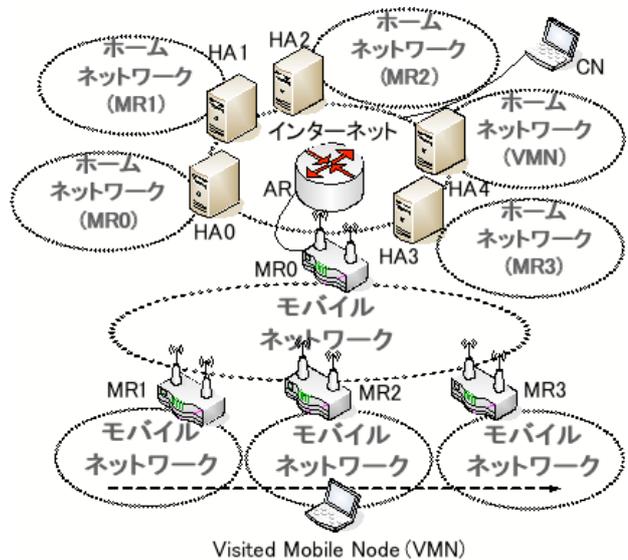


図10 シミュレーション環境

文献[14]のシミュレーションでは、インターネットの遅延を100ms(一定)としており、CNがVMNに対して500byteのデータを30ms間隔で送信した場合の遅延の結果がまとめられている。ハンドオーバーは、開始から16.26秒経過後と43.80秒経過後にそれぞれMR1→MR2とMR2→MR3へ発生している。

これと同一条件でMATアーキテクチャによる比較をするために構築した環境を図11に、使用した機器の仕様を表1に示す。この環境において、CNからiperf[15]を用いてVMNに対してUDPのペイロードが500byteのデータを送信する。送信間隔は約30msになるように帯域を133Kbpsと指定しデータを送信した。また、インターネットの遅延が100msと想定されているためCNとARの間は100msの遅延が発生するものとし、遅延の測定を行った。ハンドオーバー開始となるIMS Updateメッセージは通信開始から15.60秒経過後と42.90秒経過後にそれぞれMR1→MR2とMR2→MR3へハンドオーバーするように送信した。

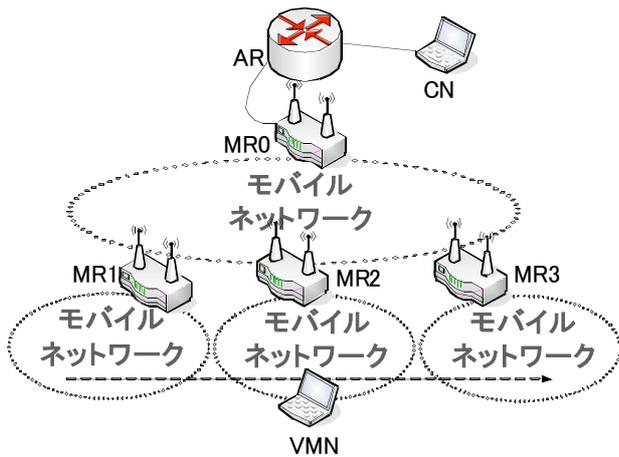


図 1 1 MAT 測定環境

表 1 機器仕様

	CPU	Memory	OS
AR	IntelPentium4 2.6GHz	512MB	Debian GNU/Linux 2.6.16.5
MR0	IntelPentiumM 1.7GHz	2GB	Debian GNU/Linux 2.6.16.5
MR1	IntelPentiumM 1.6GHz	512GB	Debian GNU/Linux 2.6.16.5
MR2	IntelPentium4 2.6GHz	512MB	Debian GNU/Linux 2.6.16.5
MR3	Genuine Intel U1400 1.2GHz	512MB	Debian GNU/Linux 2.6.16.5
MN	IntelPentiumM 1.2GHz	1GB	Debian GNU/Linux 2.6.16.5

文献[14]のシミュレーション結果と測定した比較結果を図 1 2 に示す. 図 1 2 の MAT の結果は 0.5 秒間隔で遅延をプロットしたものである. NEMO, Improved NEMO については文献[14]の Figure 5 を基にプロットし直した結果を載せている.

NEMO の通信経路例は図 1 0 において, CN→HA4→HA3→HA0→AR→MR0→MR3→VMN となり, Improved NEMO の通信経路例は CN→HA4→AR→MR0→MR3→VMN となるため, 図 1 2 の遅延は大部分がシミュレーションで設定した, インターネットの遅延 100ms が占めるものとなる.

NEMO と Improved NEMO ではハンドオーバー前後で

パケットの遅延が増大している. これはハンドオーバー先のネットワークに接続する処理と Binding Update の遅延時間によって発生したものである.

MAT の通信経路例は図 1 1 において, CN→AR→MR0→MR3→VMN なので, 通信遅延はインターネットの遅延 100ms 程度となった.

MAT では別インタフェースでハンドオーバー先のネットワークにあらかじめ接続し, CN との通信以外の新たなパケットで IMT の更新を図る. そのため, NEMO や Improved NEMO で見られたハンドオーバー前後の大きな遅延のゆれはみられず, シームレスなハンドオーバーが実現できている. MAT の測定は 20 回程度行ったが, どの結果についても同様の遅延結果であった.

MAT の通信経路例は図 1 1 において, CN→AR→MR0→MR3→VMN なので, 通信遅延はインターネットの遅延 100ms 程度となった.

MAT では別インタフェースでハンドオーバー先のネットワークにあらかじめ接続し, CN との通信以外の新たなパケットで IMT の更新を図る. そのため, NEMO や Improved NEMO で見られたハンドオーバー前後の大きな遅延のゆれはみられず, シームレスなハンドオーバーが実現できている. MAT の測定は 20 回程度行ったが, どの結果についても同様の遅延結果であった.

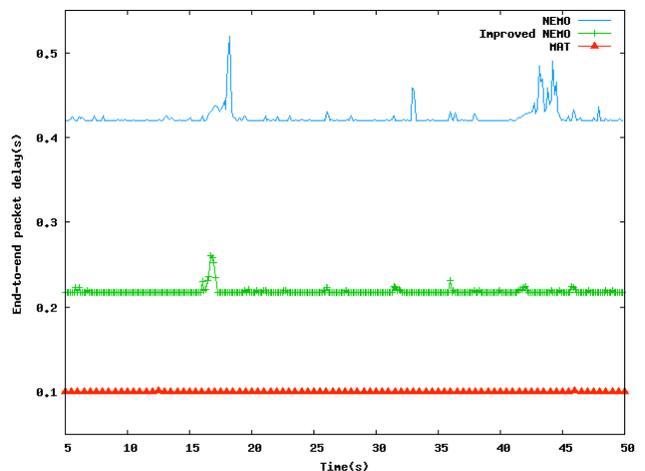


図 1 2 遅延比較結果

#### 4.2 ハンドオーバー前後のパケットロスの比較

NEMO と Improved NEMO のハンドオーバー前後における通信経過時間とパケットシーケンスナンバーの関係を図 1 3, MAT の計測結果を図 1 4 に示す. 図 1 3 については文献[14]の Figure 6 を基にプロットし直した結果を載せている. NEMO と Improved NEMO についてはともにハンドオーバー先のネットワークに接続し, 通信に使用できるアドレスを取得するまでの時間

Movement Detection Delay (MDD) と Binding Update や Binding ACK の処理に要する時間 Registration Delay (RD) の和をハンドオーバー遅延としている。VMN が MR3 の配下にいた場合、NEMO の Binding Update の経路は VMN→MR3→MR0→AR→HA0→HA3→HA4 となり、シミュレーションパラメータであるインターネットを経由する際の遅延 100ms を考えると、RD に要する時間は Binding Update と Binding ACK の通信経路から 600ms 程度かかり、図 1 3 では MDD と RD の和であるハンドオーバー遅延は 890ms となる。

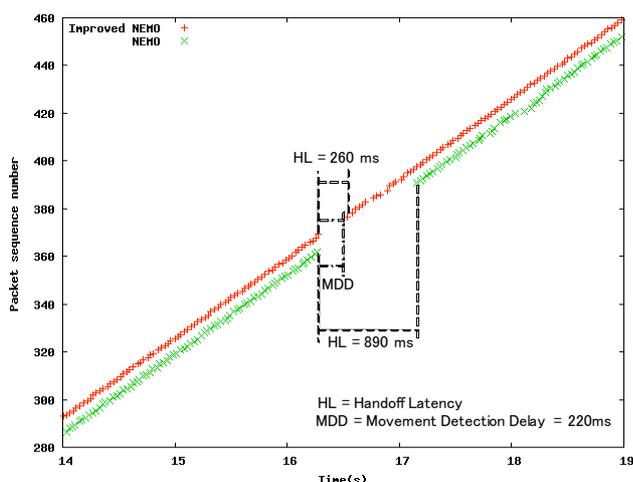


図 1 3 NEMO と Improved NEMO のハンドオーバー前後の packets シーケンス

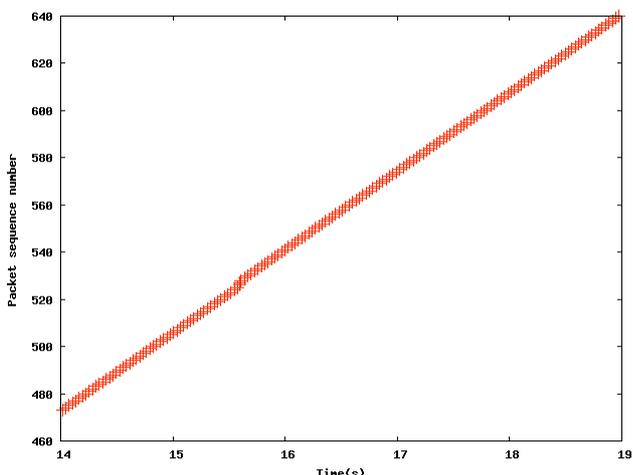


図 1 4 MAT のハンドオーバー前後の packets シーケンス (複数インタフェース利用の場合)

Improved NEMO では、図 1 0 のようなネストされたモバイルネットワーク間を移動する場合 Binding Update をホームエージェントが受信する必要がない。

そのため VMN が MR3 の配下にいた場合、Binding Update に要する時間は VMN→MR3→MR0 の経路というモバイルネットワーク内での通信遅延の結果となるため、RD が小さくなる。図 1 3 を見ると MDD と RD の和であるハンドオーバー遅延は 260ms となり、NEMO と比較してハンドオーバー遅延を非常に小さくできた結果となっている。NEMO と Improved NEMO とともにハンドオーバーにかかる時間だけパケットロスしていることがわかる。

図 1 4 では、ハンドオーバー開始となる IMS Update メッセージを送信した 15.60 秒経過後においてもパケットロスは発生していない。MAT では別インタフェースを使用してあらかじめハンドオーバー先のネットワークに接続するため、ハンドオーバー前後でパケットロスが発生しない。ただし、ハンドオーバーにかかる時間分、MN がハンドオーバー前後のネットワークの両方に接続しているときに限る。MAT では RD として、MN による IMS Update, IMS Update ACK や Mapping Update が、また CN による IMS Query, IMS Reply がそれぞれインターネットを経由するため今回の設定では 500ms 程度かかる。そのため、ハンドオーバー開始となる IMS Update メッセージ送信開始から 500ms 程度はハンドオーバーする前のネットワークと接続性をもっていない場合はパケットロスが生じる。

## 5 おわりに

NEMO と Improved NEMO についてのシミュレーション結果と我々が提案している、移動透過アーキテクチャ MAT のプロトタイプ実装の性能比較した。結果、MAT では階層化されたモバイルネットワークにおいても最適経路の通信が可能で、パケットロスのないシームレスなハンドオーバーが可能であることを示した。

MAT デーモンに実装した、様々な通信メディアを使用可能にする構想や、通信インタフェース選択機構については、シームレスなハンドオーバーを実現するには必要であると考えられる。今後はこれらの機能の拡張を行いたい。

## 謝辞

MAT デーモンの開発に関して御助言を頂きました、株式会社ディアイティの藤田貴大氏、広島大学の西村浩二准教授、田島浩一助教、正岡元氏、広島市立大学の河野英太郎助教、ネットワンシステムズ株式会社の赤座正樹氏、杉本康則氏、岸田崇志氏に深く感謝致します。本研究の一部は広島市立大学平成 19 年度特定研究費 (7203) と日本学術振興会科学研究費補助金基盤

研究(B) (課題番号 19300019) の支援を受けて実施しています。ここに記して感謝の意を示します。

## 参考文献

[1] Wifi: <http://www.wi-fi.org/>.

[2] WiMAX: <http://www.wimaxforum.org/home/>.

[3] Bluetooth:  
<http://www.bluetooth.com/Bluetooth/Default.html/>.

[4] GSM: <http://www.gsmworld.com/index.shtml/>.

[5] V.Devarapalli, R. Wakikawa, A.Petrescu, and P.Thubert, "Network Mobility (NEMO) Basic Support Protocol," IETF, RFC 3963, Jan. 2005.

[6] A. Banno, T. Oiwa, and F. Teraoka, "γ LIN6-NEMO: A network mobility protocol based on LIN6," IEICE Trans. Commun., Vol.E89-B, No.4, pp1070-1079, Apr. 2006.

[7] M. Ishiyama, M. Kunishi, K. Uehara, H. Esaki, and F. Teraoka, "LINA: A new approach to mobility support in wide area networks," IEICE Trans. Commun., Vol.E84-B, No8, pp.2076-2086, Aug. 2001.

[8] 江川尚志, "NGN(NEXT Generation Network)概説", JPNIC ニュースレター, No.31, Sep. 2005.  
<http://www.nic.ad.jp/ja/newsletter/No31/020.html/>.

[9] 相原, 藤田, 前田, 野村, "アドレス変換方式による移動透過性インターネットアーキテクチャ," 情報処理学会論文誌, Vol.43, No.12, pp.3889-3897, Dec. 2002.

[10] 相原 他, "常に最適経路で通信を行う移動透過アーキテクチャ MAT の性能評価," インターネットカンファレンス 2006 論文集(日本ソフトウェア科学会), pp.13-20, Oct. 2006.

[11] 藤田 他, "MAT によるモバイルネットワークの実現," 分散, 協調とモバイル (DICOMO) 2003 シンポジウム論文集, pp.105-108, Jun. 2003.

[12] A. Gustafsson, "Handling of Unknown DNS Resource Record (RR) Types," IETF, RFC3597, Sep. 2003.

[13] 梶原 他, "MAT-MONET におけるアドレス割当と移動通知プロトコル," 信学技報 MoMuC2005-72, pp.81-86, Nov. 2005.

[14] M.Sabeur, B.Jouaber and D.Zeglache "Seamless Handoff Solution For Nested Mobile Networks", JOURNAL OF NETWORKS, Vol.1, No.4, pp21-28, Aug. 2006.

[15] Iperf: <http://dast.nlanr.net/Projecs/Iperf/>.