

携帯通信網における Locator/ID分離に基づくムービングセルの実現

落合 孝壮^{†1,a)} 松枝 耕平^{†1,b)} 金子 晋丈^{†2,c)} 寺岡 文男^{†2,d)}

概要: LTE/LTE-Advanced は、端末ごとに移動を管理しておりムービングセルをサポートしていない。ムービングセルは列車などの移動体に設置される小型基地局である。ムービングセルは移動体内の端末を収容し、端末に代わって制御メッセージのやり取り（シグナリング）を代表して行うことで、ハンドオーバーの際に端末毎のシグナリングが大量発生することを防ぐ。また、LTE/LTE-Advanced では、データ転送にトンネリングを利用するため、トンネリングオーバーヘッドが発生する。そこで本稿では、トンネリングを利用しないムービングセルをサポート可能なネットワーク制御型ネットワークモビリティプロトコル MocLis を提案する。本提案では、IPv6 アドレスを Locator（位置情報）と ID（識別子）に分割し、これらの関係をマッピングとして保持する。パケットは、マッピングに対応して Locator を変換することで転送される。提案手法ではトンネリングオーバーヘッドは発生せず、入れ子のムービングセルもサポートする。提案手法を Linux のカーネルに実装し、ネットワークの各ノードを仮想マシン上にデーモンとして実装することで、端末・中継基地局の接続およびハンドオーバーの処理時間を測定した。測定の結果、音声・ライブストリーミング動画配信の許容遅延時間内で動作することが示された。

キーワード: ネットワークモビリティ (NEMO), ムービングセル, ID/Locator 分離

Realization of Moving Cells based on Locator/ID Separation in Mobile Network

Abstract: LTE/LTE-Advanced supports terminal mobility but does not support network mobility, i.e., moving cells. Moving cells are small cells located in the mobile entities, for instance, trains and busses, accomodating all terminals in them. They play an important role for preventing explosion of signalling messages in handover. They exchange signalling messages on behalf of all terminals under them. Also, LTE/LTE-Advanced uses tunneling for data transmission, which causes tunneling overheads. This paper proposes, MocLis, the network-based network mobility protocol without tunneling. In MocLis, the IPv6 address is split into two fields: locator and identifier, and the relation of two fields is retained as mapping information. Packets are transferred by replacing the locator, according to mapping information. The proposed protocol generates no tunneling overheads and also support nested moving cells. We implemented in Linux kernel and MocLis daemon on each node in the network. We evaluate the attach procedure time of terminals and relay base stations and handover time. The results show that MocLis can operate under allowable time for streaming applications such as voice and movie.

Keywords: Network Mobility (NEMO), Moving cell, ID/Locator Separation

¹ 情報処理学会

IPJS, Chiyoda, Tokyo 101-0062, Japan

^{†1} 現在, 慶應義塾大学大学院 理工学研究科 〒 223-8522 神奈川県横浜市港北区日吉 3-14-1

Presently with Graduate School of Science and Technology, Keio University 3-14-1 Hiyoshi, Kohoku-ku, Yokohama, Kanagawa 223-8522 Japan

^{†2} 現在, 慶應義塾大学 理工学部 〒 223-8522 神奈川県横浜市港北

区日吉 3-14-1

Presently with Faculty of Science and Technology, Keio University 3-14-1 Hiyoshi, Kohoku-ku, Yokohama, Kanagawa 223-8522 Japan

a) apollo@inl.ics.keio.ac.jp

b) baron@inl.ics.keio.ac.jp

c) kaneko@dmc.keio.ac.jp

d) tera@ics.keio.ac.jp

1. はじめに

近年、移動端末の性能向上と通信サービスの高度化、それに伴うユーザの QoE (Quality of Experience) 向上の要求に合わせ、公衆移動体通信システムは高速・大容量へと進化を遂げてきた。これらの変化に加え、あらゆるモノがインターネットに接続する IoT (Internet of Things) が提唱されている。そのため、今後の公衆移動体通信システムにはユーザ端末の他に多数の機械が接続され、ネットワークに接続する端末の数は急増することが想定される。

現在普及している LTE/LTE-Advanced は、ムービングセルをサポートしていない。ムービングセルは、バスや電車等の移動体に中継基地局であるリレーノード (Relay Node : RN) が乗客の端末を収容し、RN と端末が一体として移動するものである。ユーザが持ち運ぶモバイルルータも RN の一種である。RN は基地局からの電波を受信し、車内に送信することで車内の複数ユーザを収容する。したがって、従来は電波の届きにくかった車内の電波状況が改善され、スループットの向上が可能である。ムービングセルの移動は複数の端末の集合移動であり、この移動はネットワークモビリティに分類される。しかし、LTE/LTE-Advanced では、ネットワーク制御型ターミナルモビリティプロトコルである Proxy Mobile IPv6 (PMIPv6)[1]を採用しているため、各端末が移動制御のために LTE/LTE-Advanced のコアネットワークとシグナリングする。このため、例えば電車が基地局間を移動したとき、電車内のユーザ端末全てが一斉に移動するため、シグナリングが大量発生し、シグナリングオーバーヘッドとなる。また、LTE/LTE-Advanced では、携帯通信網に接続する端末がデータ転送を行う際にトンネリング処理が行われる。この処理の際に、P-GW (Packet Data Network Gateway), S-GW (Serving Gateway) といった携帯通信網を構成する各ノードで制御メッセージのやりとり (シグナリング) が発生し、シグナリングオーバーヘッドが発生する。トンネリング確立後の通信ではトンネリングオーバーヘッドが発生する。携帯通信網に接続される端末が増加するほど、コアネットワークを占めるシグナリングとトンネリングによるオーバーヘッドの占める割合は大きくなる。したがって、接続端末数が急増することが予想される今後の移動通信システムでは、これらのオーバーヘッドはサービス品質に悪影響を及ぼすことが懸念される。

そこで本稿では、トンネリング処理を利用せず、かつムービングセルをサポート可能なネットワーク制御型モビリティサポートプロトコルである MocLis (Moving Cell by Locator/ID Separation) を提案する。Locator/ID 分離に基づき、IPv6 アドレスを位置と識別子の 2 つの情報に分離して利用することで、シグナリングを単純化し、データ転送におけるヘッダオーバーヘッドをなくすことを目標と

する。MocLis を Linux のカーネルに実装することで基本性能を測定した。

2. 関連研究

2.1 設計目標

本稿では、トンネリングオーバーヘッドをなくし、ムービングセルをサポートするため、次の設計目標を考慮したネットワーク制御型モビリティサポートプロトコルを提案する。(1) トンネリングを利用しない、(2) シグナリングの単純化、(3) 入れ子ムービングセルのサポート、(4) UE や RN の経路が最適化、(5) ネットワークに接続した端末数に対してコアネットワーク全体の経路情報は非依存。

既存のネットワーク制御型モビリティサポートプロトコルは、全てトンネリングを利用している。また、その他の設計目標についても全ての点を満足するものはない。

2.2 NEMO-enabled PMIPv6 (N-PMIPv6)

N-PMIPv6[2] は、PMIPv6 を拡張した NEMO をサポートするプロトコルである。N-PMIPv6 は、LMA (Local Mobility Anchor), MAG (Mobility Access Gateway), MN (Mobile Node), mMAG (moving MAG) で構成される。mMAG は PMIPv6 ドメイン内を移動する MAG である。mMAG は MAG に接続し、LMA から PMIPv6 ドメインに属する IPv6 プレフィックスを提供されることで、PMIPv6 ドメインを拡張する。MAG は通常の PMIPv6 と同様に動作する。mMAG に MN が接続すると、mMAG が MN のシグナリングを代替する。LMA が保持する Binding Cache Entry (BCE) は、mMAG の ID, MAG の ID, mMAG に割り当てたプレフィックスを管理する。MAG が保持する Binding Update List Entry (BULE) は、接続する mMAG の ID を管理する。mMAG に接続する MN が送信するパケットは、LMA-mMAG 間と LMA-MAG 間で多重に確立したトンネルで転送される。N-PMIPv6 はトンネリングを利用するためトンネリングオーバーヘッドが発生する。また、mMAG が入れ子する場合、mMAG に予めプレフィックスを割り当てておく必要がある。mMAG がユーザが所持するモバイル LTE ルータである場合、予めプレフィックスを割り当てて管理するのは煩雑であり、アドレス空間の利用効率が低下する。

2.3 Network Mobility Support in PMIPv6 Network (N-NEMO)

N-NEMO[3] は、N-PMIPv6 の多重トンネルによるシグナリング・ヘッダオーバーヘッドを低減するプロトコルである。N-PMIPv6 は、LMA-MAG 間と MAG-MR (Mobile Router) 間のそれぞれにトンネルを確立する。MR が入れ子になる場合、上流の MR は下流の MR に対して MAG として動作する。N-NEMO はトンネリングを利用するた

め、ヘッダオーバーヘッドが発生する。また、LMA はモバイルネットワーク内の全 MMN, MR の経路情報を保持するが、経路情報の集約ができない可能性が高い。

2.4 Proxy Mobile Network Mobility (PNEMO)

PNEMO[4] は、PMIPv6 を拡張し NEMO のサポートするプロトコルである。PNEMO における MN は移動端末または移動ルータであり、2 つに分類される。1 つ目の Local Fixed Node (LFN) は LFN は常に同じモバイルネットワークに接続されている MN である。LFN は MIPv6 をサポートしていないため接続点を変更することができない。2 つ目の Visited Mobile Node (VMN) は異なるモバイルネットワークに移動できる MNN である。VMN は MIPv6 をサポートしているため接続点を変更することができる。PNEMO は、入れ子のモバイルネットワークでも、LMA と MAG 間のトンネルのみを生成することでトンネルの多重化を防ぐ。このため、ヘッダのオーバーヘッドは増大しない。トンネルを多重化せずにモバイルネットワークの移動をサポートするために、LMA と MAG では MN が接続する MR の情報を、MR では接続される MN の情報をそれぞれテーブルで管理する。これらのテーブルで保持する情報を基に、LMA は通信相手からのパケットを MN まで転送する。PNEMO は、トンネリングを利用するためヘッダオーバーヘッドが発生する。また、MIPv6 をサポートしていないノードはモバイルネットワーク間を移動不可である。

2.5 PMIPv6-Based NEMO (P-NEMO)

P-NEMO[5] は、ITS (Intelligent Transport Systems: 高度道路交通システム) 通信のために PMIPv6 を拡張し NEMO のサポートするプロトコルである。P-NEMO は、LMA の BCE と MAG の BULE を拡張して MR に割り当てたプレフィクス (HNP) を管理する。MR が MAG に接続すると、MR が MAG を中継して LMA に Router Solicitation (RS) を送信し、LMA は MAG を中継して MR に Router Advertisement (RA) を返送する。MR は、受信した RA から HNP と MNP を取得し、MR の下流に MNP を送信する P-NEMO はトンネリング処理を行うため、ヘッダオーバーヘッドが発生する。また、IP アドレスを取得済みの MN が MR に接続する場合を考慮していない。

2.6 Prefix Delegation Support for Proxy Mobile IPv6 (PMIPv6-NEMO)

PMIPv6-NEMO[6] は、PMIPv6 ドメイン内の MR が DHCPv6[7] を使用して IP プレフィクスを取得することで NEMO をサポートする PMIPv6 の拡張プロトコルである。MR に接続する VMN は LFN のみである。MR が MAG に接続すると、LMA-MAG 間にトンネルを確立し LMA から MAG を中継してプレフィクスを MR に委譲する。LMA

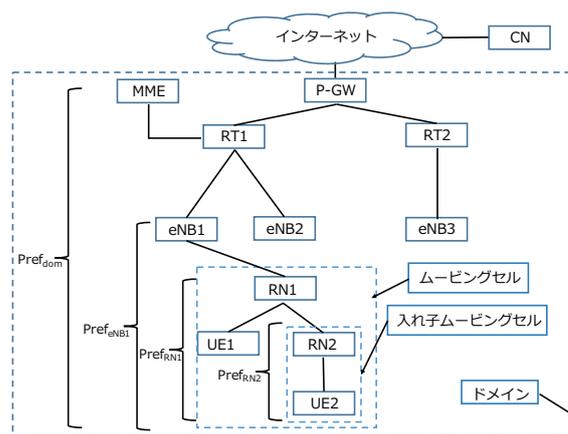


図 1 MocLis システム構成

がプレフィクスを委譲する委譲ルータ、MR がプレフィクスを委譲される要求ルータである。MR は委譲されたプレフィクス空間の中から LFN に /64 の IP プレフィクスを配布する。PMIPv6-NEMO はトンネリングを利用しているためヘッダオーバーヘッドが発生する。また、MIPv6 を実装していないノードはモバイルネットワーク間を移動できない。さらに、IP アドレス取得済みの MN が MR に接続する場合モバイルネットワークの入れ子を考慮していない。

3. MocLis の設計

3.1 概要

MocLis は、トンネリングを利用せずムービングセルをサポートする通信プロトコルである。本提案プロトコルが動作する範囲をドメインと定義する。ドメインの例としては携帯電話網のような IPv6 が標準動作するネットワークを想定している。

MocLis のシステム構成は、LTE/LTE-Advanced と類似したものを想定している (図 1)。ドメインは P-GW を介してインターネットに接続する。RT は通常のルータである。eNB は基地局であり、有線網と無線回線の中継するルータとして機能する。UE はユーザが利用する端末であり、アプリケーションが動作する。RN は無線回線を利用して eNB に接続する移動ルータである。UE は無線回線を通じて eNB または RN に接続する。RN は UE としても動作するため、RN 上でアプリケーションが動作することもある。RN とその下流に接続された範囲をムービングセルとする。RN は他の RN の下流に接続することもある。Mobility Management Entity (MME) は、ドメイン内の UE や RN の接続場所を管理する。CN はドメイン外に位置する通信相手である。MocLis ではトンネリングを利用しないため、LTE/LTE-Advanced における P-GW-S-GW 間のトンネリングは不要である。したがって、本提案プロトコルのネットワークに S-GW は省略可能であり、ネットワーク構成が簡素化される。

想定される接続、ハンドオーバーのパターンは、以下の通

りである。(1) RN1がeNB1に接続,(2) UE1がRN1に接続,(3) RN2がRN1に接続,(4) UE2がRN2に接続,(5) RN1のeNB1からeNB2への移動。

MocLisでは、ID/Locator分離に基づきIPv6アドレスを上位64ビットのプレフィクス、下位64ビットのNode-IDの組み合わせで表現する。プレフィクスはネットワークの位置情報、Node-IDはドメイン内でユニークな端末の識別子である。MocLisでは、ドメイン全体に共通のIPv6プレフィクス(共通プレフィクス)が、eNBとRNの無線インタフェースにはそれぞれ固有のIPv6プレフィクスがそれぞれ割り当てられていると仮定する。接続するRN、UEは接続先のノードが持つプレフィクスと自身のNode-IDを連結してIPv6アドレスを生成する。ムービングセル内のRNやUEへのパケットの転送は、P-GWやRNにおいて終点IPv6アドレスのプレフィクスを終点ノードのNode-IDに対応したものに交換することによって実現する。この結果、パケットはプレフィクスに対応した経路に従ってルーティングされる。トンネリングを利用しないため、ヘッダオーバーヘッドは発生しない。MMEはドメイン内に接続するUE、RNのNode-IDとプレフィクスの関係(マッピング)を保持し、RNは下流に接続したUE、RNのマッピングを保持する。ドメインに端末が接続されるたびにMMEは新たにマッピングを保持するため、端末数が増加するほどマッピング情報は増加するが、トンネリングを利用しないためコアネットワークは端末ごとの経路情報を保持する必要がない。したがって、端末数の増加によってコアネットワーク内の経路情報が増加することを防ぐ。また、パケットはプレフィクスに対応するIPv6経路に従ってルーティングされるため、最適経路での転送が可能である。

次節以降にてUE、RNの接続、ハンドオーバー手順とパケットの転送手順を示す。ドメインに割り当てられたプレフィクスを $Pref_{dom}$ 、eNBやRNのIPv6プレフィクスとNode-IDをそれぞれ $Pref_{eNB}$ 、 $Pref_{RN}$ 、 ID_{eNB} 、 ID_{RN} と表記する。

3.2 接続手順

本節では、RNやUEがIPアドレスを生成するまでのシグナリングの手順を示す。接続手順後にRNやUEのアプリケーションはインターネット上の通信相手と接続可能になる。

ドメイン内の有線ネットワーク構成は固定的であるので、MMEはドメイン内のeNBのNode-IDと無線インタフェースのIPv6プレフィクスの対応をあらかじめMapping Tableに保存している。保存しているMapping Tableの状態は表3.2の通りとなる。表3.2の1行目は、eNB1の無線インタフェースのIPv6プレフィクスが $Pref_{eNB1}$ であることを示す。2、3行目も同様である。

$ID_{eNB1} \rightarrow Pref_{eNB1}$
$ID_{eNB2} \rightarrow Pref_{eNB2}$
$ID_{eNB3} \rightarrow Pref_{eNB3}$

図2 初期状態のMMEが保持するMapping Table

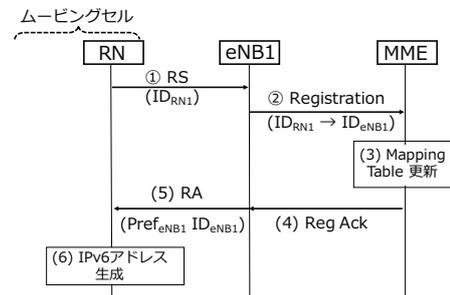


図3 RN1がeNB1に接続する手順

$ID_{eNB1} \rightarrow Pref_{eNB1}$
$ID_{eNB2} \rightarrow Pref_{eNB2}$
$ID_{eNB3} \rightarrow Pref_{eNB3}$
$ID_{RN1} \rightarrow ID_{eNB1}$

図4 RN1のeNB1への接続後にMMEが保持するMapping Table

3.2.1 RNのeNBへの接続

例として図1においてRN1がeNB1に接続する場合を考える。無線回線確立後のメッセージ交換の手順を図3に示す。

- (1) RS (Router Solicitation) メッセージ
RN1はeNB1にRouter Solicitation (RS) メッセージを送信する。このメッセージにはRN1のNode-IDである ID_{RN1} を含む。
- (2) Registration メッセージ
eNB1はRSを受信すると、RegistrationメッセージをMMEに送信する。このメッセージにはRN1とeNB1のNode-IDである ID_{RN1} と ID_{eNB1} を含む。
- (3) MMEのMapping Table更新
MMEはRegistrationメッセージを受信すると図3.2.1に示すように自ノードが保持するMapping Tableを更新する。表3.2.1の4行目は、RN1がeNB1に接続していることを示す。
- (4) Registration Ack メッセージ
MMEはMapping Tableを更新した後、Registration AckメッセージをeNB1に返送する。
- (5) Router Advertisement (RA) メッセージ
eNB1はRegistration Ackメッセージを受信すると、Router Advertisement (RA)メッセージをRN1に送信する。このメッセージにはeNB1の無線インタフェースのIPv6プレフィクスである $Pref_{eNB1}$ を含む。

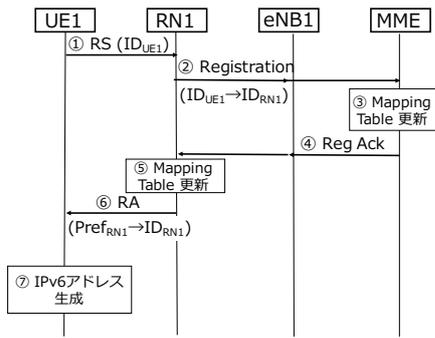


図 5 UE1 が RN1 に接続する手順

$ID_{eNB1} \rightarrow Pref_{eNB1}$
$ID_{eNB2} \rightarrow Pref_{eNB2}$
$ID_{eNB3} \rightarrow Pref_{eNB3}$
$ID_{RN1} \rightarrow ID_{eNB1}$
$ID_{UE1} \rightarrow ID_{RN1}$

図 6 UE1 の RN1 への接続後に MME が保持する Mapping Table

$ID_{UE1} \rightarrow ID_{RN1}$

図 7 UE1 の RN1 への接続後に RN1 が保持する Mapping Table

(6) IPv6 アドレス生成

RN1 は RA メッセージを受信すると、eNB1 の無線インタフェースの IPv6 プレフィクスである $Pref_{eNB1}$ と自身の Node-ID である ID_{RN1} から IPv6 アドレスを生成する。

以上の手順により、RN1 はドメインに接続される。ドメイン内で変化する情報は MME の Mapping Table に RN1 が追加されるのみである。ドメイン内の各ノードで個別に経路情報を追加する必要はなく、コアネットワーク内の経路情報は接続する端末に依存して増加しない。

3.2.2 UE1 の RN1 への接続

前述の RN1 の接続後に、さらに UE1 が eNB1 に接続する場合を考える。UE1 の無線回線確立後のメッセージ交換の手順を図 5 に示す。まず最初に、UE1 は RN1 に RS メッセージを送信する。RS メッセージは RN1 の Node-ID を含む。RS メッセージを受信した RN1 は MME と Registration メッセージと Registration Ack メッセージを交換する。その結果、MME と RN1 はそれぞれの Mapping Table に UE1 の情報を追加する (表 3.2.2, 3.2.2)。Mapping Table 更新後、RN1 は UE1 に RA メッセージを送信し、受信した UE1 は RN1 のプレフィクスと自身の Node-ID から IPv6 アドレスを生成する。UE1 の接続によって MME と RN1 のマッピング情報は増加するが、ドメイン内の各ノードは UE1 に対して個別の経路情報を保持する必要がない。したがって、コアネットワーク内の経路情報はムービングセル内に接続する UE の台数に依存しない。

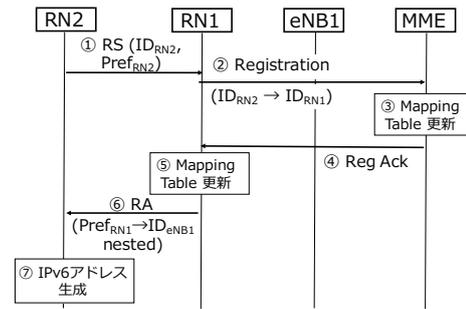


図 8 RN2 の RN1 への接続手順

$ID_{eNB1} \rightarrow Pref_{eNB1}$
$ID_{eNB2} \rightarrow Pref_{eNB2}$
$ID_{eNB3} \rightarrow Pref_{eNB3}$
$ID_{RN1} \rightarrow ID_{eNB1}$
$ID_{UE1} \rightarrow ID_{RN1}$
$ID_{RN2} \rightarrow ID_{RN1}$

図 9 RN2 の RN1 への接続後に MME が保持する Mapping Table

$ID_{UE1} \rightarrow ID_{RN1}$
$ID_{RN2} \rightarrow ID_{RN1}$
$ID_{RN2} \rightarrow Pref_{RN2}$

図 10 RN2 の RN1 への接続後に RN1 が保持する Mapping Table

3.2.3 RN2 の RN1 への接続

前述の UE1 の接続後に、さらに RN2 が RN1 に接続する場合を考える。RN2 の無線回線確立後のメッセージ交換の手順を図 8 に示す。RN2 は RN1 に、RS メッセージを送信する。RS メッセージは、RN2 の Node-ID と RN2 が保有する IPv6 プレフィクスを含む。RS メッセージを受け取った RN1 は、MME と Registration メッセージと Registration Ack メッセージを交換する。その結果、MME と RN1 はそれぞれの Mapping Table に RN2 の情報を追加する (表 3.2.3, 3.2.3)。Mapping Table 更新後、RN1 は UE1 に RA メッセージを送信し、受信した RN2 は RN1 のプレフィクスと自身の Node-ID から IPv6 アドレスを生成する。RN2 の接続によって MME と RN1 のマッピング情報は増加するが、ドメイン内の各ノードは RN2 に対して個別の経路情報を保持する必要がない。したがって、コアネットワーク内の経路情報はムービングセル内に接続する入れ子の RN の台数に依存しない。

3.2.4 UE2 の RN2 への接続

前述の RN2 の接続後に、さらに UE2 が RN2 に接続する場合を考える。UE2 の無線回線確立後の接続手順を表 11 に示す。UE2 は RN2 に RS メッセージを送信する。RS メッセージは UE2 の Node-ID を含む。RS メッセージを受け取った RN2 は RN1 を介して MME と Registration メッセージと Registration Ack メッセージを交換する。その結果、MME、RN1、RN2 はそれぞれの Mapping Table に

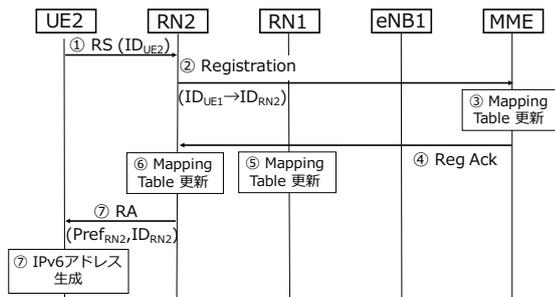


図 11 UE2 の RN2 への接続手順

$ID_{eNB1} \rightarrow Pref_{eNB1}$
$ID_{eNB2} \rightarrow Pref_{eNB2}$
$ID_{eNB3} \rightarrow Pref_{eNB3}$
$ID_{RN1} \rightarrow ID_{eNB1}$
$ID_{UE1} \rightarrow ID_{RN1}$
$ID_{RN2} \rightarrow ID_{RN1}$
$ID_{UE2} \rightarrow ID_{RN2}$

図 12 UE2 の RN2 への接続後に MME が保持する Mapping Table

$ID_{UE1} \rightarrow ID_{RN1}$
$ID_{RN2} \rightarrow ID_{RN1}$
$ID_{RN2} \rightarrow Pref_{RN2}$
$ID_{UE2} \rightarrow ID_{RN2}$

図 13 UE2 の RN2 への接続後に RN1 が保持する Mapping Table

$ID_{UE2} \rightarrow ID_{RN2}$

図 14 UE2 の RN2 への接続後に RN2 が保持する Mapping Table

UE2 の情報を追加する (表 3.2.4, 3.2.4, 3.2.4). Mapping Table 更新後, RN2 は UE2 に RA メッセージを送信し, 受信した UE2 は RN2 のプレフィクスと自身の Node-ID をから IPv6 アドレスを生成する. UE2 の接続によって MME, RN1, RN2 のマッピング情報は増加するが, ドメイン内の各ノードは UE2 に対して個別の経路情報を保持する必要がない. したがって, コアネットワーク内の経路情報量はムービングセル内に接続する UE や RN の台数に依存しない.

3.3 パケットの転送手順

本節では, パケット転送手順を示す. 前述の UE2 が RN2 に接続した状況について, 通信相手である CN-UE 間のパケットの転送を考える.

最初に, インターネットに接続しているノードである CN から UE2 へのパケット送信を考える. パケットの転送における終点アドレスの変化を図 15 に示す. CN は UE2 の Node-ID である ID_{UE2} と UE2 が接続しているドメインの IPv6 プレフィクス $Pref_{dom}$ を知っているものとする. CN

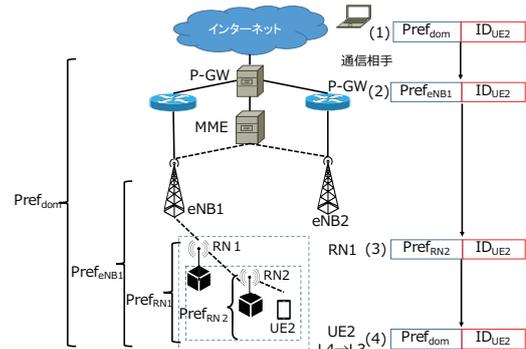


図 15 転送パケットの終点アドレスの変化

は終点アドレスとして図 15-(1) に示す IPv6 アドレスを生成し, 始点アドレスは CN の IPv6 アドレスである $IPv6_{CN}$ となる. インターネット内のルータは $Pref_{dom}$ に関する経路情報を保持しているため, パケットは P-GW に到達する.

P-GW は MME から ID_{UE2} に対応する IPv6 プレフィクスを MME に問い合わせる. MME は表 3.2.4 の 7 行目から UE2 は RN2 に接続していることを, 6 行目から RN2 は RN1 に接続していることを, 4 行目から RN1 は eNB1 に接続していることを, さらに 1 行目から eNB1 の無線インタフェースの IPv6 プレフィクスは $Pref_{eNB1}$ であることを知る. MME はこの結果を P-GW に返送する. P-GW はこの結果をキャッシュする. 以降, 同様の終点アドレスを持つパケットを受信した場合は MME に問合せずキャッシュした情報を用いる. 次に, MME から取得した情報に基づき, P-GW は CN から受信したパケットの終点アドレスを図 15-(2) に示したものに書き換える. このパケットはドメイン内の IPv6 経路制御により RN1 に到達する.

RN1 はこのパケットを受信すると, 表 3.2.4 の 4 行目から UE2 は RN2 に接続していることを, 3 行目から RN2 の無線インタフェースの IPv6 プレフィクスは $Pref_{RN2}$ であることを知る. この情報に基づき, RN2 は受信したパケットの終点アドレスを図 15-(3) に示したものに書き換え, 中継する. このパケットは RN2 に到達する.

RN2 はこのパケットを受信すると, 表 3.2.4 の 1 行目から UE2 は自ノードに接続していることを知り, そのままパケットを中継する.

UE2 はこのパケットを受信すると, ネットワーク層にて終点アドレスをドメインのプレフィクスである $Pref_{dom}$ (図 15-(4)) に書換え, 上位層に渡す. この結果, UE2 と CN のトランスポート層以上からは, $IPv6_{CN}$ と図 15-(1) に示した IPv6 アドレス間で通信しているように見える.

次に, UE2 から CN へのパケット送信を考える. パケット転送における始点アドレスの変化を図 16 に示す. UE2 のアプリケーションは始点アドレスとしてドメインのプレフィクス $Pref_{dom}$ と自身の Node-ID である ID_{dom} を連結

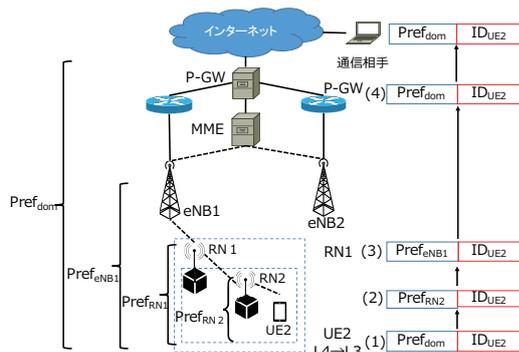


図 16 転送パケットの始点アドレスの変化

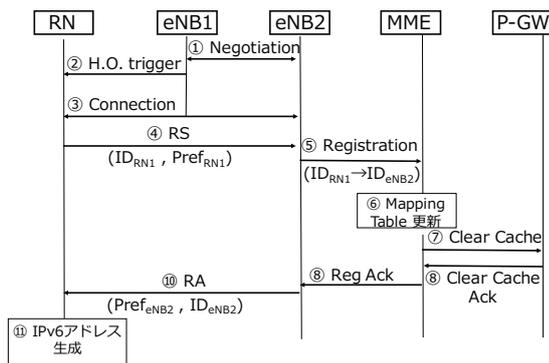


図 17 RN1 の eNB1 から eNB2 へのハンドオーバー手順

した IPv6 アドレスを生成する (図 16-(1)). 終点アドレスは CN の IPv6 アドレスである $IPv6_{CN}$ を指定する. UE2 はネットワーク層にて始点アドレスのプレフィックスを自身のプレフィックス $Pref_{RN2}$ に書き換える (図 16-(2)). このパケットは RN2 を介して RN1 に到達する. RN1 は始点アドレスのプレフィックスを自身のプレフィックス $Pref_{RN1}$ に変換し (図 16-(3)), 中継する. このパケットはドメイン内の経路制御に従って P-GW に到達する. P-GW は始点アドレスのプレフィックスをドメインのプレフィックス $Pref_{dom}$ に示す IPv6 アドレスに書換え (図 16-(4)), パケットを中継する. このパケットはインターネット内の経路制御に従って CN に到達する. この結果, RN2 と CN のトランスポート層以上の視点では $IPv6_{CN}$ と図 16-(1) に示した IPv6 アドレス間で通信が行われているように見える.

以上のように IPv6 アドレスのプレフィックスを変換することによって, パケットは転送される. トンネリングによるヘッダオーバーヘッドは発生せず, コアネットワークの持つ経路情報によってのみパケットはルーティングされる.

3.4 RN1 の eNB1 から eNB2 へのハンドオーバー

本節では, RN1 が eNB1 から eNB2 へハンドオーバーする手順を示す. ハンドオーバーの際のメッセージ交換の手順を図 17 に示す. まず, eNB1 と eNB2 の間で基地局間交渉が行われ, その結果, RN1 の eNB1 から eNB2 へのハンドオーバーが決定する. eNB1 は RN1 へ Handover Trigger

$ID_{eNB1} \rightarrow Pref_{eNB1}$
$ID_{eNB2} \rightarrow Pref_{eNB2}$
$ID_{eNB3} \rightarrow Pref_{eNB3}$
$ID_{RN1} \rightarrow ID_{eNB1}$
$ID_{UE1} \rightarrow ID_{RN1}$
$ID_{RN2} \rightarrow ID_{RN1}$
$ID_{UE2} \rightarrow ID_{RN2}$

図 18 RN1 のハンドオーバー後に MME が保持する Mapping Table

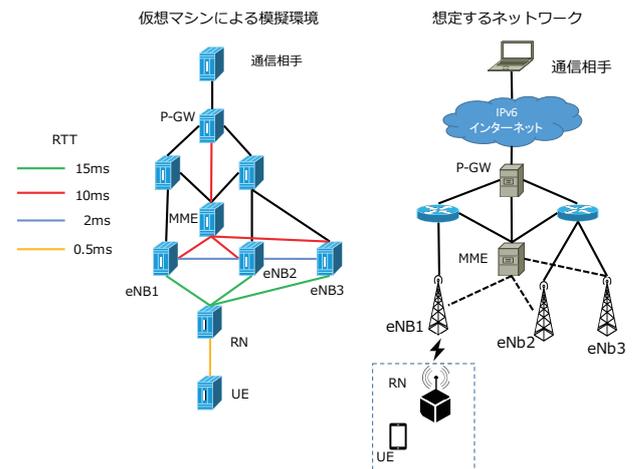


図 19 実装上のネットワーク構成

メッセージを送信し, eNB1 から eNB2 にハンドオーバーすることを RN1 に通知する. 通知を受けた RN1 は eNB2 との無線回線を確立後, eNB2 に RS メッセージを送信する. eNB2 は MME と Registration メッセージを交換し, その結果, MME は自身の持つ Mapping Table の RN1 の情報を移動先の基地局に更新する (表 3.4). この際に, MME は P-GW に Clear Cache メッセージを送信し, P-GW が保持している RN1 のマッピングキャッシュを削除する. Mapping Table 更新後, eNB2 は RN1 に RA メッセージを送信し, 受信した RN2 は eNB2 のプレフィックスと自身の Node-ID から新たに IPv6 アドレスを生成する. RN1 のハンドオーバーに際して, MME の Mapping Table の RN1 の情報が更新されるのみで, マッピング情報は増加しない. また, RN1 の下流に接続するノードは RN1 の Node-ID に関連付けられているため, 移動を知ることなく通信を続けることができる.

4. 実装

PGW, MME, eNB, RN, UE の各ノードをデーモンとして実装し, UE と RN の接続, ハンドオーバーのシグナリングを検証した. ネットワークは, 図 19 の右図を想定し, 10 台の仮想マシンにて左図のトポロジを構成し, 各ノードのデーモンをそれぞれの仮想マシン上に配置した. また, PGW, RN, UE について, カーネル内でのプレフィックスの変換とマッピングキャッシュの保持をするため, カーネ

表 1 MocLis の基本性能評価結果: 処理時間

評価項目	処理時間 / ms
RN が eNB に接続	25.98 ± 0.13
UE が RN に接続	27.77 ± 0.19
RN の基地局間のハンドオーバー	38.45 ± 0.18

ルのネットワーク層の一部を改造した。各仮想マシンの環境は、OS は Ubuntu 14.04 LTS, カーネルは Linux Kernel version 4.9.0 となっている。

PGW はユーザ空間上のデーモンとカーネルモジュールから構成される。カーネルモジュールにて、PGW を通過するパケットのプレフィクスを変換する。もし対応するマッピングキャッシュを保持していない場合はデーモンに通知し、MME からマッピング情報を取得しカーネル内にキャッシュする。RN も PGW と同様にユーザ空間上のデーモンとカーネルモジュールから構成されるが、PGW の様に sMME にマッピングを問合せず、RN 自身が持つマッピング情報を基にプレフィクスを変換する。

5. 評価

本節では、MocLis おける UE1 の eNB1 への接続、UE1 の RN1 への接続、RN1 の eNB1 から eNB2 へのハンドオーバーの手順を検証し、その基本性能を評価した。

5.1 評価環境

図 19 左図において、MME-PGW, MME-eNB 間は近距離間で有線接続していることを想定し、RTT (Round Trip Time) を 10ms とした。eNB-RN 間は無線区間の伝送遅延を考慮し、RTT を 15ms とした。RN-UE 間は近接していることから RTT を 0.5ms とした。

5.2 評価項目、結果

UE1 の eNB1 への接続、UE1 の RN1 への接続、RN1 の eNB1 から eNB2 へのハンドオーバーそれぞれに要する時間を計測した。RN のハンドオーバーは eNB-RN 間の有線リンクの切替によって代替した。それぞれ 10 回ずつ計測し、平均値 ± 標準偏差を結果とした。評価結果を表 1 に示す。

6. 考察

実験の結果、MocLis による接続ならびにハンドオーバーが実行可能であることが確認された。

最も処理時間が短いものは RN の接続であった。UE の RN への接続については、RN の接続と同様の処理に加えて RN-UE 間でメッセージの交換が加わるが、UE と RN は近接していることから遅延は非常に小さいため、RN の接続とほぼ同一の時間で接続が出来ることが分かった。一方、RN のハンドオーバーは一番処理時間が大きくなった。これは、RN と UE の接続とは違い、MME-PGW 間でメ

ッセージを交換し、PGW 内のキャッシュを削除する処理があることが原因と考えられる。3GPP TS 23.203 Version 14.2.0[8] によると、LTE における携帯端末の主な利用用途である動画のストリーミング配信や TCP ベースのサービスの許容遅延時間は 300ms, 音声・ライブストリーミング動画、インタラクティブゲーミングの場合は 100ms である。したがって、本提案手法では RN が基地局間をハンドオーバーした場合も QoE は低下しないと考えられる。これは、ID/Locator 分離によるトンネリング処理の撤廃とそれに伴うシグナリングの低減によると考えられる。

7. おわりに

本稿では、トンネリングを利用しない、ムービングセルをサポート可能なネットワーク制御型モビリティサポートプロトコルの MocLis を提案した。仮想環境で構築したネットワークにて、基本動作の検証、性能評価をした結果、RN, UE の接続とハンドオーバーの基本動作を確認し、プレフィクス変換によるトンネリングを利用しない通信方式が動作することが確認できた。しかし、本稿におけるハンドオーバーは基地局間の切替のみを想定しているため、ハンドオーバー時に送信されたパケットは、ハンドオーバー前の経路に従って移動前の場所に転送される可能性がある。今後は、ハンドオーバー中に移動前の経路に到達してしまったパケットを、移動元の基地局から移動先の基地局に転送する機能が必要になる。

参考文献

- [1] Gundavelli, S., Leung, K., Devarapalli, V., Chowdfury, K. and Patil, B.: Proxy Mobile IPv6, RFC 5213, *IETF* (2008).
- [2] Soto, I., Bernardos, C., Calderon, M., Banchs, A. and Azcorra, A.: NEMO-Enabled Localized Mobility Support for Internet Access in Automotive Scenarios, *IEEE Communications Magazine*, Vol. 47, No. 5, pp. 152–159 (2009).
- [3] Yan, Z., Zhou, H. and You, I.: N-NEMO: A Comprehensive Network Mobility Solution in Proxy Mobile IPv6 Network, *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, Vol. 1, No. 2, pp. 52–70 (2010).
- [4] Arita, T. and Teraoka, F.: PNEMO: A Network-Based Localized Mobility Management Protocol for Mobile Networks, *Journal of Information Processing*, Vol. 20, No. 2, pp. 1–11 (2012).
- [5] Lee, J.-H., Ernst, T. and Chilamkurti, N.: Performance Analysis of PMIPv6-Based Network Mobility for Intelligent Transportation Systems, *IEEE Transactions on Vehicular Technology*, Vol. 61, No. 1, pp. 74–85 (2012).
- [6] Zhou, X., Korhonen, J., Williams, C., Gundavelli, S. and Bernardos, C.: Prefix Delegation Support for Proxy Mobile IPv6 (2014). RFC7148.
- [7] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C. and Carney, M.: Dynamic Host Configuration Protocol for IPv6 (DHCPv6) (2003). RFC3315.
- [8] 3GPP: 3GPP TS 23.203, <http://www.3gpp.org/DynaReport/23203.html>.