

## LIN6 に基づくネットワークモビリティプロトコルの設計と実装

大岩拓馬<sup>†1</sup> 國司光宣<sup>†2</sup> 石山政浩<sup>†3</sup> 河野通宗<sup>†4</sup> 寺岡文男<sup>†1</sup><sup>†1</sup> 慶應義塾大学理工学部管理工学科 <sup>†2</sup> 慶應義塾大学大学院理工学研究科<sup>†3</sup> 東芝研究開発センター通信プラットフォームラボラトリー <sup>†4</sup> ソニーコンピュータサイエンス研究所

## 1 はじめに

今後、乗物に設置された LAN に乗客が携帯機器を接続し、乗物ごと携帯機器群も移動するようになるだろう。すなわち移動ネットワークである。

Mobile IPv6(MIPv6)[1] や LIN6[2] をはじめ、単一のホストに移動透過な通信を可能にするプロトコルは多く研究されている。しかし、ネットワーク全体が移動し同時に多数のホストが移動通信を行う状況では、それらのプロトコルは適していない。ネットワークの移動時に移動ネットワーク内部に接続しているノードが各々独立に位置登録処理を行うと、登録に必要なパケットは移動ネットワーク内部のノードの数に比例して増加してしまう。この時ネットワークに膨大なトラフィックを発生させ、遅延を生じさせる。

ネットワークに対して移動透過性を提供するプロトコルをネットワークモビリティ(NEMO)プロトコルと呼び、無駄の少ない効率的な NEMO プロトコルを設計・提案することを本研究の目的とする。

## 2 関連研究

既存の NEMO プロトコルの多くは MIPv6 に基づき拡張したものである。ここでは、MIPv6 を拡張した NEMO プロトコルである RNEMO[3] について説明し、その後 MIPv6 に基づくプロトコル全体で共通の問題点について述べる。

## 2.1 RNEMO

RNEMO は Base RNEMO protocol と Dynamic Route Optimization Mechanism の二つの主要機能からなる。Base RNEMO の動作概要を図 1 に示す。移動ネットワークと外部インターネットを接続する役割を担うルータを Mobile Router(MR) と呼ぶ。

RNEMO では MR のインターネット側インタフェースに、移動先のサブネットで一時的に割り当てられる Mobile Router Care-of Address(MR-CoA) と、移動ネットワーク側インタフェースに MR の移動によって変化しない Mobile Router Address(MR-A) を割り当てる。Primary Router Proxy Agent(PRPA) が prefix binding という MR-A と MR-CoA の対応関係を保持し、MIPv6 の Home Agent(HA) のように移動ネットワーク内のノード宛のパケットをトンネリングして転送する役割を果たす。移動ネットワークが移動先サブネットで獲得した MR-CoA を PRPA へ登録すると、PRPA には prefix binding が生成される。移動ネットワーク宛のパケットは PRPA に到達し、PRPA はそのパケットを prefix binding にある MR-CoA 宛へトンネリングして MR へ転送する。パケットが MR に到達するとトンネル部分が取り除かれ、移動ネットワーク内のノードへ転送される。MIPv6 と同様に通信ノード (CN) が prefix binding を保持する時、経路制御ヘッダに MR の MR-CoA を格納することでパケットは MR へ直接転送される。

Dynamic Route Optimization Mechanism は最適ではないが、より短い経路にする。Router Proxy Agent(RPA) は、PRPA と同様に prefix binding を保持し MR へパケットを

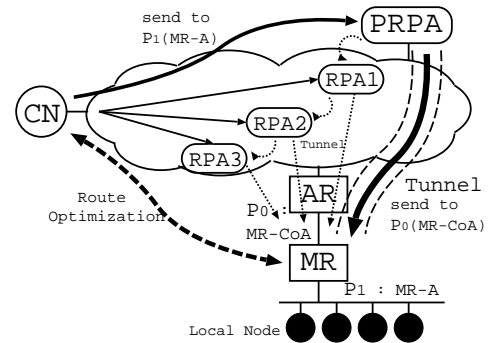


図 1: Base RNEMO の動作概要

転送するもので、インターネット上に任意に設置する。CN と MR の間に PRPA を通過するよりも経路が短くなる RPA があるなら、パケットはその RPA を通過し MR へ転送される。この経路は最適ではないが三角経路の冗長度は小さい。CN と MR 間の経路がより短くなる RPA1 の検索には OSPF shortest path アルゴリズムなどを用いる。

## 2.2 Mobile IPv6 に基づくプロトコルの問題点

RNEMO では冗長経路となってしまう問題をできるだけ小さくするための手法が導入されてはいるが、本質的には MIPv6 のトンネリングによるヘッダオーバーヘッドや冗長経路の問題を依然として抱えたままである。

移動ネットワークがネストする場合には、冗長経路やヘッダオーバーヘッドの問題はさらに深刻となる。ネストする場合、送信されたパケットは MR の数だけ各 HA を通過し、その度にトンネリングされて転送されるので、ヘッダオーバーヘッドは非常に大きくなる (図 2)。

RNEMO に限らず、MIPv6 に基づくプロトコルは必ず MIPv6 の問題点を継承する。Internet-Draft でも MIPv6 に基づく NEMO プロトコルの問題の解決手法が多く提案されている。しかし、これらの多くは問題をできるだけ小さくするもので、本質的な解決手法ではない。また問題の軽減するプロセスによって別のオーバーヘッドが増加してしまうことも少なくない。

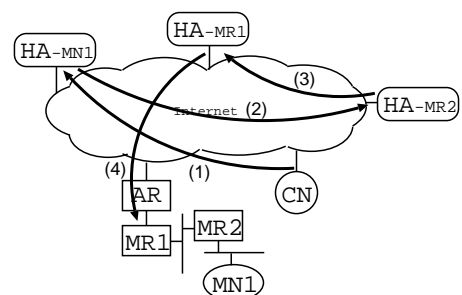


図 2: Mobile IPv6 に基づく NEMO プロトコルの冗長経路

### 3 LIN6

本論文では、MIP6 が抱える問題を解決し優れた特性を持つ LIN6 に基づくネットワークモビリティプロトコルを提案する。

#### 3.1 LIN6 の基本概念

LIN6 ではネットワークアドレスが持つ位置指示子とノード識別子を概念的に分離させる。ノード識別子を用いて位置に依存しないコネクションを確立し、位置指示子を用いて経路制御をすることで移動透過性を保証する。

ノード識別子は LIN6 ID と呼ばれるグローバルユニークな識別子で、位置指示子には既存の IPv6 のネットワークプレフィックスを用いる (図 3)。LIN6 アドレスが実際にパケット配送に用いられ、LIN6 汎用識別子は位置に依存しない IPv6 アドレスと見なせる。

IPv6 アドレス	Network Prefix	Interface ID (EUI-64)
LIN6 アドレス	Network Prefix	LIN6 ID
LIN6 汎用識別子	LIN6 Prefix	LIN6 ID
	Network Prefix	Node ID

図 3: IPv6 アドレス, LIN6 アドレス, LIN6 汎用識別子

#### 3.2 LIN6 の通信手順

Mapping Agent(MA) は LIN6 ID と現在のネットワークプレフィックスの対応関係を保持し、ノードから要求を受けると位置情報を返す。DNS には LIN6 ID を管理する MA のアドレスが記述されている。

CN は DNS から MN に対応する MA のアドレスを取得し、この MA に MN の位置情報を要求する。MA から取得したネットワークプレフィックスと LIN6 ID から MN 宛のパケットの宛先アドレスを生成する。MN が移動すると MA が管理する位置情報を更新し、CN に Mapping Update を送信する。Mapping Update は、通信していた CN がキャッシュしている MN の位置情報を更新するためのメッセージである (図 4)。

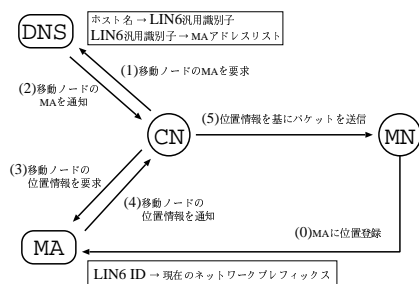


図 4: LIN6 の通信手順

LIN6 には常に最適経路でヘッダオーバーヘッドなしに通信できるという優れた特性がある。

### 4 提案プロトコルの設計

図 5 を用いて本提案プロトコルの設計・通信手順を説明する。本提案プロトコルにおける構成ノードは、通常の LIN6 と同様の MN, CN, MA に加え Mobile Router (MR) がある。図 5 は MR1 が管理する移動ネットワーク内に MR2 が管理する移動ネットワークが移動して接続した状態を示している。

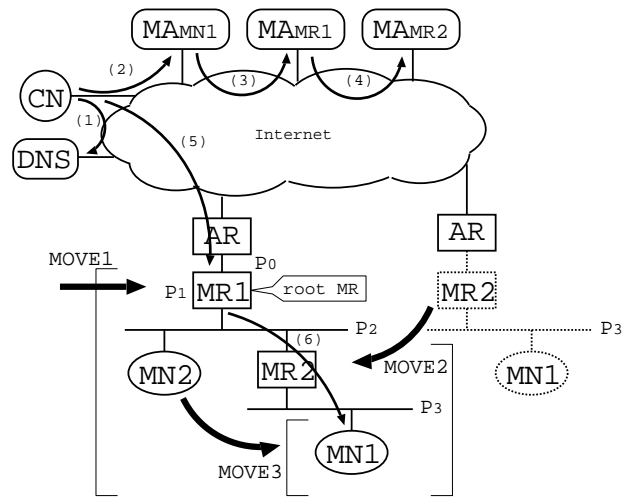


図 5: 提案プロトコルの通信手順

#### 4.1 設計指針

前章までで述べた問題点や NEMO に対する要求事項をふまえ、次のような設計指針とする。

- 移動ネットワーク内のノードの位置登録パケットなどの制御パケットの不必要な外部への送信を避ける。
- 常に最適経路でヘッダオーバーヘッドなしに通信可能という、LIN6 の特性を維持する。
- LIN6 によって単独に移動できるノードは移動ネットワーク内外に自由に移動し、通信を継続できる。
- 移動ネットワークは別の移動ネットワークに接続することができ、任意数の移動ネットワークのネストが可能である。

#### 4.2 Mobile Router (MR)

MR は移動ネットワークと外部のインターネットを接続する役割を担う。図 5 で示される MR1 のようにインターネットに直接接続する MR を、MR2 のように別の移動ネットワーク内に接続する MR と区別し、root MR と呼ぶ。root MR のネットワークプレフィックス (図 5: P1) を root MRLoc と定義し、移動ネットワーク内の全ての移動ノードは root MRLoc をキャッシュする。

MR が移動し MR が root MR の時、AR から受け取る Router Advertisement にはよらず、Prefix Delegation によって 2 つ以上のネットワークプレフィックスを獲得するものと仮定する。ひとつはインターネットに接続するインタフェースに付けられ、もうひとつは MR が持つ疑似インタフェースに付けられるものである。移動ネットワーク内に接続するノードへのパケットはこの疑似インタフェースに付けられたネットワークプレフィックスへ向けられて送信されるものとする。

MR は自身の LIN6 ID を Router Advertisement (RA) に付加した拡張 RA を移動ネットワーク内に送信する。MN, MR は移動先で受信した RA が LIN6 ID が付加されたものである時、別の移動ネットワーク内に接続したと判断する。

#### 4.3 Mapping Agent(MA)

LIN6 の MA は LIN6 ID とそれに対応する位置情報との Mapping を保持する。本提案プロトコルではこの MA を拡張

し、LIN6 ID に対応する保持情報はネットワークプレフィックスだけではなく、ネットワークプレフィックスと LIN6 ID のどちらかを格納する。また、どちらが格納されているかを判断するためのフラグも保持する。

位置情報を要求された LIN6 ID に対して Mapping されている情報がネットワークプレフィックスではなく LIN6 ID なら、その LIN6 ID の位置情報を管理する別の MA に間接的に位置情報を要求する。

#### 4.4 MR1 が移動した時の処理

MR1 と MR1 に接続するネットワーク全体が移動した時の処理を説明する (図 5:MOVE1)。AR から受信した RA に LIN6 ID が含まれないことから、MR1 は自身が root MR であると判断する。MR1 は root MR なので Prefix Delegation によって二つのネットワークプレフィックス ( $P_0, P_1$ ) を取得し、一方 ( $P_1$ ) を疑似インタフェースに割り当て  $MA_{MR1}$  に登録する (図 6)。また、疑似インタフェースに割り当てたネットワークプレフィックス ( $P_1$ ) を root MRLoc として、MR1 の移動ネットワーク内のサブネット ( $P_2$ ) に接続するノードに通知する。この通知を受け取った MR2 はこの root MRLoc とキャッシュされている root MRLoc を比較する。キャッシュと異なる場合はキャッシュを更新し、通信中ノードがあるならそのノードに root MRLoc を付加した Mapping Update を送信する。さらに MR2 は MR1 が行ったように、自身の下位のサブネットへ root MRLoc を通知する (図 7)。

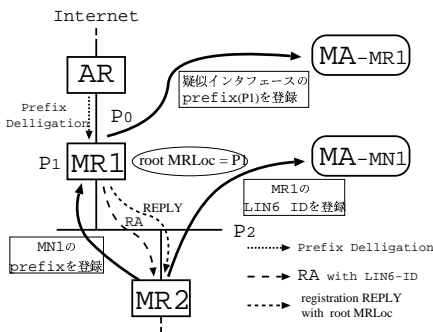


図 6: 移動時における位置登録処理

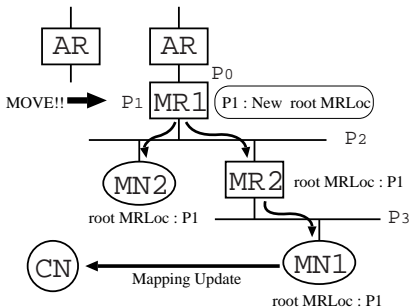


図 7: root MRLoc 変更検知後の処理

#### 4.5 MR2 が移動した時の処理

MR2 と MR2 が管理する移動ネットワークが移動し、MR1 の管理する移動ネットワークに接続した時の処理を説明する (図 5:MOVE2)。受信した RA に LIN6 ID ( $ID_{MR1}$ ) が含

まれることから、MR2 は自身が root MR ではなく別の移動ネットワーク内に接続したと判断する。この時特別な登録処理を行う。MR1 が送信した RA から取得したネットワークプレフィックス ( $P_2$ ) は MR1 に登録し、RA に含まれる ( $ID_{MR1}$ ) を  $MA_{MR2}$  に登録する。MR1 は MR2 の登録処理に対する REPLY メッセージに、自身がキャッシュしている root MRLoc ( $P_1$ ) を付加して送信する (図 6)。MR2 はこの REPLY メッセージに含まれる root MRLoc と自身のキャッシュを比較する。キャッシュと異なるなら、通信中ノードに Mapping Update を送信し、MR2 の下位サブネットに root MRLoc を通知する (図 7)。

MR2 の移動後、MR1 への登録処理と同時にもうひとつのメッセージを MR1 へ送信する。MR2 の Mapping Table には自身の下位のサブネットに接続する全てのノードの位置情報が登録されている。MR2 は Mapping Table の情報を全て含むメッセージを MR1 へ送信する。MR1 はこのメッセージを受信し Mapping Table に追加する。この時点で MR1 の Mapping Table は MR2 の Mapping Table を全て含有することになる (表 1)。さらに、MR1 は MR2 から受信した情報から、MR2 の下位サブネットのネットワークプレフィックスを全て抽出する。そして抽出したネットワークプレフィックス宛の packets の next hop がメッセージの送信元の MR2 になるようにルーティングテーブルを更新し、表 2 のようになる。

表 1: 各 MR の Mapping Table

MR	LIN6 ID	Network Prefix
MR1	$ID_{MR2}$	$P_2$
	$ID_{MN1}$	$P_3$
MR2	$ID_{MN1}$	$P_3$

表 2: MR1 の Routing Table

MR	Destination	Gateway
MR1	$P_3$	MR2

#### 4.6 MN2 が移動した時の処理

MN2 が MR1 の下位サブネットから移動し、同移動ネットワーク内の MR2 の管理する移動ネットワークに接続する時の処理を説明する (図 5:MOVE3)。受信した RA に LIN6 ID ( $ID_{MR2}$ ) が含まれることから、MN2 は移動ネットワーク内に接続したと判断する。そして、MR2 が送信した RA から取得したネットワークプレフィックス ( $P_3$ ) と  $ID_{MN1}$  との Mapping を MR2 に登録し、RA に含まれる  $ID_{MR2}$  を  $MA_{MN1}$  に登録する。MR2 が送信する REPLY メッセージに含まれる root MRLoc ( $P_1$ ) は MN2 がキャッシュしている root MRLoc と等しい。この時、登録処理だけを行い Mapping Update は送信しない。

#### 4.7 通信プロセス

本プロトコルの通信プロセスは、MN 宛の packets は MN が接続する移動ネットワークの root MR に packets が到達し、root MR が packets の宛先アドレスのネットワークプレフィックス部分を MN に現在割り当てられているネットワークプレフィックスに書き換えて転送し、MN に到達する

ものである。このプロセスに必要な登録情報や経路情報は、前節に説明した移動時の処理によって完成している。表 1, 表 2, 表 3 は図 5 において生成された登録情報と経路情報である。

表 3: 各 MA の Mapping Table

MA	LIN6 ID	Prefix or LIN6 ID	Indirect
$MA_{MR1}$	$ID_{MR1}$	$P_1$ (root MRLoc)	0
$MA_{MR2}$	$ID_{MR2}$	$ID_{MR1}$	1
$MA_{MN1}$	$ID_{MN1}$	$ID_{MR2}$	1

表 1, 表 3 は各々の MA, MR の Mapping Table を表し、保持する MA, MR と LIN6 ID に対応するネットワークプレフィックス、または LIN6 ID を示す。また、表 3 に示すように、MA の Mapping Table には Indirect bit フラグがあり、LIN6 ID に対応するものが LIN6 ID であればフラグを立てる。

表 2 は MR に動的に追加されたルーティングテーブルのエントリを表し、MR1 では  $P_2, P_3$  宛の packets は MR2 に中継することを示している。

#### 4.7.1 CN と MN1 の通信例

CN は DNS から MN1 の位置情報を管理する  $MA_{MN1}$  のアドレスを獲得し、 $MA_{MN1}$  に MN1 の位置情報を要求する。表 3 にあるように  $MA_{MN1}$  には MR2 の LIN6 ID ( $ID_{MR2}$ ) が登録されているので、 $MA_{MN1}$  は  $MA_{MR2}$  に位置情報を要求する。 $MA_{MR2}$  には MR1 の LIN6 ID ( $ID_{MR1}$ ) が登録されているので、さらに  $MA_{MR2}$  が  $MA_{MR1}$  に位置情報を要求し、ここで初めて位置情報が取得され CN に通知される。こうして最終的に獲得した位置情報は root MR のネットワークプレフィックス (root MRLoc :  $P_1$ ) である。

取得した root MRLoc ( $P_1$ ) をネットワークプレフィックス、MN1 の LIN6 ID ( $ID_{MN1}$ ) をノード ID として LIN6 アドレスを生成し、送信パケットの宛先アドレスとする。このパケットは MR1 に到達し、MR1 はパケットの宛先アドレスから LIN6 ID ( $ID_{MN1}$ ) を抽出し、保持している Mapping Table から  $ID_{MN1}$  に対応するネットワークプレフィックスを検索する。そして宛先アドレスのネットワークプレフィックス部を対応するネットワークプレフィックス ( $P_3$ ) で書き換えてパケットを転送する (図 8)。移動ネットワーク内部では表 2 のようなルーティングテーブルができていたので、このパケットは経路制御に従って MN1 に到達する。

MN1 から CN へのパケットの送信元アドレスのネットワークプレフィックスは MN1 が接続するサブネットのネットワークプレフィックス ( $P_3$ ) である。この送信元アドレスのパケットは root MR と外部インターネットの接続点の ingress フィルタに捕捉される可能性がある。root MR が移動ネットワーク内から外部へのパケットの送信元アドレスのネットワークプレフィックスを root MRLoc に書き換えることで、この問題を回避する (図 8)。

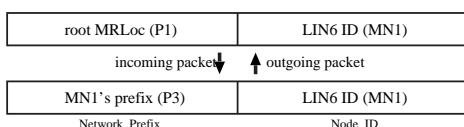


図 8: root MR におけるアドレス書き換え

## 5 実装

本プロトコルを NetBSD1.6K 上で実装し、以下の環境で実験を行った。

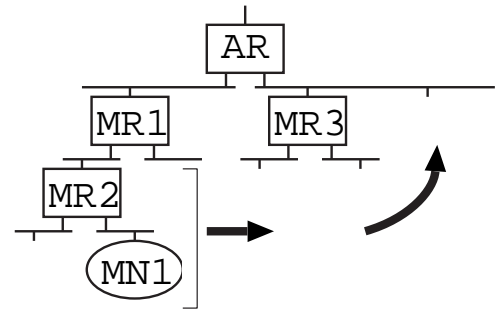


図 9: 実験環境

移動ネットワーク外のノードと MN1 が通信しながら図 9 の矢印のように移動し、通信が継続できることを確認した。今後登録処理やハンドオフなどの処理時間を測定し、本プロトコルの性能評価を行う予定である。

## 6 まとめ

本論文では LIN6 に基づく NEMO プロトコルを提案した。本提案プロトコルは LIN6 のアドレス構造を応用することで、移動ネットワークのネストの有無に関わらず常に最適経路で、かつヘッダオーバーヘッドなしに、ネットワークに対して移動透過性を提供することを可能とした。移動ネットワークのネストによる影響は、MA への位置情報の要求の回数のみである。MA への位置情報の要求は通信を始める時だけなので、ネットワークに対するトラフィックの影響や遅延の影響は無視できる程度のものである。

## 参考文献

- [1] David B. Johnson, Charles E. Perkins. Mobility Support in IPv6. Internet Draft. work in progress. IETF. Apr. 2002
- [2] M. Kunishi and M. Ishiyama and K. Uehara and H. Esaki and F. Teraoka. LIN6: A New Approach to Mobility Support in IPv6. The Third International Symposium on Wireless Personal Multimedia Communications. Nov 2000
- [3] Ryuji Wakikawa, Susumu Koshihara, Keisuke Uehara, Heikki Kokkinen, Jun Murai. RNEMO: Reactive Network MObility Protocol with Dynamic Route Optimization. ICCCN. 2002
- [4] Timothy J. Knveton. Jari T. Malinen. Vijay Devarapalli. Charles E. Perkins. Mobile Router Support with Mobile IP. Internet Draft. work in progress. IETF. July. 2002