エンドエンドで移動透過性を実現するMobile PPCの提案と実装

竹内元規[†] 鈴木秀和[†] 渡邊晃[†]

無線ネットワーク環境の広がりにより，多くのモバイル端末がどこからでもネットワークに接続できるようになってきている．このような背景から，ネットワークに接続しながら自由に移動したいというニーズが広まっている．しかし，TCP/IP ではネットワークに接続中の端末が移動すると IP アドレスが変化し，通信が切断されてしまう．このため通信中に移動しても，通信に影響を与えない移動透過性が要求される．これまで移動透過性を実現する技術はいくつか提案されているが，多くの場合，移動透過性を実現するために特有の装置による基盤が必要となる．また IPv4 で唯一実用となっている Mobile IP は経路が冗長になったり，パケット長が変わったりするなどのオーバヘッドが増加する課題がある．このような課題は，今後普及する P2P 通信の特徴を損なううえ，二重化などの措置をとるために，管理負荷が増すなどの課題がある．そこで本論文では，モバイル端末の IP アドレスが変化した場合に，両エンド端末においてアドレス変換処理を実行する Mobile PPC（Mobile Peer to Peer Communication）を提案する．Mobile PPC は経路の冗長が発生せず，パケット長が変化しないため高スループットを実現できる．また既存端末との上位互換性を有しており，段階的な普及が可能である．Mobile PPC を FreeBSD 上に実装し検証をした結果，高スループットを確保したままで移動透過通信が行えることを示した．

A Proposal of Mobile PPC that Realizes End-to-End Mobility and Its Implementations

Motoki Takeuchi,[†] Hidekazu Suzuki[†] and Akira Watanabe[†]

In ubiquitous world, there are strong needs that we can communicate each other continuously while we move. However, in TCP/IP, an IP address of a terminal changes when the terminal moves to other places, and the communication breaks. Therefore there needs a technology called mobile transparency that does not break communications as terminals move. There have been proposed some methods that can achieve mobile transparency, however, most methods need a server having special functions in the system. Such methods do not fit in the spreading P2P communications, and extra management loads are needed because they usually need a measure of duplication of the server. In this paper, we propose Mobile PPC (Mobile Peer to Peer Communication) that realizes mobile transparency by executing IP address translation in both end terminals when IP addresses change. Mobile PPC dose not need any extra devices. Also it dose not need to alter upper layer software in terminals, and it can achieve high throughput because packet sizes do not change. We implemented Mobile PPC in FreeBSD and evaluated the system. As a result, it is shown that mobile transparency can be achieved with high throughput.

1. はじめに

ノート PC や PDA などのモバイル端末を持ち歩き，行く先々でインターネットに接続して利用するユーザが増加している．このような状況下では，通信中にユーザが移動しても，通信を継続できることが要求される．TCP/IP では，IP アドレスがノード識別子としての役割だけでなく位置の情報も含んでいるため，端末がネットワークを移動すると異なる IP アドレスが割り振られる．トランスポート層では IP アドレスが通信識別子の一部として用いられており，IP アドレスが異なると別の通信と見なされ通信を継続することができない．この課題を解決するために，端末が移動しても通信を継続できる機能を移動透過性と呼び，これまで多くの方式が研究されている[1)]．移動透過性とは端末の位置に依存せず通信を開始できる移動ノード到達性，端末が移動しても通信端末との間に確立したコネクションを維持する通信継続性に分けられ，両者を実現する必要がある．

移動透過性の研究を大きく分類すると，特殊な中継サーバを用いるプロキシ方式とそれを必要としない

† 名城大学大学院理工学研究科
Graduate School of Science and Technology, Meijo University

エンドツーエンド方式がある．プロキシ方式は，移動ノードと通信相手ノードの間にプロキシサーバが介在し，プロキシサーバが移動ノードの IP アドレス変化を通信相手ノードから隠蔽する．エンドツーエンド方式はエンド端末間で課題を解決し，上位ソフトウェアに対して IP アドレスの変化を隠蔽する．また，別の分類方法として，移動透過性を実現するレイヤの違いにより，ネットワーク層で実現する方式とトランスポート層で実現する方式がある．トランスポート層では通信識別子の制御がやりやすいという利点があるが，TCP または UDP のどちらに適用するかによりその方式が異なる．これに対し，ネットワーク層での実現方法は，TCP/UDP のいずれにも対応できる点で有効である．

Mobile IP [2)~6)] は，プロキシ方式をネットワーク層で実現する．プロキシサーバとして移動ノード MN（Mobile Node）の位置を管理するホームエージェント HA（Home Agent）を導入し，通信相手ノード CN（Correspondent Node）側から MN への通信パケットは HA が代理受信し，MN へトンネリング転送を行う．MN 側から CN への通信パケットは直接送信される．CN は通信相手が HA のように見えるため，MN が移動しても通信識別子が変化せず通信を継続できる．Mobile IP は IETF での十分な検討を経て確立された技術であるが，HA という特殊な装置が必要であるほか，通信経路に冗長が発生したり，トンネリング転送時に余分なヘッダが必要になったりするなどの問題点がある．

MSOCKS [7)] は，プロキシ方式をトランスポート層で実現する．プロキシサーバとして，socks サーバを導入する．DNS サーバには，MN のホスト名に対して socks サーバの IP アドレスを登録する．CN は socks サーバが通信端末であると認識する．socks サーバは MN と socks サーバ間，socks サーバと CN 間で確立された異なる TCP コネクションを結合しなおすことにより通信を継続する．MSOCKS は，ヘッダオーバヘッドは発生しないが，両方向の通信とも socks サーバを経由するので冗長な経路が発生する．

TCP-R [8)]，TCP Migrate [9)]，MMSP [10)] はエンドツーエンド方式をトンランスポート層において実現する．TCP-R，TCP Migrate は，MN の IP アドレスが変化したときに，TCP オプションフィールドを用いて MN から CN に変更情報を通知し，エンド端末間で TCP コネクションを張り直す．この方式では，TCP 機能の拡張が必要であり，またアプリケーションも TCP に限定される．MMSP は，UDP を拡張し，MN の IP アドレスが変化したときに，独自に定義したパケットで相手に通知する．IP アドレスの通知が完了するまでの間，新旧の IP アドレスを保持しておくことなどによりパケットロスを軽減する工夫をしている．この方式では，アプリケーションが MMSP に対応している必要があり，かつ UDP に限定される．

LIN6（Location Independent Networking for IPv6）[11)]，MAT（Mobile IP with Address Translation）[12)] はエンドツーエンド方式をネットワーク層において実現する．LIN6 では，IPv6 アドレス空間の内容をノード識別子と位置指示子という 2 種類の空間に分離させ，ノード識別子と IP アドレスの対応を保持する位置管理装置を設けることにより，IP アドレスの変化を上位ソフトウェアから隠蔽する．しかし，LIN6 では，アドレス空間のビット数が半分になることからアドレス利用効率が大きく低下するうえ，独自のアドレス体系をグローバルユニークに割り当てる必要がある．また，IPv4 ではアドレス空間が不足するため適用できない．MAT は，LIN6 のこのような課題を解決するもので，アドレス空間を分割することはせず，ノード識別子と位置指示子に対応する IP アドレスを別途定義して両者を変換する．この方式では通常の IP アドレス体系を適用することができ，IPv4 でも同様の考えを適用できる．しかし，独自の位置管理装置が必要になる点は変わっていない．また，MAT 非対応のノードは，通信開始時に MN がホームネットワーク上にいないと MN の位置指示子となる IP アドレスを知ることができず，通信を開始することができないという課題がある．

Mobile IPv6 [13)] は，Mobile IP を IPv6 用に拡張したもので，MN が移動後に経路最適化と呼ぶ機能が標準で追加され，冗長な経路を通さない通信が可能となった．しかし，通信開始時には HA を経由しなければならないため，HA が必須となることに変わりない．また，経路最適化時にはヘッダのオーバヘッドが常時発生する．

今後のユビキタス社会を想定するとネットワークを最大限に活かせる P2P（Peer-to-Peer）通信の要求がますます増加すると考えられる．ここで P2P 通信とは IP 電話のようなリアルタイム性を必要とするアプリケーションが想定されるべきである．プロキシ方式のように一般通信において特殊な装置を経由する方式では，P2P 通信の特徴である柔軟性やリアルタイム性が失われる懸念がある．また，エンドツーエンド方式でも特殊な位置管理装置を必要とする方式は，十分な普及に至るまでその機能が発揮できないうえ，サーバ

の二重化などの対策が必須であり管理負荷が大きい．P2P 通信が個人間の通信が主体となることをふまえると，エンドツーエンド方式でかつ，特殊な位置管理装置を必要せずに移動透過な通信を実現できることが望まれる．また，実装レイヤについては，TCP/UDP の区別なく利用可能なネットワーク層での実現方法が有利と考えられる．さらに，現状のネットワークは IPv4 が主体であることから，IPv4 での実装が可能であることが望ましい．

本論文では，エンド端末の IP 層にアドレス変換処理機能を導入し，エンドツーエンド方式をネットワーク層で実現する Mobile PPC（Mobile Peer to Peer Communication）を提案する．本提案では移動ノード到達性と通信継続性を明確に分離する．移動ノード到達性には既存の Dynamic DNS（DDNS）[14] を適用し，通信継続性に対して Mobile PPC を適用する．Mobile PPC では，MN の IP アドレスが変化したとき，MN から CN に対して変化情報を直接報告し，両端末の IP 層の中にアドレス変換テーブルを生成する．以後の通信パケットは上記アドレス変換テーブルに基づきアドレス変換する．この方式により，IP アドレスの変化は上位ソフトウェアから隠蔽することができ，通信継続性を容易に実現することができる．Mobile PPC は IPv4/IPv6 の両者に適用可能であり，かつ既存システムとの上位互換性を有していることから，段階的な普及が期待できる．Mobile PPC を FreeBSD の IPv4 上に実装し，動作確認と性能測定を実施した結果，Mobile PPC はスループットの低下がほとんどない通信継続性を実現できることを確認した．

以下，2 章で従来技術の例として Mobile IP，LIN6，MAT について記述し，3 章で Mobile PPC の原理と詳細について記述する．4 章で Mobile PPC の実装，5 章で性能測定結果とセキュリティおよび既存システムとの互換性について考察する．最後に 6 章でまとめる．

2. 従来技術

従来技術として，プロキシ方式の代表 Mobile IP，エンドツーエンド方式の代表 LIN6，MAT を取り上げる．いずれもネットワーク層による実現方法であり，トランスポート層より上位のソフトウェアにいっさい影響を与えないという利点がある．

2.1 Mobile IP

図 1 に Mobile IP の通信を示す．MN は移動によって変化しないホームアドレス HoA と，移動先ネットワークで割り当てられる気付アドレス CoA の 2 つの

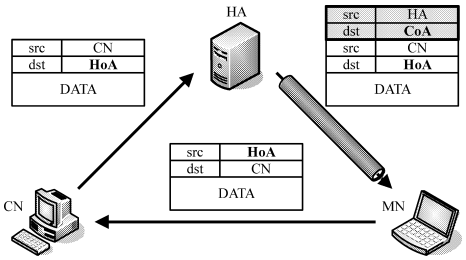


図 1 Mobile IP の通信
Fig. 1 Communication of Mobile IP.

IP アドレスを持つ．HA は，MN の HoA と CoA の対応付けを行い，HoA 宛のパケットを代理受信し，CoA 宛に転送する役割を持つ．

Mobile IP の動作は，HA への登録とデータ通信に分けることができる．MN は別のネットワークへ移動した場合，移動先のネットワークで新しく取得した CoA を HA へ登録する．HA は MN の HoA と CoA の対応付けを更新する．CN から MN へ通信パケットを送信する場合は，宛先を HoA とする．HA はこのパケットを代理受信し，CoA 宛の IP ヘッダでカプセル化して MN に転送する．MN から CN への通信パケットは CN 宛に直接送信される．このとき送信元アドレスは HoA とする．

Mobile IP は，このように HA という特殊な装置を導入し，CN がつねに HA と通信しているように見せかけることにより移動透過性を実現する．MN 宛のパケットは必ず HA を経由するため，通信経路が冗長な三角経路となり，HA と MN 間は IP トンネルとなる．また，MN から CN へパケットを送信する場合に，送信元アドレスとして使われる HoA は MN のインターネット上での位置を正しく表していないため，途中のルータが送信元アドレスを偽っている不正パケットと見なし，破棄する可能性がある．

Mobile IP は，クライアントサーバ環境においては，CN として従来の固定サーバをそのまま利用できる点で有効である．しかし，P2P 通信が主体となる今後のネットワーク環境においては，必ずしも最適な方式とはいえない．

2.2 LIN6

LIN6 は，IP アドレスに含まれているノード識別子と位置指示子としての情報を明確に分離させ，IPv6 アドレス体系自体を見直す提案である．すなわち IPv6 アドレスの上位 64 ビットを位置指示子，下位 64 ビットをノード識別子として扱う．また，上位 64 ビットに対し LIN6 プレフィックスと呼ばれる固定値を定義しておき，IP 層よりも上位層ではノード識別子と LIN6

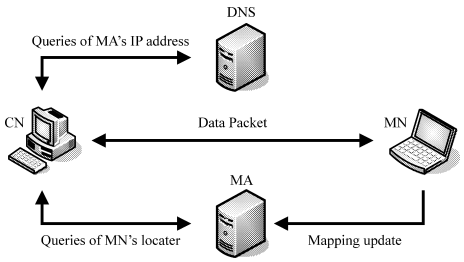


図 2 LIN6 の通信方式
Fig. 2 Communication of LIN6.

プレフィックスを合わせた LIN6 汎用アドレス，下位層では位置指示子とノード識別子を合わせた LIN6 アドレスとなるように IP 層で変換を行う．上位層ではノードの位置や移動にかかわらずつねに LIN6 汎用アドレスを用いる．

図 2 に LIN6 の通信方式を示す．LIN6 はエンドツーエンド方式であるため両端末は対等の関係にあるが，説明のため移動する側の端末を MN，通信相手側の端末を CN と呼ぶ．MA（Mapping Agent）は MN のノード識別子と現在の位置情報との対応関係を常時保持している．CN が MN の LIN6 アドレスを知るためには，DNS からまず MA の IP アドレスを知り，MA から MN の LIN6 アドレスを取得する．MN が CN の LIN6 アドレスを知るときも同様の手順をとる．MN が CN と通信中に別のネットワークに移動した際には MA に位置指示子に変化を通知し，CN に対して MA から MN の LIN6 アドレスを再取得するように通知する．

LIN6 は，上記のように IP アドレスの役割を明確に分割したという点で評価できるが，IPv6 のアドレス構造を 2 分割するためアドレスの利用効率が大きく低下する．さらに，独自のアドレス体系を持つことになるため，ノード識別子のグローバルユニークな割当てが必要となりその管理機構が必要になる．また，位置管理装置として MA のような特殊な装置が必要になる．IPv4 に対してはアドレス空間を分割する余裕がないため適用が困難である．

2.3 MAT

MAT は，LIN6 と同様にノード識別子（ホームアドレス）と位置指示子（モバイルアドレス）を示す 2 つの IP アドレスを定義しているが，両者の対応関係（以下，マッピング情報）を保持する位置管理装置 IMS（IP Address Mapping Server）をネットワーク上に設置し，両者の間でアドレス変換を行う点が異なる．

MAT も LIN6 と同様に DNS から IMS のアドレスを取得する．通信相手の IP アドレスを知る順序は，LIN6 における MA を IMS に置き換えたものと似ている．ただし，MN から CN へ通信を開始する場合には，新規に定義した IP ヘッダオプションを用いて，MN のホームアドレスを通知する．CN がパケットを返信する際には，通知されたホームアドレスをもとに IMS から MN のホームアドレスとモバイルアドレスの対応を取得する．MN が CN と通信中に別のネットワークに移動した際には IMS にモバイルアドレスの更新を通知する一方，CN に対して IMS から MN のマッピング情報を再取得するように通知する．

MAT では，ホームアドレスとモバイルアドレスはともに通常の IP アドレス体系を使用することができ，原理的に IPv4 と IPv6 のどちらにも適応することが可能である．

このように，MAT では LIN6 の考えをもとにしているが，アドレス変換を行うことで LIN6 の課題をいくつか解決している．しかし，マッピング情報を保持する特殊な装置が必要である点は同様である．また，DNS に独自のレコードを追加するため，MAT 非対応のノードは，MN のモバイルアドレスを知ることができない．そのため MN がホームネットワーク上にいないと通信を開始できず，移動ノード到達性に制約がある．

3. Mobile PPC の提案

3.1 位置づけと概要

従来技術では通信継続性を実現する場合においても位置管理装置（HA，MA，IMS）に IP アドレスを照会するため，いずれも移動ノード到達性と通信継続性の機能を 1 種類の位置管理装置で実現しようと試みている．提案方式では両者の機能を明確に分離する．移動ノード到達性の実現には，ホスト名と IP アドレスの関係を動的に管理する DDNS を用いることができる．DDNS は DNS の延長技術であり，すでに実用になっている．MN は初期立ち上げ時や移動時に新たな IP アドレスを取得すると必ず DDNS にその情報を登録するため，移動ノード到達性を満たすことができる．以後の説明では，通信が開始されるときには DDNS により通信相手の IP アドレスを知っていることを前提とする．本論文で提案する Mobile PPC は通信継続性を実現するための技術であり，LIN6，MAT と同様にエンドツーエンド方式と位置づけられる．

Mobile PPC の機能は，移動情報の通知処理と IP アドレスの変換処理に分けられる．通知処理は，MN が別のネットワークへ移動した場合，移動先のネットワークで新しく取得した IP アドレスを CN に通知す

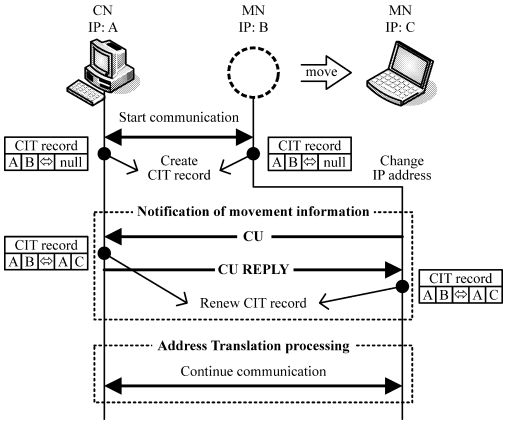


図 3 移動情報の通知
Fig. 3 Notification of movement information.

る．通知処理により，MN と CN は移動前と移動後の IP アドレスの対応関係を記すテーブル CIT（Connection ID Table）を更新する．CIT レコードは，通信が開始される際にコネクション単位で生成されるもので，MN が移動するたびにその内容が書き換えられる．IP アドレスの変換処理は，すべてのパケットに対して CIT を参照しながらアドレス変換を実行する．このような方式により，上位層に対しては移動による IP アドレスの変化が隠蔽され，上位層はアドレスの変化に気づくことなくコネクションを維持できる．

3.2 移動情報の通知処理

図 3 に Mobile PPC による移動情報の通知方法を示す．MN と CN 間で新しく通信が開始されると，エンド端末は送受信パケットをもとに上位層がコネクションを識別する際に用いられる通信識別子（両端末の IP アドレスとポート番号，プロトコル番号の組）が記された CIT レコードを生成する．CIT レコードは，移動前と移動後の通信識別子の情報から構成される．通信開始時点では IP 層でのアドレス変換は実行されないので移動後の情報を示すフィールドには null が入っている．

MN が CN と通信中に別のネットワークへ移動すると，MN は移動先で DHCP [15] サーバなどから新しく IP アドレスを取得する．ここで MN は，新しい IP アドレスと移動前に確立していたコネクションに関する通信識別子の情報を含む CU（CIT Update）パケットを生成し，CN へ送信する．CU は CN に対して移動を通知するとともに CIT の更新を要求する．CN は通知された情報をもとに自身の CIT を更新し，CIT の更新が完了したことを通知する CU REPLY パケットを返信する．MN は，CU REPLY を受信後に自身の保持する CIT を更新する．エンド端末で更新された CIT は，MN の移動前と移動後の IP アドレスの対応関係が登録され，以後の通信パケットに対する IP アドレス変換処理に用いられる．

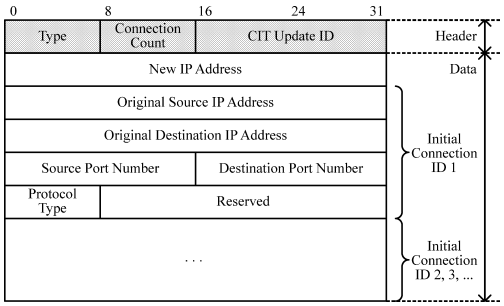


図 4 CU フォーマット
Fig. 4 CU packet format.

CU および CU REPLY は ICMP ECHO REQUEST をベースに定義されている．**図 4** に CU フォーマットを示す．CU と CU REPLY は共通のフォーマットである．ヘッダ部には CU/CU REPLY の識別（Type），CN と確立していたコネクション数（Connection Count），CU の識別番号（CIT Update ID），データ部には，新 IP アドレス（New IP Address）と初期通信識別子（Initial Connection ID）がコネクション数の分だけ含まれる．初期通信識別とはコネクションを確立した際の IP アドレスとポート番号の組である．

3.3 アドレス変換処理

図 5 に MN の IP アドレスが B から C へと変化した場合のアドレス変換処理を示す．CN から送信されるパケットの宛先 IP アドレスは，IP 層で CIT の情報を参照し移動後の IP アドレス C へ変換される．このパケットを受信した MN は，同様に CIT を参照しパケットの宛先 IP アドレスを移動前の IP アドレス B へ変換を行い上位層へ渡す．逆方向のパケットについても上記と同様なアドレス変換を行う．

このように IP 層において正しくルーティングされるようにアドレス変換し，上位層に対してはその変化を隠蔽するため通信中に MN が移動してもコネクションを維持させることが可能となる．

図 6 にアドレス変換の適用方法を示す．図 6 は MN と CN が通信中に MN が移動を繰り返し，IP アドレスが B から C，C から D へと変化した場合を表している．アドレス変換処理は，コネクション確立時点における MN の IP アドレスがベースとなる．移動を繰り返した場合には，通信開始時の IP アドレスと移動先で取得した IP アドレスとの間でアドレス変換を行う．図 6 において MN が 2 回移動した状態では，通

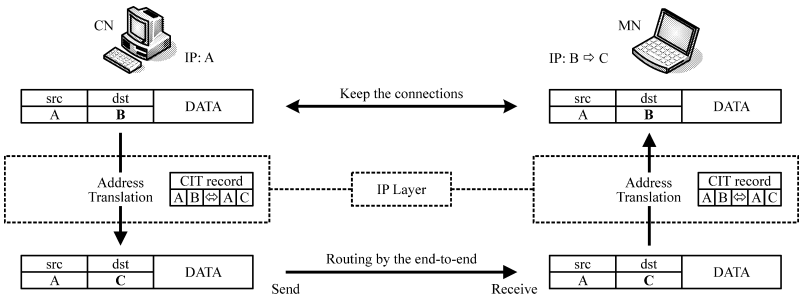


図 5 アドレス変換処理
Fig. 5 Address translation process.

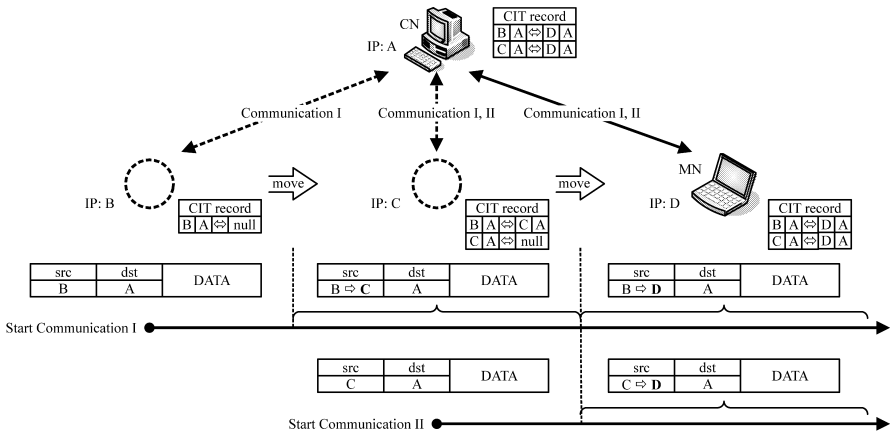


図 6 アドレス変換の適用方法
Fig. 6 Application method of address translation.

信 I の上位層では自端末 IP アドレスを B，通信 II の上位層では自端末 IP アドレスを C として認識する．このように Mobile PPC では，通信ごとに上位層で認識する自分の IP アドレスが異なる場合がある．IP アドレスが D に移行した時点では，IP アドレス B，C は実際に MN のインタフェースに割り当てられた IP アドレスではなくなる．その結果，別の移動端末が IP アドレス B，C を取得する可能性がある．まれなケースとして，アドレス変換を適応する通信と，新しく開始する通信の上位層における通信識別子が一致する可能性が考えられる．この場合，Mobile PPC では新たな通信の通信識別子が通信中の通信識別子と一致することを検出すると，新たな通信に対してポート変換を適用する．この方法により通信識別子の一致を防止することができる．したがって，CIT 内の変換情報としては IP アドレスだけでなくポート番号も含まれている．

3.4 従来技術との比較

Mobile PPC と従来技術の比較を**表 1** に整理する．従来技術の課題については 2 章で記述済みであるため，本節では Mobile PPC の利点を中心に記述する．Mobile PPC は移動ノード到達性を実現するために第 3 の装置として DDNS を利用するが，これは通信継続性を実現するために必要となる特有の装置ではなく，既存環境への適用が容易である．ほかの従来方式でも DNS サーバは必須であり，その意味では，提案方式は余分な装置を極力排除している．このため特有の装置による一点障害の課題がなく，二重化などの措置も不要で管理が容易である．Mobile PPC はエンドツーエンド方式であるため，冗長な通信経路は発生せず，アドレス変換を行うだけなので，パケットサイズの冗長はなく高スループットが期待できる．ただし，通信継続性を実現するために CN への実装が必要となる．同じエンドツーエンド方式である LIN6 においては，MA をプロキシとして拡張することにより，既存端末に対する移動透過性を提供する方式[16] が検討されている．Mobile PPC は既存端末との通信継続性はサポートしていないが，既存端末との上位互換性があるため，移動しない限り通信は可能である．移動ノード到達性については実績のある DDNS により満たす

表 1 従来技術との比較
Table 1 Comparison with existing technologies.

	Mobile IP	Mobile IPv6	LIN6	MAT	提案方式
特有の装置の存在	×（HA）	×（HA）	×（MA）	×（IMS）	○（なし）
一点障害	×	△	△	△	○
通信経路の冗長	×	△	○	○	○
パケットサイズの冗長	△	△	○	○	○
CN への実装	○	○	○	△	△
移動ノード到達性	○	○	○	△	○
通信開始時のオーバヘッド	○	○	○	○	△
IPv4/IPv6 両者に対応	×	×	×	○*	○
アドレス制約	○	○	×	○	○

* IPv4 における実装は未完了

ことができる．ただし，MN への誤接続を回避するためにネームキャッシュの有効期限を短くする必要があり，従来技術と比較して通信開始時のオーバヘッドが若干大きくなると考えられる．DDNS に関する考察については 5.5 節で述べる．

Mobile PPC は原理的に IPv4 と IPv6 の両者に適用可能である．多くの従来技術は検討対象や実装が IPv6 ベースであるが，IPv6 が普及していない現状では IPv4 における実装が可能であることは大きな利点である．また IP アドレスは従来のアドレス体系をそのまま利用できるため，アドレスの制約はない．さらにエンド端末のみに実装すればよいこと，既存端末との上位互換性があるため，段階的な普及が期待できる．以上の比較結果より，提案方式は今後のユビキタス社会に最も適した方式と考えられる．

4. Mobile PPC の実装

Mobile PPC を FreeBSD 5.2.1 上に実装し動作を検証した．本章では Mobile PPC のモジュール構成と CIT のフォーマット詳細について記述する．

4.1 モジュール構成

Mobile PPC のモジュール構成を**図 7** に示す．パケット受信時には IP 入力関数である `ip_input` から，パケット送信時には IP 出力関数である `ip_output` から Mobile PPC モジュールを呼び出し，アドレス変換処理を終えたら差し戻す形をとっている．IP アドレス変更時には ARP 関数より Mobile PPC モジュールが呼ばれ，移動情報通知処理を行う．既存の処理にはいっさい変更を加えない．

Mobile PPC を実現するモジュールは CIT 操作モジュール，アドレス変換モジュール，移動管理モジュールの 3 つがある．

CIT 操作モジュールは，アドレス変換モジュールと移動通知モジュールから呼び出され，CIT レコードの検索・生成・更新を実行する．また CIT の状態を監視し，無通信状態にある CIT レコードを削除する．

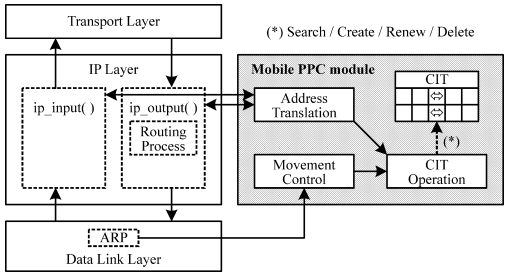


図 7 モジュール構成
Fig. 7 Module structure.

アドレス変換モジュールは，パケットの送信および受信時に呼び出される．入出力パケットの通信識別子をキーとして CIT 検索を行い，アドレス変換処理が必要であれば，CIT レコードの内容に従って，IP アドレスを変換し，それにともなうチェックサムの差分計算を行う．

移動管理モジュールは，IP アドレス変更時における CU および CU REPLY による移動情報通知処理を行う．MN がネットワークの移動を行った場合，移動先で DHCP による IP アドレスの取得を行う．DHCP サーバから IP アドレスの使用許可 DHCP ACK を受信した時点では，IP アドレスがまだ確定しておらず，その後に必ず Gratuitous ARP を用いた IP アドレスの重複チェックが行われる．そのため，移動管理モジュールは，上記 ARP の重複チェックタイムアウトと同時に呼び出される．

4.2 CIT

図 8 に CIT フォーマットを示す．CIT は，通信開始時の初期通信識別子，新アドレス情報，変換処理フラグ，無通信カウンタ（cnt），次テーブルアドレス（next）からなり，2,048 レコードから構成される．初期通信識別子は，3.2 節で述べたように通信が開始された際に登録される．新アドレス情報は，移動後の IP アドレスの組（tsIP/tdIP）とポート番号の組

Initial Connection Identifier　New IP Address Information

sIP	dIP	sport	dport	proto	tsIP	tdIP	tsport	dsport	trans	cnt	*next
MN1	CN	s1	d1	TCP	MN2	CN	s1	d1	ON	n	−
MN1	CN	s2	d2	UDP	MN2	CN	s2	d2	OFF	n	NAD

図 8 CIT フォーマット
Fig. 8 CIT format.

（tsport/tdport）からなり，CU および CU REPLY による通知処理によって登録されるフィールドである．通常はポート変換を行わないが，3.3 節で示したように MN の移動前のアドレスがほかのノードに割り当てられた場合，上位層で認識する初期通信識別子がつねにユニークになるように，必要な場合に限りポート変換を適用する．

変換処理フラグ（trans）は，該当する通信パケットに対してアドレス変換が必要かどうかを示す．通信開始時は OFF でありアドレス変換を行わない．MN が移動して，新アドレス情報が登録されたときに ON となり，以後の通信パケットにアドレス変換が適応される．無通信カウンタ（cnt）は，該当する CIT レコードを削除するためのフィールドである．この値は，スケジューラにより定期的にデクリメントされるが，テーブルが検索されるごとに初期値（n）に戻される．値が 0 になると該当するコネクションにより通信が行われていないと判断され，該当レコードは削除される．CIT はハッシュテーブルとして実装し，検索キーは送受信パケットの通信識別子のハッシュ値である．テーブルアドレス（next）は，ハッシュ値が衝突した場合に次のリンク先のテーブルアドレス（NAD）を示す．1 つのコネクションに対して送信用と受信用の 2 つの CIT レコードが生成される．CIT が更新された場合には，旧レコードは削除される．

5. Mobile PPC の性能測定と評価

Mobile PPC を試作し，両エンド端末が移動を繰り返しても通信を継続できることを確認した．本章では，試作システムの性能測定結果，および同一条件下における Mobile IP とのスループット比較を行った．

5.1 パケット処理時間

表 2 に Mobile PPC モジュールのパケット処理時間の測定結果を示す．ここで Mobile PPC の処理時間とは，`ip_input/ip_output` から Mobile PPC モジュールが呼び出され，Mobile PPC による処理が行われた後，差し戻すまでの時間である．これは Mobile PPC を実装することにより，すべてのパケットに対して CIT 検索が行われるため，これらにかかるオーバヘッドを調査するものである．Pentium（1.8 GHz）の CPU を搭載し，100BASE-TX で接続された PC 上で RDTSC（Read Time Stamp Counter）[17] を用いて測定した．RDTSC は CPU のカウンタから周波数クロックを取得する命令で，モジュール処理に費やした時間を正確に算出することができる☆．測定結果は FTP の通信中に流れた 1,500 バイトの通信パケット 1,000 個の処理時間の平均である．測定結果は，アドレス変換を行わない場合は 0.31 μ 秒☆☆，アドレス変換を行う場合は 0.54 μ 秒であった．1 パケットにかかる IP 層全体の処理時間は約 21 μ 秒であり，Mobile PPC モジュール処理時間の占める比率はアドレス変換を行わない場合は 1.47%，アドレス変換を行う場合は 2.53%であった．このことから Mobile PPC を実装したことによるオーバヘッドの増加は十分小さいといえる．

表 2 Mobile PPC モジュールのパケット処理時間
Table 2 Packet processing time of Mobile PPC module.

アドレス変換の有無	変換なし	変換あり
IP 層全体の処理時間	21.03	21.26
Mobile PPC モジュールの処理時間	0.31	0.54
（Mobile PPC モジュールの比率）	(1.47%)	(2.53%)

単位：μ 秒

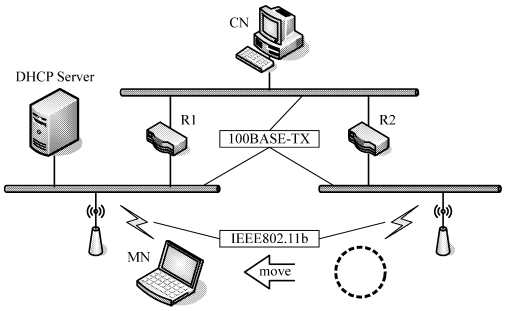


図 9 測定環境
Fig. 9 Measurement system.

5.2 移動時間の測定

Mobile PPC の移動透過性にかかわる処理時間を**図 9** に示す測定環境で測定した．2 つのルータ R1，R2 によりサブネットが異なる 3 つのネットワークを用意し，MN の移動先となるネットワークには DHCP サーバを設置した．**表 3** に装置仕様を示す．MN と CN に Mobile PPC を実装し，DHCP サーバおよび

☆ 測定に用いた機器の CPU 周波数が 1.8 GHz のため，分解能は 1/1.8 GHz = 0.56 ns となる．

☆☆ 文献 18) において類似の処理が RDTSC により性能測定されており，約 0.27 μ 秒という測定結果が示されている．本論文での測定値も妥当な結果であると考えられる．

表 3 装置仕様
Table 3 Device specifications.

	MN / CN / R1 / R2
CPU	Pentium4 3.0 GHz
Memory	512 MB
NIC	100BASE-TX
OS	FreeBSD 5.2.1-RELEASE

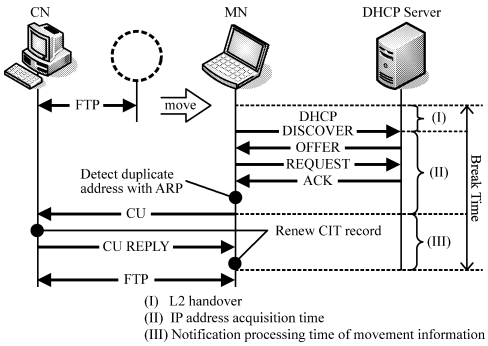


図 10 MN 移動時のシーケンス
Fig. 10 Sequence when a MN moves.

表 4 DHCP サーバからの IP アドレス取得時間
Table 4 IP address acquisition time with DHCP.

	最大	最小	平均
アドレス取得時間	4.85	2.99	3.34

単位：秒

クライアントには，ISC DHCP v2 パッケージ[19] を使用し，パラメータはプロトコルデフォルト値を使用した．有線 LAN は 100BASE-TX で構成し，MN は IEEE802.11b で接続した．MN から CN へ連続的に FTP を用いたデータ転送を実行させておき，MN を別のネットワークに移動させ，MN 側で直接コマンドに入力することにより，DHCP サーバから新しく IP アドレスを取得させた．

図 10 に MN が異なるネットワークへ移動した際に発生するシーケンスを示す．通信を再開するまでの通信中断時間は L2 ハンドオーバ時間，MN が DHCP サーバからの IP アドレス取得時間とエンド端末間で行われる移動情報通知処理時間の合計である．IP アドレス取得時間については，本提案方式の主題ではないが参考のために測定を行った．

表 4 に IP アドレス取得時間の測定結果を示す．この時間には MN と DHCP サーバ間の 2 往復の DHCP シーケンスと IP アドレス取得後に行われる ARP による重複アドレスチェックが含まれる．表 4 に示すように約 2〜5 秒（平均 3.34 秒）の時間を要し，通信中断時間のほとんどの割合を占める．

表 5 に移動情報通知処理時間の測定結果を示す．移

表 5 移動情報の通知処理時間
Table 5 Notification processing time of movement information.

	エンド端末のコネクション数				
	1	2	3	4	5
MN/CN の CIT 更新時間	38	40	44	45	47
CU/CU Reply 到達時間	288	258	253	267	326
移動情報通知処理時間	326	298	293	312	373

単位：μ 秒

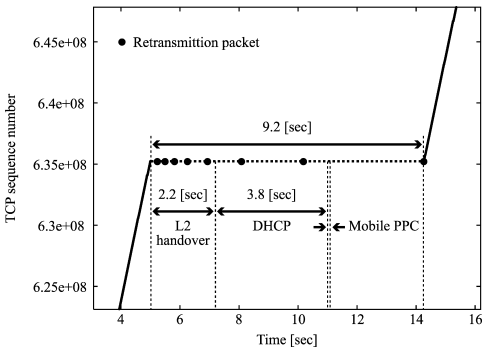


図 11 MN 移動時における TCP シーケンス番号の変化
Fig. 11 Changes of TCP sequence number when a MN moves.

動情報通知処理時間には，MN と CN の CIT 更新時間，CU および CU REPLY の伝達時間が含まれる．エンド端末間で行われているコネクション数を 1 から 5 に増やした場合，MN と CN の CIT 更新時間は 38〜47 μ 秒となった．また，CU および CU REPLY の到達時間は 253〜326 μ 秒となった．パケット到達時間に多少ゆらぎがあるのは，測定環境に無線 LAN があり，周囲の環境による影響を受けたためだと考えられる．このことから，移動情報通知処理時間は，パケットの伝達時間が大半をしめており，エンド端末間のコネクション数による影響はほとんどないといえる．

図 11 に，図 10 と同様の条件において MN が移動したときに TCP シーケンス番号が変化する様子を示す．無線は周囲の条件に依存するため，評価システム内はすべて 100BASE-TX の有線 LAN とした．ネットワークの移動は MN の LAN ケーブルを移動先ネットワークにつなぎ直し，その後ただちに MN から IP アドレスを取得するコマンドを実行することによりエミュレートした．上記時間は MN が物理的にネットワークから切り離された状態であり，L2 ハンドオーバの時間と見なすことができる．測定の結果，L2 ハンドオーバに約 2.2 秒，DHCP による IP アドレス取得に 3.8 秒を要した．表 5 から分かるように Mobile PPC による移動情報通知処理時間は合計 300 μ 秒程度であり，ほとんど無視できる．このように通信を継

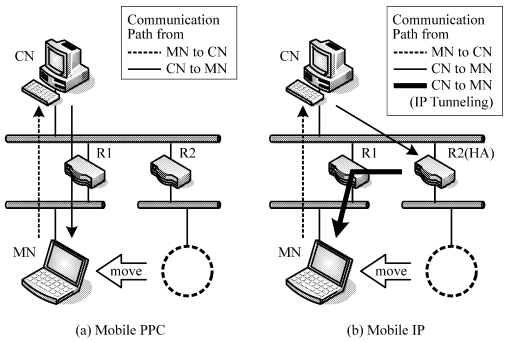


図 12 評価システムの構成
Fig. 12 Structure of evaluation systems.

続するために必要な処理は約 6 秒☆で完了したが，実際の通信が再開されるまでには約 9.2 秒を要した．これは TCP 再送制御が機能したためであり，図 11 から分かるように，8 回目の再送パケットにより通信が再開されたためである．実際には無線 LAN の L2 ハンドオーバは 50〜400 ミリ秒で完了するため[20]，通信中断時間を減少するには IP アドレス取得時間を短縮することが重要であることが分かる．本件の解決策としては，DHCP ソフトウェアの最適化によるシームレスハンドオーバ[21] や重複アドレスチェックの高速化[22] などが有効と考えられる．

5.3 Mobile IP とのスループット比較

Mobile PPC と Mobile IP のスループット比較を行うために**図 12** のような評価システムを構築した．Mobile PPC 環境（図 12 (a)）では MN，CN に Mobile PPC を実装し，Mobile IP 環境（図 12 (b)）では，MN に Mobile IPv4 を実装し，R2 に HA の機能を実装した．Mobile IP には PSU Mobile-IP パッケージ[23] を使用し，動作モードは Mobile PPC との比較をしやすくするため FA が不要な co-located care-of address モードとした．使用した装置仕様は表 3 に示したものと同一で，ネットワークの移動は LAN ケーブルをつなぎ直すことでエミュレートした．図 12 中の矢印は，点線が MN から CN への通信経路，実線が CN から MN への通信経路を示している．

上記環境下で，以下のケースにおける MN-CN 間のスループットを測定した．

case1：Mobile PPC を実装しない状態

case2：MN と CN が Mobile PPC を実装しているがアドレス変換をする前（移動前）

case3：Mobile PPC を実装し，かつアドレス変換をしているとき（移動後）

☆ $2.2\,\mathrm{sec} + 3.8\,\mathrm{sec} + 300\,\mu\mathrm{s} \fallingdotseq 6\,\mathrm{sec}$

表 6 スループットの比較
Table 6 Comparison of throughput.

状態		スループット	低下率 [%]
General	case1	93.237	—
Mobile PPC	case2	93.236	0.001
	case3	93.193	0.047
Moible IP	case4	93.231	0.006
	case5	85.202	8.618

スループット単位：Mbps

case4：MN が Mobile IP を実装し HA 配下にいるとき（移動前）

case5：Mobile IP を実装して IP トンネリング通信をしているとき（移動後）

表 6 に測定結果を示す．スループットの測定には，ネットワークベンチマークソフト Netperf[24] を使用し，20 回の平均値とした．表 6 より分かるように Mobile PPC では，何も実装していない状態（case1）に比べ移動前（case2）と移動後（case3）ともにスループットの低下はほとんど見られなかった．

Mobile IP は，移動前（case4）ではスループットの低下はほんどないが，移動後（case5）では，IP カプセル化によるオーバヘッドと通信経路の冗長により，スループットが 8.6%ほど低下した．図 12 (b) の測定構成では，HA と MN が 1 ホップ分離れている構成であるが，一般的なネットワーク構成では HA を経由することによりラウンドトリップ遅延がさらに増加する可能性があり，スループットがさらに低下することが予想される．

5.4 セキュリティの考察

エンドツーエンド方式で移動透過性を実現する場合，通信継続の際には悪意あるユーザによるなりすましを防止するため，端末間における確実な認証が必要である．グローバルな環境では，通信相手は不特定多数となるため，事前に認証に必要な共有鍵や証明書を共有することは難しい．Mobile IPv6 では，MN と HA が共有鍵を事前に保持していることを前提とし，CN が HA と MN に宛てて送信した両方のデータを MN が正しく受信することによって共有鍵を生成する仕組みが提案されている．このような方式は，特定の装置を使用しない Mobile PPC には適していない．そこで，Mobile PPC では認証機能を実現するために通信開始時に Diffie-Hellman 鍵交換[25] を利用した共有鍵の生成および移動時のなりすましを防止するための認証機構[26] を適用する．**図 13** に Mobile PPC における認証機構の原理を示す．通信開始に先立ち，MN と CN の間で DH 鍵交換を行い，共通鍵を共有する．この共通鍵を使用して，移動情報の通知時になりすまし防止

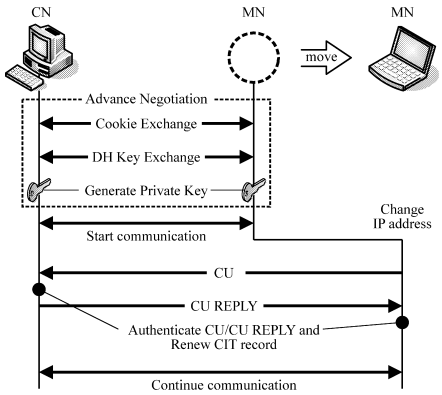


図 13 Mobile PPC における認証方式
Fig. 13 Authentication method on Mobile PPC.

のための認証を行う．DH 鍵交換は動作が重いため，直前にクッキー交換を行い，DoS 攻撃を防止する．本方式を試作した結果，通信開始時のネゴシエーションに 2.92 秒を要し，このうち 2.89 秒が DH 演算を占めていた．今後は DH 演算時間の低減が検討課題になると考えられる．また，移動情報の通知処理は認証処理を含めても 0.33 ミリ秒であり，表 5 に示す認証処理のない場合と比べほぼ同等の時間で処理が完了している．

Mobile PPC で暗号化通信を行う場合，IPsec[27] を適用することが可能である．IPsec では TCP/UDP チェックサムフィールドが暗号化，完全性保証の範囲に含まれているため，IP アドレスが変換されると偽造パケットと見なされ破棄されてしまう問題がある．しかし，Mobile PPC のアドレス変換処理は IP 層の上位部分で実行されており，IPsec 処理が Mobile PPC 処理に影響を受けることはない．ただし，IPsec 適用時には通信継続時に新たな IP アドレスに対する SA（Security Association）を再生成する必要があり，IKE（Internet Key Exchange）[28] が実行されるため通信中断時間が増加する．文献 29) によると，IKE のネゴシエーションには約 1 秒を要しており，Mobile PPC による移動情報通知処理の前に実行される．

なお本論文では切り離して記述しているが，Mobile PPC はもともと GSCIP（Grouping for Secure Communication for IP）[30] と呼ぶアーキテクチャの枠組みの中で移動透過性を実現する手段として考案されたものである．GSCIP においては，あらかじめ同一の通信グループに対して同一のグループ暗号鍵を割り当てる．このような環境下では通信開始時に DH 鍵交換を行う必要はなく，移動時にグループ暗号鍵を用いた認証を容易に実現することができる．また GSCIP では暗号化通信に PCCOM（Practical Cipher Communication）[18] を適用することを想定している．PCCOM ではパケット長を変えないまま認証を含む暗号化通信が可能であり，高スループットを維持することが可能である．また PCCOM の暗号化処理に必要な動作処理情報は，GSCIP の主要プロトコルである動的処理解決プロトコル DPRP（Dynamic Process Resolution Protocol）[29] によって通信開始時に高速に生成される．

5.5 既存環境への対応

Mobile PPC は上位層に対してアドレスの変化を隠蔽するため，上位層で IP アドレスを直接意識する SIP（Session Initiation Protocol）[31] や FTP のようなプロトコルでも問題なく動作する．また，Mobile PPC は IP 層にモジュールを実装しているが，上位層とのインタフェースにはいっさい影響を与えておらず，SCTP（Stream Control Transmission Protocol）[32] や DCCP（Datagram Congestion Control Protocol）[33] など IP 層の上位で新規に定義されたプロトコルに対しても適用することができる．

Mobile PPC 適用システムにおいて，MN が NAT 配下に移動することも想定される．Mobile IP では MN と HA の間に UDP トンネリングを形成することにより，NAT 越えに対応した移動透過性を実現している[34]．Mobile PPC も同様の原理で対応することができる．Mobile PPC では MN と CN の間にトンネルを形成する．送信側はアドレス変換処理後のパケットを UDP でカプセル化して通信相手へ送信することにより，NAT 変換処理の影響を受けることなく，Mobile PPC のアドレス変換処理を実行することができる．

本提案では移動ノード到達性を実現するために既存の DDNS を利用しているが，MN への誤接続を回避するためにネームキャッシュの有効期限を短く設定する必要がある．このため MN の IP アドレス取得時に発生するオーバヘッドの多くは，ネームキャッシュがない場合の問合せ時間となる．文献 12) によると，DNS への問合せ時間の平均は，キャッシュがある場合で約 15 ミリ秒，キャッシュがない場合で約 350 ミリ秒である．この通信開始時のオーバヘッドが実用上の問題となるか今後検討を行う必要がある．また DNS へ登録されたレコードは明示的に削除しない限り残り続けることになるため，レコードに登録した IP アドレスが別の端末に再利用された場合にも誤接続の可能性がある．この解決策として，IP アドレスを割り当てる DHCP サーバと DDNS が協調してネットワークから離脱した端末のレコードを削除するなどの方法

が考えられる．

5.6 残された課題

一般に無線環境でネットワークの移動を行った場合，無線レイヤと IP 層が独立してハンドオーバを実行するためパケットロスが避けられない．また，MN どうしの通信では，両者がまったく同時に移動した場合に CU が通信相手に到達せず，MN は互いに移動したことを知ることができないという可能性が考えられる．これらの課題はエンドツーエンド方式共通の課題である．これらの解決策として，MN が一時的に新旧 2 つの IP アドレスを保持させたり，無線レイヤと Mobile PPC が連携するなどの工夫が今後必要になると考えられる．文献 35) では無線 LAN カードを 2 枚搭載し，旧アドレス宛のパケットも一定の時間内であれば受信できるようにすることによって，この課題の解決を試みている．

6. ま と め

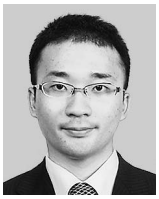
本論文では，エンド端末間で移動透過性を実現する Mobile PPC について提案した．Mobile PPC は，エンド端末の IP 層にアドレス変換処理機能を導入する．MN の IP アドレスが変化したとき，MN から CN に変化情報を報告し，アドレス変換テーブルを更新する．移動後の通信パケットは上記アドレス変換テーブルに基づき IP 層でアドレス変換する．この方式により，IP アドレスの変化は上位ソフトウェアから隠蔽される．IP アドレスの変換は，IP 層で行われるため，上位ソフトウェアを変更する必要がなく，IP アドレスを単に変換する方式であるためパケット長が変化することがない．特殊な管理サーバが不要であり，既存システムとの上位互換性を有することから，従来技術の方式に比べ段階的な普及が期待できる．IPv4 において Mobile PPC を FreeBSD 上に実装し，動作の確認と性能測定を行った．その結果，IP アドレス変換による性能の低下がほとんどないことが分かった．さらに，Mobile IP とスループットの比較を行い，Mobile PPC の処理が通信に与える影響は，Mobile IP に比べて小さいことを示した．移動の際には Mobile PPC 自体のオーバヘッドは少ないものの，アドレス取得によるオーバヘッドを減らす工夫が別途必要であることが分かった．今後は，高速ハンドオーバを検討していくとともに，IPv6 についても本手法の適用を検討する．

参 考 文 献

1) 寺岡文男：インターネットにおけるノード移動透過性プロトコル，電子情報通信学会論文誌，Vol.J87-D1, No.3, pp.308–328 (2004).
2) Perkins, C.: IP Mobility Support for IPv4, RFC 3344, IETF (2002).
3) Perkins, C.: IP Encapsulation within IP, RFC 2003, IETF (1996).
4) Calhoun, P. and Perkins, C.: Mobile IP Network Address Identifier Extension, RFC 2794, IETF (2000).
5) Perkins, C. and Calhoun, P.: Mobile IP Challenge/Response Extensions, RFC 3012, IETF (2000).
6) Montenegro, G.: Reverse Tunneling for Mobile IP, RFC 3024, IETF (2001).
7) Bhagwat, P., Maltz, D. and Segall, A.: MSOCKS+: an architecture for transport layer mobility, *Computer Networks*, Vol.39, No.4, pp.385–403 (2002).
8) Funato, D., Yasuda, K. and Tokuda, H.: TCP-R: TCP Mobility Support for Continuous Operation, *Proc. IEEE International Conference on Network Protocols* (*ICNP*), Atlanta, Georgia, pp.229–236 (1997).
9) Snoeren, A. and Balakrishnan, H.: An End-to-End Approach to Host Mobility, *Proc. 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking* (*MobiCom*), Boston, Massachusetts, ACM, pp.155–166 (2000).
10) 松岡保静，吉村 健，大矢智之：エンドツーエンド型 IP ソフトハンドオーバ，電子情報通信学会論文誌，Vol.J86-B, No.8, pp.1369–1378 (2003).
11) Ishiyama, M., Kunishi, M., Uehara, K., Esaki, H. and Teraoka, F.: LINA: A New Approach to Mobility Support in Wide Area Networks, *IEICE Trans. Comm.*, Vol.E84-B, No.8, pp.2076–2086 (2001).
12) 相原玲二，藤田貫大，前田香織，野村嘉洋：アドレス変換方式による移動透過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 (2002).
13) Johnson, D., Perkins, C. and Arkko, J.: Mobility Support in IPv6, RFC 3775, IETF (2004).
14) Vixie, P., Thomson, S., Rekhter, Y. and Bound, J.: Dynamic Updates in the Domain Name System (DNS UPDATE), RFC 2136, IETF (1997).
15) Droms, R.: Dynamic Host Configuration Protocol, RFC 2131, IETF (1997).
16) 石山政浩，國司光宣，河野通宗，寺岡文男：移動体通信プロトコル LIN6 における後方互換性拡張の一方式，電子情報通信学会技術研究報告，Vol.102, No.362, pp.23–28 (2002).
17) Intel Corp.: *Using the RDTSC Instruction for Performance Monitoring* (1998).

http://developer.intel.com/drg/pentiumII/appnotes/RDTSCPM1.htm
18) 増田真也, 鈴木秀和, 岡崎直宣, 渡邊 晃：NATやファイアウォールと共存できる暗号通信方式PCCOMの提案と実装, 情報処理学会論文誌, Vol.47, No.7, pp.2258–2266 (2006).
19) Internet Systems Consortium: ISC DHCP Version 2. http://www.isc.org/index.pl?/sw/dhcp/
20) Mishra, A., Shin, M. and Srbaugh, W.: An Empirical Analysis of the IEEE802.11 MAC Layer Handoff Process, *ACM SIGCOM Computer Communication Review*, Vol.33, No.2, pp.93–102 (2003).
21) 小川猛志, 伊東 匡：DHCP をベースとしたシームレスハンドオーバ方法の研究, 電子情報通信学会論文誌, Vol.J88-B, No.11, pp.2228–2238 (2005).
22) Moore, N. and Daley, G.: Fast Address Configuration Strategies for the Next-Generation Internet, *Proc. the Australian Telecommunications, Networks, and Applications Conference* (*ATNAC*), Melbourne, Australia (2003).
23) The Portland State University Secure Mobile Networking Project: PSU Mobile-IP. http://www.cs.pdx.edu/research/SMN/
24) Jones, R.: Netperf: a network performance monitoring tool. http://www.netperf.org/netperf/NetperfPage.html
25) Diffie, W. and Hellman, M.: New Directions in Cryptography, *IEEE Trans. Inf. Theory*, Vol.IT-22, No.6, pp.644–654 (1976).
26) 瀬下正樹, 渡邊 晃：Mobile PPC における認証方式の実装, マルチメディア, 分散, 強調とモバイル（DICOMO2006）シンポジウム論文集（II), Vol.2006, No.6, pp.809–812 (2006).
27) Kent, S. and Atkinson, R.: Security Architecture for the Internet Protocol, RFC 2401, IETF (1998).
28) Harkins, D. and Carrel, D.: The Internet Key Exchange (IKE), RFC 2409, IETF (1998).
29) 鈴木秀和, 渡邊 晃：フレキシブルプライベートネットワークにおける動的処理解決プロトコル DPRP の実装と評価, 情報処理学会論文誌, Vol.47, No.11, pp.2976–2991 (2006).
30) 鈴木秀和, 竹内元規, 加藤尚樹, 増田真也, 渡邊晃：フレキシブルプライベートネットワークを実現するセキュア通信アーキテクチャGSCIP の提案, マルチメディア, 分散, 強調とモバイル（DICOMO2005）シンポジウム論文集, Vol.2005, No.6, pp.441–444 (2005).
31) Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and Schooler, E.: SIP: Session Initiation Protocol, RFC 3261, IETF (2002).
32) Stewart, R., Xie, Q., Morneault, K., Sharp, C., Schwarzbauer, H., Taylor, T., Rytina, I., Kalla, M., Zhang, L. and Paxson, V.: Stream Control Transmission Protocol, RFC 2960, IETF (2000).
33) Kohler, E., Handley, M. and Floyd, S.: Datagram Congestion Control Protocol (DCCP), RFC 4340, IETF (2006).
34) Levkowetz, H. and Vaarala, S.: Mobile IP Traversal of Network Address Translation (NAT) Devices, RFC 3519, IETF (2003).
35) 金本綾子, 瀬下正樹, 竹内元規, 渡邊 晃：Mobile PPC におけるパケットロスなしハンドオーバの提案, マルチメディア, 分散, 強調とモバイル（DICOMO2006）シンポジウム論文集（II), Vol.2006, No.6, pp.817–820 (2006).

(平成 18 年 3 月 20 日受付)
(平成 18 年 10 月 3 日採録)



竹内 元規（正会員）

2004 年名城大学理工学部情報科学科卒業．2006 年同大学大学院理工学研究科情報科学専攻修了．同年日本コムシス株式会社入社．IT ビジネス事業本部に所属．修士（工学）．

鈴木 秀和（学生会員）

2004 年名城大学理工学部情報科学科卒業．2006 年同大学大学院理工学研究科情報科学専攻修了．現在，同大学院理工学研究科電気電子・情報・材料工学専攻博士後期課程に在学中．ネットワークセキュリティ，モバイルネットワーク等の研究に従事．修士（工学）．2006 年 IEEE 名古屋支部学生奨励賞受賞．2006 年 DICOMO2006 松下賞（最優秀プレゼンテーション賞）受賞．電子情報通信学会所属．

渡邊　晃（正会員）

1974 年慶應義塾大学工学部電気工学科卒業．1976 年同大学大学院工学研究科修士課程修了．同年三菱電機株式会社入社後，LAN システムの開発・設計に従事．1991 年同社情報技術総合研究所に移籍し，ルータ，ネットワークセキュリティ等の研究に従事．2002 年名城大学理工学部教授，現在に至る．博士（工学）．電子情報通信学会，IEEE 各会員．