Vol.2009-MBL-51 No.3
Vol.2009-ITS-39 No.3
2009/11/5

移動透過性を考慮したNAT越え通信の提案

水谷 智大[†1] 鈴木 秀和[†1,†2] 渡邊 晃[†1]

IPv4においてエンドノードのみで移動透過性を実現する有効な技術として，Mobile PPCがある．しかし，通信ノードがグローバルアドレス空間とプライベートアドレス空間を跨って移動する場合の方法については，一部の移動パターンを除いて充分な検討がなされていない．本論文ではNATが介在するあらゆる移動パターンにおいて，Mobile PPCによる移動透過性を実現できる方式について検討した．

Proposal of NAT Traversal Technology Considering Mobility

Tomohiro Mizutani,[†1] Hidekazu Suzuki[†1,†2] and Akira Watanabe[†1]

Mobile PPC is the useful technology that can realize mobility with only end nodes in IPv4 network. However, except some cases, it is not studied enough when communication nodes move between global address area and private address area. In this paper, we studied the method that can realize mobility with Mobile PPC in every cases NATs exist in communication paths.

1. はじめに

公衆無線網の普及により屋外でネットワークに接続する機会が増加している．またWi-Fiを搭載した小型携帯端末も増加しており，移動しながらIPネットワークを利用したいという需要がある．ところがIPネットワークでは，ノード識別情報となるIPアドレスに位置情報が含まれているため，ノードが通信中に場所を移動するとIPアドレスが変化し，通信が切断される．この課題を解決する技術は移動透過性[1]と呼ばれ，これまでに様々な研究が行われている[2)–6)]．移動透過性の研究は将来のIPv6への移行を想定し，IPv6を前提とした方式が主流となっている．しかしIPv6はIPv4との互換性がないことから，普及が思うように進んでいない．そのためIPネットワークは今後IPv6への移行が進んだとしても，長期に渡ってIPv4と混在する環境が続くと考えられ，IPv4における移動透過性技術も重要である．

IPv4ではインターネットと組織のネットワークの境界にNAT（Network Address Translation）を設置する通信形態が一般的である．しかしこのようなネットワークでは，NATの外側のノードEN（External Node）がNATの内側のノードIN（Internal Node）に対して通信を開始することができない．これはNAT越え問題と呼ばれ，IPv4の汎用性を損なう要因となっている．IPv4において移動透過性を実現しようとした場合，NAT越え問題を解決することが重要な課題である．

IPv4における移動透過性技術としてMobile IP[2)]やMobile PPC（Mobile Peer to Peer Communication）[6)]がある．Mobile IPはインターネット上のサーバアクセスを前提とした方式であり，サーバ側が移動することは想定していない．またHA（Home Agent）と呼ばれる第三の装置を経由した通信を行うため，HAの障害に対して弱い点やスループットの低下などが指摘されている．また，MN（Mobile Node）からCN（Correspondent Node）への通信パケットの宛先がHAのアドレスであるため，途中にNetwork Ingress Filtering[7)]が適用されたルータが存在するとパケットが破棄されるなどの問題がある．Mobile PPCではこれらの課題が解決されており，両エンドノードが移動可能である．また第三の装置を必要とせず，常にエンドエンド通信を行うことができる．Mobile PPCではNAT越えを考慮した通信についても検討されているが[8),9)]，すべての移動パターンに対応できるわけではなく，十分とは言えない．

既存のNAT越え技術を大きく分類すると，NATを改造する方法[10),11)]と，NATに改造を加えずにエンドノードの改造や第三の装置を利用する方法[12),13)]がある．本稿では，あらゆる通信環境に適用できることから，NATに改造を加えない方式に着目する．NATは全てのパケットが集約する場所に設置されるため，セキュリティ上有効な手段となり得る．事実，組織内のネットワークトポロジがNATにより隠蔽されるという利点がある．また近年のNATにはSPI（Stateful Packet Inspection）が適用されることが多い．SPIではセッション毎にTCPのシーケンス番号などの通信のログを保持して，整合性がないパケットを

†1 名城大学大学院理工学研究科
Graduate School of Science and Technology, Meijo University
†2 日本学術振興会特別研究員 PD
Research Fellow of the Japan Society for the Promotion of Science

Vol.2009-MBL-51 No.3
Vol.2009-ITS-39 No.3
2009/11/5

破棄する．SPI までを考慮すると，NAT を改造しないまま TCP の NAT 越えを実現することは極めて難しい状況になっている．

SPI を考慮しても NAT 越えを実現できる既存技術として TURN（Traversal Using Relay NAT）[13] がある．TURN は事前に NAT の内側のノードが TURN サーバにセッションを確立しておき，全てのパケットを TURN サーバを経由させることにより NAT 越えを実現することができる．しかし TURN は通信開始時にのみ使用される技術であり，移動透過性についての考慮は全くなされていない．更に，TURN の機能はアプリケーション毎に実装する必要があるという課題がある．

本研究では Mobile PPC と，TURN を参考にした NAT 越え技術を組み合わせることにより，通信経路上に NAT を含むあらゆる移動パターンに対応できる移動透過性の実現方式を提案する．提案方式では MN が移動すると中継ノードとの間で一時的に通信経路を確立し，この通信経路を利用して Mobile PPC を動作させる．その後，エンドエンド通信が可能と判断した場合はエンドエンド通信へと移行する．また提案方式を Mobile PPC の拡張としてカーネルに実装することにより，アプリケーションに依存しない方式とすることができる．

以降，第 2 章では既存技術として Mobile PPC と TURN の概要とその課題について述べる．次に第 3 章で本提案方式について述べ，最後に第 4 章でまとめる．

2. 既存技術の概要及びその課題

本章では既存技術として Mobile PPC と TURN を紹介し，その課題を述べる．本章以降で用いる記号を以下に定義する．

Gn $(n = 1, 2, 3 \ldots)$ グローバルアドレス

Pn プライベートアドレス

$A : a$ IP アドレス A，ポート番号 a

$A : a \rightarrow B : b$ 送信元 $A : a$ から宛先 $B : b$ のパケット

$A : a \leftrightarrow B : b$ $A : a$ と $B : b$ 間の通信

$A : a \Leftrightarrow B : b$ $A : a$ と $B : b$ のアドレス／ポート変換

2.1 Mobile PPC

Mobile PPC は IPv4 において第三の装置の助けを借りることなくエンドエンドで移動透過性を実現するプロトコルである．エンドノードの IP 層で全ての通信パケットの IP アドレスを変換することにより上位層に対してアドレスの変化を隠蔽し，かつパケットを正しくルーティングすることにより通信を継続することができる．MN が移動した際の Mobile PPC の動作概要を図 1 に示す．

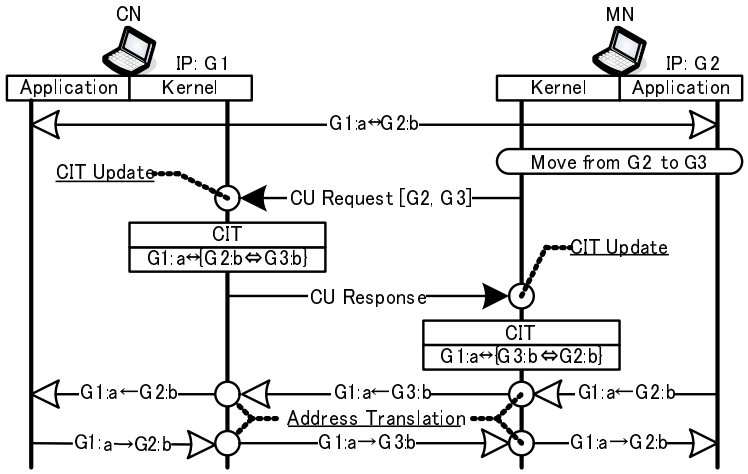

図 1 ノード移動時の Mobile PPC の動作概要
Fig. 1 Operation of Mobile PPC when MN moves

CN と MN は既に通信 $G1 : a \leftrightarrow G2 : b$ を開始しているものとする．通信の開始時の通信相手の IP アドレスの発見方法は Mobile PPC の定義範囲外であるが，DDNS（Dynamic DNS）などの既存技術を適用することで実現できる．

通信中に MN が移動して新しく IP アドレス $G3$ を取得すると，MN は CN との間で CU（CIT Update） Request/Response を交換して MN の新旧のアドレス $G2$，$G3$ の対応関係を記録する．CN と MN はカーネルの IP 層に以下の様な共通の CIT（Connection ID Table）と呼ぶテーブルを生成する．

$$G1 : a \leftrightarrow \{G2 : b \Leftrightarrow G3 : b\}$$

この後の通信では，両エンドノードは IP 層にて CIT を参照して，全てのパケットのアドレスを変換する．この処理により，両エンドノードのアプリケーションは IP アドレスが変化したことに気付くことなく通信を継続できる．このような動作は機能をカーネルに実装することにより初めて可能となる方式である．

ここで，通信経路上に NAT が存在する場合，CU Request 内の情報と IP ヘッダ内のアド

Vol.2009-MBL-51 No.3
Vol.2009-ITS-39 No.3
2009/11/5

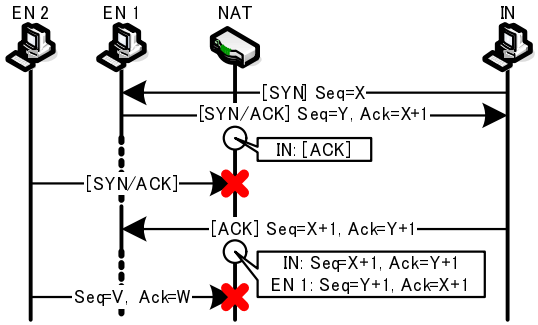

図 2 SPI によるフィルタリング処理の概要
Fig. 2 Processes of SPI filtering

レスが一致しなくなる．このままでは正しく動作することができない．9) では Mobile PPC を改造して NAT を含む一部の移動パターンでの実現が可能となっているが，NAT にも改造を加えることが前提であり，本論文の主旨とは異なる．8) では NAT を改造しないまま Mobile PPC を改造することにより一部の移動パターンおける移動透過性の方式が提案されている．しかし，NAT に SPI が搭載されていると TCP の NAT 越えは難しい．これらの研究は一部の移動パターンにのみ着目しており，それ以外の移動パターンには対応できない．

2.2 SPI と TURN

TURN は NAT に SPI 機能が実装されていても NAT 越えを実現できる技術である．以下，SPI の詳細について説明した後，TURN について説明する．

2.2.1 SPI

SPI は通信パケットを監視し，TCP ヘッダ内の SYN や ACK などのフラグやシーケンス番号を解析し，整合性がないパケットを破棄する．SPI によるフィルタリング処理の概要を図 2 に示す．

IN は EN1 と TCP セッションを確立するためにシーケンス番号初期値 X の SYN パケットを EN1 に送信する．EN1 はシーケンス番号初期値 Y，確認応答番号 $X+1$ の SYN/ACK パケットを応答する．NAT はこのセッションに関するログを保持し，次のパケットは IN からの ACK パケットのみを通過させる．そのため，EN2 から偽装したパケットを IN に送信しても，このパケットは破棄される．

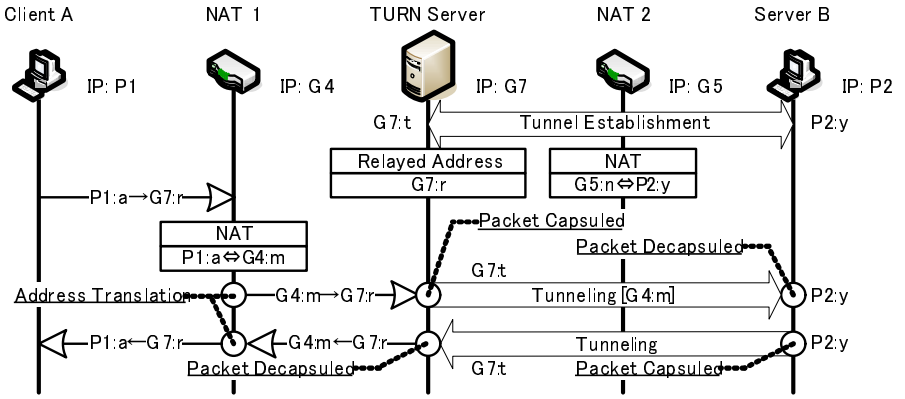

図 3 TURN の動作概要
Fig. 3 Operation of TURN

IN からシーケンス番号 $X+1$，確認応答番号 $Y+1$ の ACK パケットを EN1 に送信すると，EN1 との TCP セッションが確立する．NAT はこのセッションのログを保持し，このセッションにおいて IN からはシーケンス番号 $X+1$，確認応答番号 $Y+1$ のパケットのみを，EN1 からはシーケンス番号 $Y+1$，確認応答番号 $X+1$ のパケットのみを通過させる．従って上記以外の TCP パケットは一切受け付けられず，破棄される．

このように SPI はセキュリティ面で非常に強固なフィルタリング機能であり，NAT 越えを極めて困難にする要因となっている．

2.2.2 TURN

TURN の動作概要を図 3 に示す．最も動作が複雑な例として，サーバ B とクライアント A がそれぞれ異なるプライベートアドレス空間に存在し，インターネットを介して接続されているものとする．

TURN の実現にはインターネット上に TURN サーバが必要であり，TURN サーバはグローバル IP アドレス $G7$ を持つ．サーバ B は，グローバル IP アドレス $G5$ を持つ NAT 配下に存在し，IP アドレス $P2$ を取得している．クライアント A は，グローバル IP アドレス $G4$ を持つ NAT 配下に存在し，IP アドレス $P1$ を取得している．サーバ B は TURN を実行するために必要な機能を保持しており，また NAT1，NAT2 は共に SPI 機能を保持しているものとする．

Vol.2009-MBL-51 No.3
Vol.2009-ITS-39 No.3
2009/11/5

まず，サーバ B は TURN サーバとの間で予めネゴシエーションを行ってトンネルセッション $G7:t \leftrightarrow P2:y$ を確立し，リレードアドレスとして $G7:r$ の登録を行っておく．このとき，NAT には NAT テーブル $G5:n \Leftrightarrow P2:y$ が生成される．

クライアント A はサーバ B と通信を開始したい時，何らかの手段を用いてサーバ B のリレードアドレス $G7:r$ を取得する．次に最初の通信パケットを $G7:r$ 宛に送信すると，NAT には NAT テーブル $P1:a \Leftrightarrow G4:m$ が生成される．このパケットを受信した TURN サーバはパケットのメッセージ部に NAT1 のマッピングアドレス $G4:m$ の情報を記載し，トンネルセッション $G7:t \leftrightarrow P2:y$ を用いてサーバ B 宛にパケットを転送する．このパケットは NAT2 の NAT テーブルを通過してサーバ B に到達し，サーバ B はパケットからアプリケーションデータを取り出す．以上の処理により，クライアント A はサーバ B に通信を開始することができる．

クライアント A の TURN サーバへのアクセスや，サーバ B の TURN サーバとのトンネルセッションの確立は共に NAT の内側から確立されたセッションであるため，NAT に SPI が実装されていても影響を受けない．なお，クライアント A がリレードアドレス $G7:r$ を知る方法は TURN の規定外であり，実装方法はシステムにより異なる．

このように TURN は必ず TURN サーバを中継した通信となるが，SPI を実装する NAT を，改造しないまま NAT 越えを実現できる唯一のプロトコルである．しかし TURN はアプリケーションで動作するプロトコルであるため，アプリケーション毎に実装する必要がある．また，通信開始時に実行されることを想定しており，移動透過性との連携は全く考慮されていない．

3. 提 案 方 式

本章ではノードの移動パターンと要求仕様を整理した後，提案方式を説明する．

3.1 ノードの移動パターンと要求仕様

NAT が介在する場合の MN の移動パターン一覧を表 1 に示す．CN はグローバルアドレス空間に存在する場合とプライベートアドレス空間に存在する場合がある．表 1 において，G はグローバルアドレス空間を，P はプライベートアドレス空間を示し，P(A)，P(B)，P(C) はそれぞれが異なるプライベートアドレス空間であることを示す．

8) では (b)，(c)，(d) の移動パターンについて検討しているが，NAT が SPI 機能を持つと実現は難しい．9) では (f)，(h)，(j)，(m) の移動パターンについて検討しているが，NAT に改造を加える必要がある．

表 1 MN の移動パターン一覧
Table 1 Lists of MN's moving patterns

No.	CN	MN (Before Moving)	MN (After Moving)	Reference
(a)	G	G	G	6)
(b)	G	G	P(A)	8)
(c)	G	P(A)	G	8)
(d)	G	P(A)	P(B)	8)
(e)	P(A)	P(A)	P(A)	-
(f)	P(A)	G	G	9)
(g)	P(A)	G	P(A)	-
(h)	P(A)	G	P(B)	9)
(i)	P(A)	P(A)	G	-
(j)	P(A)	P(B)	G	9)
(k)	P(A)	P(A)	P(B)	-
(l)	P(A)	P(B)	P(A)	-
(m)	P(A)	P(B)	P(C)	9)

本論文では，Mobile PPC と TURN の技術を用いて表 1 に示す全ての移動パターンに対応できることを目的とする．本方式では全ての NAT が SPI を搭載していてもよい．また，移動時に両エンドノードの間に NAT が存在しないことが判明した場合は一時的に中継サーバを介した中継通信を行った後，エンドエンドの通信に切り替える．更に Mobile PPC の改造という形でカーネルを改造するため，CN，MN 共にアプリケーションに全く依存しない方式とする．

3.2 提案方式の処理

提案方式について，移動時の一時的な中継通信による通信の継続処理と，移動後のエンドエンド通信への切り替え処理を説明する．

3.2.1 移動時の処理

提案方式の移動時の処理を図 4 に示す．移動パターンは最も複雑な動作となる (h) とする．予め CN は CN のホスト名 CN.expl，CN が属する NAT の IP アドレス $G3$，CN の IP アドレス $P1$ を，MN は MN のホスト名 MN.expl，CN の IP アドレス $G2$ を中継サーバに登録しておく．また移動に備えて CN と MN は予めそれぞれ中継サーバとの間でトンネルセッション $P1:v \leftrightarrow G9:r$，$G9:r \leftrightarrow G2:w$ を確立しておく必要がある．図 4 では既に CN から通信を開始しており，NAT1 にはマッピング $P1:a \Leftrightarrow G3:m$ が既に生成されている．

Vol.2009-MBL-51 No.3
Vol.2009-ITS-39 No.3
2009/11/5

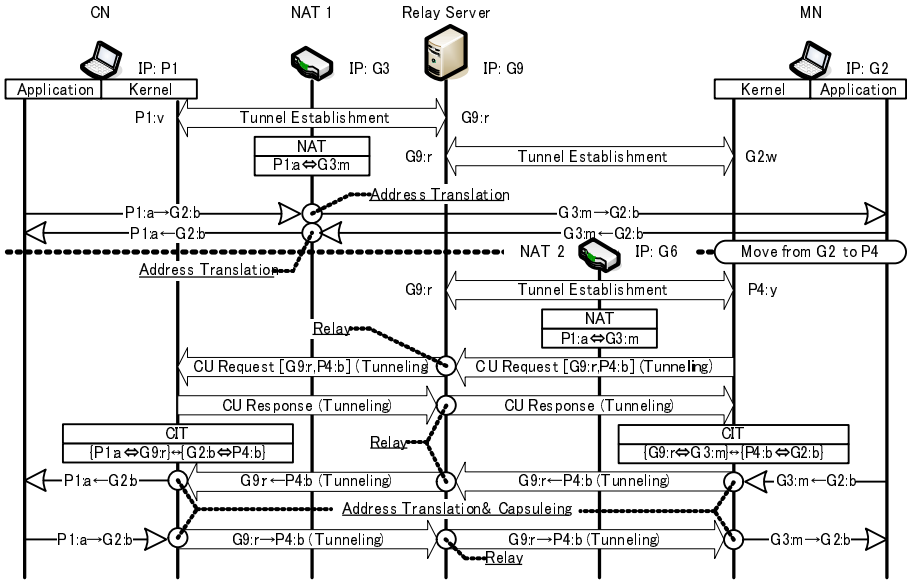

図 4 移動時の提案方式の動作
Fig. 4 Operation of the proposed system when MN moves

ここで MN が IP アドレス $G4$ を持つ NAT2 配下に移動して新しく IP アドレス $P4$ を取得する．MN はまず中継サーバとの間で新たにトンネルセッション $G9:r \leftrightarrow P4:y$ を確立する．これにより CN と MN の間に中継サーバを介した通信経路が確立する．MN と CN はこのセッションを通して Mobile PPC の CU Request/Response を交換する．これにより CN と MN は以下の様な CIT を生成する．

$$\text{CN; } \{P1:a \Leftrightarrow G9:r\} \leftrightarrow \{G2:b \Leftrightarrow P4:b\}$$
$$\text{MN; } \{G9:r \Leftrightarrow G3:m\} \leftrightarrow \{P4:b \Leftrightarrow G2:b\}$$

MN は CIT に従って送信パケットのアドレスを $G3:m \leftarrow G2:b$ から $G9:r \leftarrow P4:b$ に変換した上で，このパケットをトンネルセッション $G9:r \leftrightarrow P4:y$ でカプセル化し，中継サーバに送信する．この時，パケットのメッセージ部に CN のホスト名 CN.expl を記載する．中継サーバはこのパケットを受け取ると，ホスト名 CN.expl から CN 方向のトンネルセッションを選択し，CN に転送する．CN はこのパケットを受け取るとカプセルを外して元

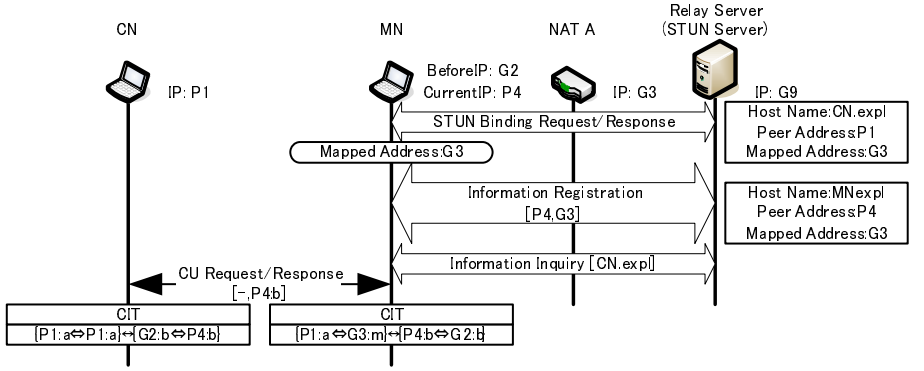

図 5 エンドエンド通信への切り替え
Fig. 5 Changing process to end-to-end communication

のパケット $G9:r \leftarrow P4:b$ を取り出し，更に CIT を参照してアドレスを $G3:m \leftarrow G2:b$ に変換する．CN がパケットを送信する際も上記と同様の処理を行う．

この処理により，ノードはどのような移動パターンであっても通信を継続することができる．本機能は Mobile PPC の改造としてカーネルに組み込むため，アプリケーションには一切依存しない．

3.2.2 エンドエンド通信への切り替え

MN は，CN が同一のアドレス空間に存在すると判断した場合はエンドエンド通信に切り替える．表 1 のパターン (g) において，中継サーバを介した通信からエンドエンド通信への切り替えの様子を図 5 に示す．

移動後，MN は中継サーバと Binding Request/Response を交換することにより，自分が属する NAT の IP アドレス $G3$ を取得する．次に MN は中継サーバに対して MN が属する NAT の IP アドレス $G3$ と自身の現在の IP アドレス $P4$ を中継サーバに報告し，MN の情報を更新する．

更に MN は中継サーバに CN のホスト名 CN.expl を報告し，CN が属する NAT の IP アドレス $G3$ と CN の IP アドレス $P1$ を取得する．MN は自身のマッピングアドレスと CN のマッピングアドレスを比較し，共に $G3$ であれば CN が同一アドレス空間内に存在することを判断できる．同一アドレス空間であると判断した MN は，CN との間で再度 CU Request/Response を交換し．CIT を更新する．これにより，以後の通信では CN はエン

Vol.2009-MBL-51 No.3
Vol.2009-ITS-39 No.3
2009/11/5

ドエンドの通信に移行することができる．

エンドエンド通信に切り替えられるパターンは表 1 中の (a)，(c)，(g)，(l) である．他の移動パターンでは，NAT に SPI が搭載されていることを前提とすると中継サーバを介した通信となることは避けられない．

4. ま と め

IPv4 における移動透過性通信では，NAT の存在を考慮する必要がある．しかし近年の NAT には SPI と呼ばれるフィルタリング技術が実装されることが多く，NAT 越えが極めて困難である．そこで，IPv4 においてエンドエンドで移動透過性を実現できる Mobile PPC と，SPI を搭載した NAT であっても NAT 越えを実現できる TURN の技術を組み合わせ，NAT を越えた移動透過性を実現する方式を提案した．本方式では，MN の移動時に中継サーバとのトンネルセッションを用いて Mobile PPC を動作させて通信を継続する．次にエンドエンドで通信可能かどうかを調査し，可能な場合はエンドエンドの通信へと移行する．また提案機能を Mobile PPC の改造という形でカーネルに実装することにより，アプリケーションに一切依存しない方式とすることができる．

謝辞 本研究の一部は，日本学術振興会科学研究費補助金（特別研究員奨励費 20・1069）の助成を受けたものである．

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) Johnson, D., Perkins, C. and Arkko, J.: Mobility Support in IPv6, RFC 3775, IETF (2004).
4) Kunishi, M., Ishiyama, M., Uehara, K., Esaki, H. and Teraoka, F.: LIN6: A New Approach to Mobility Support in IPv6, *Proceedings of WPMC2000* (2000).
5) 相原玲二，藤田貴大，前田香織，野村嘉洋：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 (2002).
6) 竹内元規，鈴木秀和，渡邊 晃：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，情報処理学会論文誌， Vol.47, No.12, pp.3244–3257 (2006).
7) Ferguson, P. and Senie, D.: Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing, RFC 2827, IETF (2000).
8) 鈴木秀和，渡邊 晃：Hole Punching を用いた NAT 越え Mobile PPC の設計，情報処理学会研究報告， Vol.2008, No.44, pp.69–74 (2008).
9) 鈴木秀和，渡邊 晃：プライベートネットワーク内のノードを通信相手とした移動透過性の実現方式，電子情報通信学会論文誌 (B)， Vol.J92-B, No.1, pp.109–121 (2009).
10) UPnP Forum: *Internet Gateway Device(IGD) Standardized Device Control Protocol V 1.0*, http://www.upnp.org/standardizeddcps/igd.asp (2001).
11) 鈴木秀和，宇佐見庄吾，渡邊 晃：外部動的マッピングにより NAT 越えを実現する NAT-f の提案と実装，情報処理学会論文誌， Vol.48, No.12, pp.3949–3961 (2007).
12) Rosenberg, J., Weinberger, J., Huitema, C. and Mahy, R.: STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network address Translators (NATs), RFC 3489, IETF (2003).
13) Rosenberg, J., Mahy, R. and Matthews, P.: Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN), Internet-Draft draft-ietf-behave-turn-16, IETF (2009).