

通信メディアの特性を利用する Mobile IP システム

森川 大樹、植田 道成、石橋 賢二、中島 達夫

{morikawa,mitinari,ishi,tatsuo}@jaist.ac.jp

http://mmmc.jaist.ac.jp:8000/

北陸先端科学技術大学院大学

通信メディアの多様化と携帯型計算機の進歩は目覚ましい。オフィスビルなどでは、さまざまな通信メディアを利用して構内にネットワークを張り巡らし、計算機を持ち歩くことでどこにいてもネットワーク上のサーバや資源にアクセスできる環境が整っている。こうした環境を活用するために、携帯型計算機は状況に応じて最も有効な通信メディアを選択できる機能を持つべきである。また、通信プロトコルは通信メディアの特性の変化に適応することが望まれる。本稿で提案する IETF Mobile IP 拡張システムはこれらの問題を解決し、より柔軟な移動計算機環境を実現する。

A Mobile IP System that takes into account the characteristics of communication media

Hiroki Morikawa, Michinari Ueda, Kenji Ishibashi, Tatsuo Nakajima

{morikawa,mitinari,ishi,tatsuo}@jaist.ac.jp

http://mmmc.jaist.ac.jp:8000/

Japan Advanced Institute of Science and Technology

Various communication media are developed recently, and portable computers are widely available. In an office building, various wireless communication media can be available, and this makes that users can bring the computers everywhere in the building for accessing information on servers from everywhere. In this situation, portable computers should select the best communication media according to their location. Also, communication protocols should be adapted to respective environments. Our extension of IETF Mobile IP system described in this paper solves the above problems, and makes portable computers more attractive.

1 はじめに

計算機ハードウェアの小型化と通信メディアの多様化によって、時間や場所を選ばずネットワーク上のさまざまなサービスや資源にアクセスできる移動計算機環境が実現している。通信メディアとしては、有線のイーサネット、無線 LAN、公衆回線を利用した ISDN や PHS などさまざまなものが利用可能である。ユーザはノート型計算機などを携帯し、移動した先で有効な通信メディアを選択してネットワークへの接続を維持することができる。

一方、ネットワークに接続された計算機で利用されるアプリケーションは、クライアント・サーバ型の形態を取るものが一般的である。これらのアプリケーションは、クライアントとサーバの間でコネクションの設定を必要とするものが多く、クライアントがサブネットを跨ぐことは想定していない。携帯型計算機の普及に伴って、大学の構内やオフィスビル

内で携帯端末からこうしたクライアント・サーバ型のアプリケーションを利用するケースが増えているが、上記のように計算機を持ち運ぶことができ、移動先でネットワークに再び接続できたとしても、ユーザは作業を中断せざるを得ないのが現状である。

コネクションの切断や設定の変更を必要としないクライアントの移動を、アプリケーション側でサポートする試みもなされているが、個々のアプリケーションで計算機の移動に対応して行こうというアプローチはいかにも非効率的である。

こうした状況を受けて、システムレベルで計算機の移動を透過にするアプローチへの期待が高まっている(図1)。このようなシステムの実現によって、移動の際に作業の中止や復元を行う必要が無い、煩雑なネットワークの設定から解放される、といったユーザにとっての利便性が生まれるほか、アプリケーションレベルで計算機の移動を考慮する必要がなくなるため開発・運用面での有用性も期待できる。また、ク

クライアントの位置的な制約が緩むことで、グループウェアなどのアプリケーションの分野において今までにはない形態のものが登場する可能性もある。

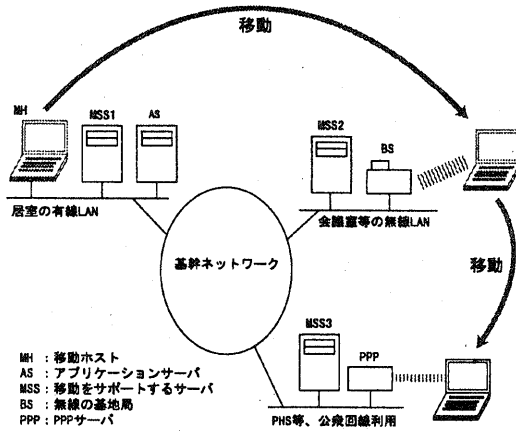


図1: 移動計算機環境

1.1 問題点

しかし、このような環境を実現するためには、解決しなければならない問題が多く残されている。本稿では以下の4つの問題に焦点を当てる。

計算機を識別するためのアドレス

現在のTCP/IPに代表されるネットワークプロトコルは、通信メディアやネットワークの接続位置が恒常的に変化しない計算機環境を想定して設計されたものである。したがって、計算機が移動し、使用する通信メディアやネットワークとの接続位置がその時々によって変化することなどは考慮していない。その顕著な例は、パケットを配送する際の宛先を示すIPアドレスが計算機の識別子としては機能しないという特徴に現れている。つまり、TCP/IPにおいてIPアドレスは計算機とネットワークの接続点の識別子として機能している。計算機のネットワークインターフェースとサブネットの端子の一つを結び付けた時に初めて、有効なIPアドレスが割り当てられるのである。したがって、サブネットをこえた計算機の移動は、この組み合わせの変更を強いるため、TCP/IPだけでは計算機の移動をサポートすることができない。

多様な通信メディア

図1のような環境を想定した場合、移動計算機は複数の通信メディアを常時利用することが可能である。この環境を活用するためには計算機の動作中にそれらを円滑に切り替える仕組みが必要不可欠であ

る。現在のところ、これに十分に対応したシステムはない。

計算機の移動のタイミングおよび移動時の状態管理

これは、一般にハンドオフ処理と呼ばれる。第2の問題とも関連するが、同一のメディア間のハンドオフに関しては種々の最適化が提案されているものの、通信メディアが変わってしまう場合のハンドオフに関して言及しているものはない。

メディア特性の変化

第2、第3の問題を解決しても、単に通信メディアを動的に切り替えられるようにするだけでは、十分な通信性能を得ることはできない。これは、通信メディアの切り替えや移動先のロケーションによって、エラー率、バンド幅、遅延などの特性が大きく異なるからである。したがって、各メディアの特性に応じて通信の性能を向上するための機構を用意することが望ましい。

1.2 本稿での提案

本稿では、以上4つの問題に対して解決方法を検討し、より柔軟な移動計算機環境を実現するシステムを提案する。

第1の問題点を解決するためには、計算機そのものを識別するための一種のアドレスが必要となる。その手段として、まったく新しいプロトコル体系を設計することも考えられるが、その場合、TCP/IPを利用している既存のアプリケーションをすべて変更しなければならない。TCP/IPがネットワークプロトコルとして支配的な存在になっている現在、TCP/IPを利用するネットワークアプリケーションは膨大な数に上り、これらを無視することは現実的ではない。

本稿では、TCP/IPの体系を崩すことなくこの問題に対処するためにIETF Mobile IPをシステムの基盤として採用した。Mobile IPでは、IPアドレスを従来通り接続点の識別子として使用し、その上の層に仮想的なネットワークアドレスの概念を導入する。この仮想的なアドレスを本来のIPアドレスと動的にバインディングすることで、計算機の移動時や通信メディアの切り替え時のIPアドレスの変化を吸収する。これによって、計算機の識別子として機能するアドレスをTCP/IPの上に実現することができる。その基本アーキテクチャは図2のようになっている。

Mobile IPではホームエージェント、フォーリンエージェントの2つのエージェントを使用することにより計算機の移動を可能としている。具体的には、各移動ホストはホームネットワーク上に現在のIPアドレスを管理してもらうホームエージェントを持ち、移動ホストがフォーリンネットワーク上にいる時は、ホームエージェントが移動先のフォーリンエージェントへパケットを転送し、フォーリンエージェント

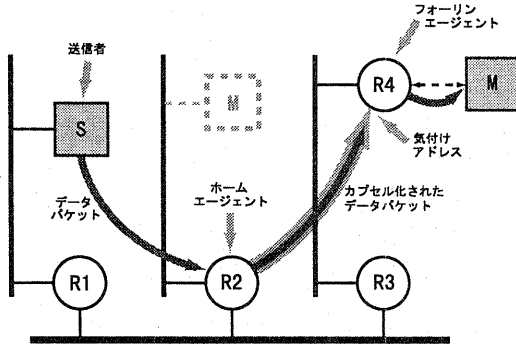


図 2: IETF Mobile IP の基本アーキテクチャ

が移動ホストへ配送する。ホームエージェントからフォーリンエージェントへパケットを転送する際には IPinIP のトンネリングが用いられる。移動ホストには、ホームアドレスと呼ばれる不変の IP アドレスが割り当てられ、これによりホストを識別する。また、移動先のネットワーク接続点を示す IP アドレスとして気付けアドレスが割り当てられる。

このメカニズムによって、とりあえずは、既存のネットワークアプリケーションをそのまま移動計算機環境で使用することができる。

しかし、第 2 の問題には Mobile IP だけでは対処できない。現在の Mobile IP の実装では、移動計算機が利用できる通信メディアは移動先のネットワーク上に存在する FA に依存し、複数のメディアを切り替えて使用することは考慮していない。本来、Mobile IP の FA は計算機の移動を可能にするためだけに存在する。そこで、本システムでは FA-MH 間のプロトコルを拡張し、複数のメディアの選択、切り替えをサポートする。また、通信メディアの選択には、本研究室で開発中の環境情報サーバ [5] から提供される情報を利用する。環境サーバは、バンド幅や遅延といった通信メディアの状態をシステムに与え、システムはこれをもとに通信メディアを決定し、ユーザがもっとも有効なメディアを利用できるようにする。

同時に、通信メディアの切り替え時にどういった効果が表れるのかを検証する。ここで得られた情報をもとに、第 3 の問題への対処として切り替えのタイミングやステートの管理を行うハンドオフサーバを設計、追加する。システムはハンドオフサーバが提供する情報によって、さまざまな通信メディア間で効率よくハンドオフを行うことが可能となる。

第 4 の問題には、弱点となる特性をカバーするためのモジュールを特性ごとに用意し、状況に応じてそれらのモジュールを切り替えることで対処する。たとえば、バンド幅が狭くなるケースでは、パケット圧縮モジュールを用いてバンド幅の有効利用を図る。また、エラー率が高くなるケースでは、エラー

リカバリモジュールを用いてエンド-エンドの不必要な再転送を避ける。この仕組みによって、通信メディアの切り替えおよびロケーションの変化に伴うメディアの特性の変化に適応し、通信の性能が低下することをできる限り防ぐ。

2 システムアーキテクチャ

本章では、本稿で提案するシステムの概要を述べる。

2.1 アドレスの分離

現在、移動計算機環境においてネットワークプロトコルとして TCP/IP を用いる場合、IP アドレスはホスト識別子ではなくネットワーク接続点の識別子となっている。このためホストの移動や通信メディアの切り替えの際に通信が持続できなくなり、計算機の設定を変更したり、再起動しなければならないという問題が起こっている。これは、プロトコルと通信メディアが密に結びつけられているため、IP アドレスが通信メディアに直接割り当てられてしまうことが原因である。

そこで、本システムではプロトコルと通信メディアを明確に分離し、プロトコルの下位に通信メディアを管理する層を配置する。プロトコル側には計算機識別子としての IP アドレスを、通信メディア側にはネットワークへの接続点としての IP アドレスを与える。ここで計算機識別子としての IP アドレスを仮想アドレス、ネットワーク接続点としての IP アドレスを物理アドレスと呼ぶ。(図 3)

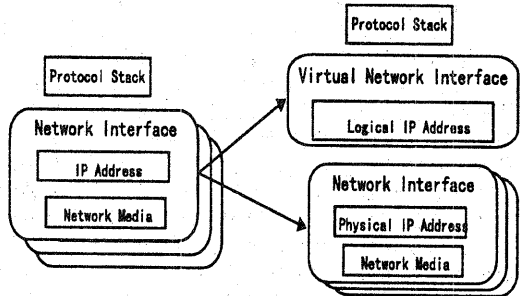


図 3: アドレスの分離

2.2 通信メディアの切り替え

PC カードなどの進歩により動作中に通信メディアを切り替えることが可能となった。例えば、屋内ではイーサネットに接続し、屋外へ移動する際には無線 LAN や PHS といった無線メディアに切り替えることによりネットワークに接続できる。このように一つのホストが二つ以上の通信メディアを備えて

いる場合、その時の状態に応じてどちらを使用するかを選択したり、また、一方の通信メディアが使用できなくなった時に自動的にもう一方に切り替わるといった機能があると通信を行う際に非常に便利である。しかし、これらを実現するには様々な問題がある。例えば、IP アドレスの変化や各通信メディアごとの IP アドレスの確保や ARP などの処理の相違などがある。

そこで本システムでは、通信メディアを仮想化しメディア間の差異を吸収する手法を提案する。IP アドレスの変化に対しては、上記のアドレス分離により計算機自身のアドレスをそのままに通信メディアのアドレスのみを切り替えるようにする。通信メディアでの処理の差異に対しては、仮想化された部分でのみその処理の相違吸収し、その部分の外からその差異を隠蔽する。そして、それらを管理する層を設け、状況によりそれらを切り替えるようにする。こうすることにより、上位層から見えるネットワークインターフェースは仮想インターフェースのみとなり、切り替えの差異に必要となる再構成部分が仮想インターフェースの一箇所に集中しているので通信メディアの切り替えが容易に行うことができる。

2.3 全体構成

本稿で提案するシステムは、IETF Mobile IP を拡張したものである。図4にこのシステムの基本アーキテクチャを示す。

この図において、R0、R1、R2、R3 はルータであり、それぞれのサブネットはこれらのルータを経由して基幹ネットワークに接続されている。とくに、R1、R3 は本システムで提案するホームエージェント（以下 HA）、プロキシーエージェント（以下 PA）であり、主としてこの二つのエージェントが計算機の移動や通信メディアの切り替え、メディア特性の変化への適応といった機能をサポートする。また、MH は移動ホスト、S は MH と通信を行っているホストである。本システムでは IETF Mobile IP と同様に移動ホストには仮想アドレスとしてホームアドレスが割り当てられる。ホームアドレスはそれぞれの移動ホストを識別するために使用される。移動ホストがホームネットワークから離れ、別のサブネットに接続するとき通信メディアに物理アドレスとしてネットワーク接続点を示す気付アドレスが割り当てられる。本システムではこの気付けアドレスは DHCP により割り当てる。

図4に示されたように、移動先のサブネットには PA が存在する場合としない場合の二通りが考えられる。PA が存在する場合としない場合にどのように MH 宛ての packets がルーティングされるかを以下で示す。

2.3.1 PA が存在しない場合

移動先のサブネット上に PA が存在しない場合、MH 宛ての packets は次のようにルーティングされる。

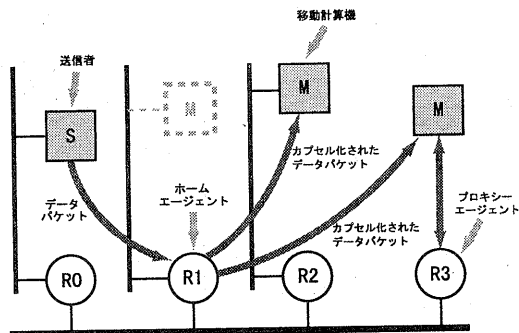


図4: 本システムの基本アーキテクチャ

1. S が MH 宛ての packets を送信する。
2. その packets が HA に届く。
3. HA はその packets に対し IPinIP のカプセル化を行ない MH へ送信する。
4. MH ではそのカプセル化された packets が届くと元の packets に戻し上位レイヤに渡す。

このように PA が存在しない場合には、MH 宛ての packets は HA-MH 間で IPinIP のカプセル化が行われ、MH へ送信される。

2.3.2 PA が存在する場合

移動先のサブネット上に PA が存在する場合、MH 宛ての packets は次のようにルーティングされる。

1. S が MH 宛ての packets を送信する。
2. その packets が HA に届く。
3. HA はその packets に対し IPinIP のカプセル化を行ない PA へ送信する。
4. PA では、その packets を再びカプセル化し MH へ送る。この際、必要に応じて通信性能を改善するためのモジュールを呼び出し適切な処理を行う。
5. MH ではカプセル化された packets を元に戻し上位レイヤへ渡す。また、packets が PA で処理されたものであれば対応する処理を packets に施す必要がある。

このように PA が存在する場合には、MH 宛ての packets は HA-PA 間および PA-MH 間で IPinIP のカプセル化が行われる。また、PA-MH 間での通信性能を改善するために PA や MH で必要なモジュールを呼び出す。

2.4 PA、HA 発見プロトコル

ホストの移動や通信メディアの切り替えによりネットワーク接続点が変わってしまったとき、まず最初に行なうことはサブネット上でPAやHAの存在を確認することである。本システムでは、IETF Mobile IPとの互換性を保つためPA、HA発見プロトコルにはIETF Mobile IPと同様のAgent Discoveryプロトコルを用いる。このプロトコルは、ICMP ルータ発見プロトコルを拡張したものである。具体的には、ホストのIPアドレスが変化するとホストはエージェント懇願メッセージをブロードキャストする。もし、PAやHAがこのメッセージを受け取るとエージェント広告メッセージを送り返すことによりホストへその存在を知らせる。PAやHAからのエージェント広告メッセージが帰って来ないとホストは、移動先がホームネットワーク上ではなく、かつ、PAが存在しないと判断する。

2.5 ホームエージェント

HAは、ホストの移動を可能にするエージェントである。HAでは以下のような処理を行なう。

● ホストの管理

- HAは自分と同じサブネットにホームアドレスを持つMHを管理する。管理下のMHが他のサブネット上に移動した場合、HAはMHが移動先で獲得したアドレス(PAが存在する場合はPAのアドレス)を保持し、そのMHのホームアドレスとの対応を取る。

● ルーティング

- MHがホームネットワーク上に存在する場合
 - * 通常のルータとして動作する。
- MHがフォーリンネットワーク上に存在する場合
 - * PAがフォーリンネットワーク上に存在する場合はMH宛ての packets をカプセル化しPA宛てにその packets を送る。
 - * PAがフォーリンネットワーク上に存在しない場合はMH宛ての packets をカプセル化しMH宛てにその packets を送る。

2.6 プロキシエージェント

PAは、ホストの移動や通信メディアの切り替えを支援し、PA-MH間の通信性能を改善するためのエージェントである。PAでは以下の処理を行なう。

● 登録

- PAは自分が置かれたサブネットにMHが移動してきた際に、そのMHがサブネット上で獲得したアドレスを保持する。同時に、そのMHを管理するHAに対してこのアドレスを登録する。

● パケットのスヌーピング

- PAはパケットのスヌーピングを行ない、そのパケットがMH宛てであると判断すると、必要に応じてPAと移動ホスト間の通信性能を改善するモジュールを呼び出す。そして、そのパケットを再びカプセル化しMHへ送る。

2.7 ネゴシエーションプロトコル

MHがフォーリンネットワーク上へ移動しそのネットワーク上にPAが存在するケースを考える。このとき、PA-MH間が無線LANなどで接続されているとこの部分が通信路上のボトルネックとなり得る。そこで、本システムではこの部分の通信性能を改善することでエンド-エンドの通信パフォーマンスの向上を図る。実現には、通信性能を改善するためのモジュールをPA、MHの両方で呼び出すという方法を用いる(詳細は第3章)。このとき、PAとMHでどのモジュールを用いるかについてPA-MH間で交渉を行うプロトコルが必要である。現段階では簡単に以下のようなプロトコルを想定している。

1. MHがPAへ使用している通信メディアの情報(通信メディアの種類、バンド幅、エラー率など)を送る。
2. PAはその情報をもとに使用するモジュールを選択しMHに対してどのモジュールを使用するかを知らせる。
3. MHがPAに対してAckを返す

3 多様な通信メディアへの対応

本システムの目的は、より柔軟な移動計算機環境を実現することにある。通信メディアが多様化したことによりネットワークに接続するための手段は確かに増えたが、それを個別にしか扱えない現行のシステムでは柔軟性に乏しい。多様な通信メディアに対応するためには、通信メディアの切り替えを許す円滑な計算機の移動を実現しなければならない。これには異なる通信メディア間のハンドオフをどのように扱うかが鍵となる。

同時に、通信メディア間の特性の違い、すなわちバンド幅、エラー率、遅延などの違いが著しいという問題に対処しなければならない。そのためには通信メディアの切り替えによる特性の変化にいかに対応し、効率の良いデータ転送をいかに維持できるか鍵となる。

また、従来の資産を活かすため、そしてより現実的なシステムを実現するためには、既存のネットワークになるべく手を加えることなく実現できることが望ましい。本章では、こうした特徴を持つシステムをどのように実現するかについて設計方針を述べる。

3.1 ハンドオフ管理

本稿で想定する移動計算機環境において、計算機は状況に応じて利用可能な通信メディアを切り替えながらネットワークとの接続を維持する。とくに無線リンクを利用する場合は、基地局の位置が通信可能距離を制限してしまうので、ロケーションに応じて接続する基地局を変更したり、通信メディア自体を切り替えなければならない場合がある。この問題を解決するための方法はハンドオフ処理と呼ばれ、移動計算機環境を実現するために不可欠な技術である。(図5)。

本節では、有線ネットワーク、無線ネットワーク、PHS などの様々な通信メディアが利用可能な環境において、効率よくハンドオフ処理を行うために計算機の移動のタイミングおよび移動時の状態を管理するハンドオフサーバを提案する。

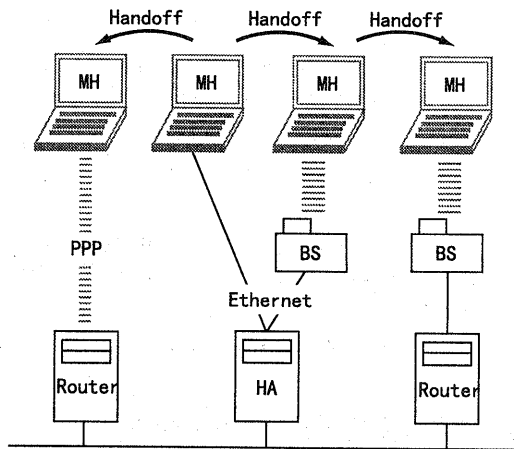


図 5: ハンドオフ

3.1.1 有線と無線

移動計算機環境において、計算機は同時に複数の通信メディアを利用できる場合がある。本研究室で開発中の環境サーバ [5] は、こうした状況下で利用可能な通信メディアの状況をモニタリングしており、ハンドオフサーバは環境サーバの情報をもとに最適な通信メディアを選択する。有線ネットワークが利用可能である場合、ハンドオフサーバは優先的に有線ネットワークに接続するように設定を行う。

3.1.2 無線間

移動計算機と無線基地局間の通信は Netwave Technologies 社の Netwave を使用する。このシステムは基地局ごとに割り当てられるドメインと呼ばれるチャンネルによってサブネットを分割している。現在の仕様では同じドメインを持つ基地局間で移動した場合、自動的に切り替えを行うことが可能であるが、ハンドオフによる遅延は考慮されていない。そこで、ハンドオフをチューニングするために、まず、基地局の切り替わりを監視し、ドメインを制御できるようにデバイスドライバの修正を行なう。これによって、ハンドオフの処理をハンドオフサーバ側でコントロールできるようにする。Netwave では無線の状態など、ハードウェアレベルの接続状況を知ることができ、これによって移動計算機と基地局間のバンド幅の状態を知ることができる。こうした情報は環境サーバが常時モニタリングしており、メディアを切り替えるタイミングは環境サーバの情報をもとに判断する。

3.1.3 PPP を用いる場合のハンドオフ

イーサネットから PHS を利用した PPP 接続へ切替える場合について考える。通信メディアを選択するためのポリシーは多様に想定できるが、本システムではできる限り接続が寸断されないことを第一とする。PHS を利用した PPP 接続は物理的な接続を行った上でダイアリングによって回線を繋がないといけない。さらに接続時に認証を必要とする場合も多い。実際に、モデムを介した PPP 接続には 10 数秒を要し、必要になってから接続し始めるのでは遅延がかなり大きくなってしまふ。一方、必要となった時にすぐに使用できるように常時 PPP の接続を維持するという方針はコストがかかり過ぎるため適切ではない。そこで本システムでは、環境サーバの情報をもとに使用中のリンクが寸断されそうだと判断した場合、あらかじめ先行してダイアリングすることによって接続を維持するという方針を採る。また、PPP とイーサネット間のハンドオフのように、ハンドオフ後にメディアの特性がまったく変わってしまう場合がある。この特性の違いをできる限り吸収し、通信メディアの切り替わりを意識させないシステムを実現する。次節ではこのシステムの詳細について述べる。

3.2 パケットスヌーピング

本来、Mobile IP で規定しているフォーリンエージェント (FA) は、計算機の移動を可能にするためだけに存在する。本システムの狙いは、ネットワーク上の FA の位置に着目しこれをより積極的に利用していこうということにある。具体的には、FA を基盤に、通信メディアの特性の変化に適應するための機能を追加したプロキシエージェント (PA) を提案する。

移動計算機 (MH) が PA を通じて通信を行う場

合、MH-PA 間は通信路の末端部分に相当し、一般に最も貧弱な無線やPHSなどの回線となると考えてよい。つまり、PAは通信路上の変速点に位置しているため、これをチューンアップすることが通信路全体のパフォーマンス向上につながると考えられる。ただし、本システムでは複数の通信メディアを状況に応じて切り替えることを想定しており、それぞれのメディアの特性に従ってチューンアップの方法を変更できるように適応性の高い仕組みが必要となる。これを実現するために、我々はパケットスヌーピングによる方法を採用した。ネットワークインターフェースからプロトコルスタックに渡る途中で全てのパケットをスヌーピングし、必要なパケットのみをピックアップして対応するチューンアップモジュールに渡す。

この方法を用いることの利点は第一に、メディア特性に応じたチューンアップモジュールをプロトコルスタックとは切り離して設計することが可能となることにある。メディアの特性ごとにモジュールを用意し、特性の変化に応じてモジュールを動的に差し替えることで通信の性能を改善する。また、既存のネットワークに手を加えることなく実現できることも大きな特徴である。

図6がこのシステムの概要を示しており、MHおよびPA内のsnooperがすべてのパケットをスヌーピングしている。snooper内でPA-MH間のパケットを横取りし、状況に応じて必要なモジュールに渡すことで、メディアの特性の変化に適応する。また、後に別のモジュールが必要になった時に簡単に付け加えられるような仕組みを提供する。また将来的には、MHがPAに使う欲しいモジュールを送り、PAはそれを動的に組み込むといった形も考えられる。現状ではテストケースとして、パケット圧縮とエラーリカバリの二つのモジュールを想定している。以下では、それぞれのモジュールの用途と設計の概要について述べる。

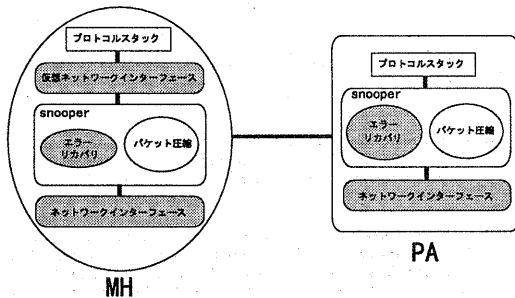


図6: パケットスヌーピング

3.2.1 パケット圧縮

パケット圧縮モジュールは、PHS等、バンド幅の狭い回線に接続している場合に、MH-PA間の双方

向のパケットに圧縮を施し、回線を通るデータのサイズをできる限り小さくすることで、バンド幅を有効に利用できるようにする。このモジュールの効果は、パケット圧縮処理によってパケットを送り出す(吸い上げる)タイムロスが生じてもバンド幅を十分に使い切ることができるような状況で期待できる。

また、TCPやUDPのポート番号などのヘッダの情報からペイロードの特性が予測できるパケットに関してはそれに依りて、ヘッダ圧縮、ペイロード圧縮、パケット連結などの各種パケット圧縮法を使い分ける。

構造としては単純に、MHとPAに同様のモジュールを用意し、snooperからMH-PA間のパケットをすべて渡してもらい、圧縮解凍処理を行う。モジュール間は独自のプロトコルを用いて各種圧縮パケットのやり取りを行う。(図7)

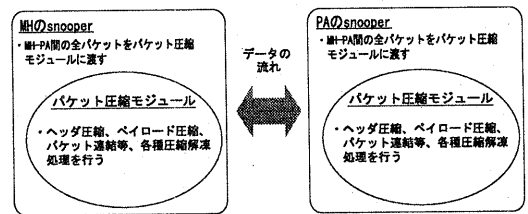


図7: パケット圧縮

3.2.2 エラーリカバリ

エラーリカバリモジュールは、無線LAN等、エラー率の高い回線に接続している場合に、TCPのデータ転送において高速なエラー回復を可能にするため、MH-PA間でローカルな再転送を処理する。

従来のTCPはパケットロスの主要原因が輻輳にあるような有線ネットワークで十分に機能するように最適化されている。一方、昨今登場してきたワイヤレスリンクにはビットエラー率が高くなるという特徴があり、ワイヤレスリンクを利用したネットワークのパケットロスは、多くの場合、ビットエラーによって生じる。このような特徴を持つネットワークで従来のTCPをそのまま用いると、たとえば輻輳回避や高速リカバリのアルゴリズムが不必要に働いてスループットを低減させてしまう可能性がある。

そこで、通信路上でMH-PA間に顕著なメディア特性のむら(エラー率がまちまちであることを意味する)を遮断するため、MH-PA間のみでエラーのリカバリを行うモジュールを提供する。基本的には[6]をもとに、パケットのトラッキングとキャッシングによってPA-MH間で失われたパケットの再転送を行う。以下では、ローカルな再転送を行うためのアルゴリズムと、それを実現するためにPAに必要なとなる

機能、MH に必要な機能の概要について述べる。

固定ホスト (FH) から MH に向かうデータのエラーリカバリには、PA のモジュールに以下の二つの機能を持たせることで対処する。

- MH から確認応答 (ack) が来ていない TCP データパケットのキャッシング
- MH からの ack をトラッキング

TCP は順番が狂って到着したパケットに対して重複 ack を返す。そこで、PA のエラーリカバリモジュールは、MH から返された重複 ack をもとに、失われたパケットを特定し、キャッシュの中から対応するパケットを取り出して再転送を行う。また、MH からの通常の ack は、キャッシュに貯えているパケットをクリアするためのトリガになる (図 8)。

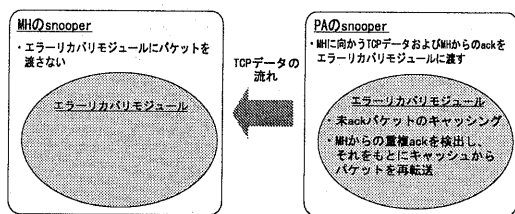


図 8: MH に向かうデータのエラーリカバリ

MH から FH へ向かうデータのエラーリカバリには、PA と MH のモジュールにそれぞれ以下の機能を持たせることで対処する。

- PA 側: MH からのパケットをトラッキング
- MH 側: 未ack TCP セグメントのキャッシングと PA からのネガティブ ack (n-ack) への応答

PA はパケットのトラッキングによって、失われたパケットを検出し、n-ack を MH に送ることで対応するパケットを再送させる (図 9)。

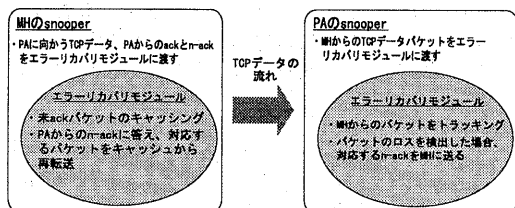


図 9: MH からのデータのエラーリカバリ

このメカニズムによってコネクションの双方向のパフォーマンス低下を防ぐことが可能である。

4 おわりに

現在の移動計算機環境には様々な問題点がある。本稿では

- サブネットを越えて移動するホストを識別するためのアドレス機構がない
- 通信を維持した状態で通信メディアを切り替えることができない
- 異なる通信メディア間でのハンドオフは言及されていない
- 通信メディアの特性の変化に対応できない

といった4つの問題に焦点を当てこれらを解決するための方法を提案した。本稿で提案したシステムを用いることで、通信を維持した状態での計算機の円滑な移動を可能にし、通信メディアの切り替えやそれに伴うメディア特性の変化にも十分に対応することができる。

現在は、このシステムを実現するために設計の詳細を詰めている段階である。今後は、このシステムを Real-Time Mach 上に実装し、性能評価と有用性について検討していく。

参考文献

- [1] 中島 達夫, "モバイルコンピューティングのための動的適応可能なソフトウェアアーキテクチャ", 第16回IPA技術発表会, 1997.
- [2] 中島 達夫, "モバイルコンピューティング" 無線LANアーキテクチャ, 第5章, 共立出版, 1997.
- [3] C.Perkins, "IP Mobility Support", RFC2002, October, 1996.
- [4] D.A.Maltz, D.B.Johnson, "The CMU Monarch Project IETF Mobile IPv4 Implementation User's Guide", <http://www.monarch.cs.cmu.edu/>, February, 1997.
- [5] Tatsuo Nakajima, Hiroyuki Aizu, Masaru Kobayashi, Kenji Shimamoto, "Environment Server: A System Support for Adaptive Distributed Application", The 2nd International Conference on Worldwide Computing and Its Applications'98 (WWCA'98), March, 1998.
- [6] Hari Balakrishnan, Srinivasan Seshan, Randy H.Katz, "Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks", ACM Wireless Networks, December 1995.
- [7] 小林 勝, 中島達夫, "動的なメディア選択が可能な Mobile IP の設計と実装", IPSJ OS, February, 1998.