

P2P ストリーミングプロトコルに関する検討

小俣栄治[†] 石川憲洋[†] ヨハンジェレム^{††} 村上慎吾^{†††} 泉知論^{††††} 中村行宏^{††††}

[†]株式会社 NTT ドコモ マルチメディア研究所

^{††}エリクソン・リサーチ

^{†††}エリクソン・リサーチ・ジャパン

^{††††}京都大学 大学院 情報学研究科 通信情報システム専攻

P2P ストリーミングとは、従来の Server/Client 環境で行われているストリーミングとは異なり、大規模なサーバを必要としないマルチメディアコンテンツ配信技術である。P2P ストリーミングでは、P2P ネットワークに参加しているノード間を、パケツリレー方式でデータの送受信を行う。本研究の目的は、様々なネットワークが混在し、多種多様な端末の利用が想定されているユビキタス環境において、効率的な P2P ストリーミングを実現することである。本稿では、ユビキタス環境で想定されている多様なネットワークや端末を利用した場合にも対応可能な、P2P ストリーミングを実現するためのプロトコルについての検討を示す。

A Study on P2P Streaming Protocol

Eiji Omata[†] Norihiro Ishikawa[†] Johan Hjelm^{††} Shingo Murakami^{†††}

Tomonori Izumi^{††††} Yukihiro Nakamura^{††††}

[†]NTT DoCoMo, Inc. Multimedia Laboratories

^{††}Ericsson Research

^{†††}Ericsson Research Japan

^{††††}Kyoto University Department of Communications and Computer Engineering

P2P streaming is multimedia contents distribution technology which does not need a large-scale server. In P2P streaming, all the nodes on P2P network play a role of distribution, relay, and reception, and transmit streaming data from multi-hop. Various networks and terminals are intermingled in a ubiquitous environment. The purpose of this research is realizing efficient P2P streaming in such a ubiquitous environment. In this paper, we show the P2P streaming protocol on P2P platform.

1. はじめに

近年、Peer-to-Peer(P2P)ネットワーク技術が注目を集めており、グループウェア[1]や分散検索システム[2]など、様々な P2P アプリケーションが開発されている。また、JXTA[3]のような汎用的な P2P プラットフォームなどの研究も行われている。一方、インターネットの急速な発展に伴い、多くのリソースがネットワーク上に存在するようになっており、従来の Server/Client モデルでのリソース管理が困難になりつつある。そのことから、P2P ネットワークを用いた新しい形の分散コンピューティングアプリケーションが注目を集めており、盛んに研究開発が行われている。

また、インターネット環境において、双方向で高速通信が可能な FTTH が普及し始めており、ユ

ーザの所有する端末(PC等)の CPU スペックも大幅に向上している。そのような理由から、ユーザ自身がマルチメディアコンテンツのような大容量データを配信することが可能になりつつあり、ユーザ間での P2P を用いたマルチメディアデータのストリーミング技術が注目されている。P2P ストリーミングは、各ユーザがリソース (CPU やネットワーク帯域) をシェアして行うことから、大規模なサーバを必要とせず、さらにネットワークの負荷を軽減させることが可能である。

そこで本研究では、ユビキタス環境における P2P ストリーミングの利用を想定し、そのような環境においても効率的なデータ転送を実現することを目的とする。

本論文の構成は、2章においてこれまでに提案

している P2P プラットフォームの概要について述べ、3 章では提案する P2P ストリーミングのアーキテクチャについて述べる。さらに、4 章では検討に基づいて設計したプロトコルを述べる。次に 5 章では、P2P ストリーミングを用いたアプリケーション例を挙げる。6 章では、本研究の関連研究を挙げ、7 章において今後の検討を示す。最後に 8 章でまとめを述べる。

2. P2P プラットフォーム

2.1. P2P アーキテクチャ

これまでに我々は、図 1 に示すようなユビキタス環境（IP ネットワークや Bluetooth[4]、IEEE1394[5]などが混在した環境）上に、分散している端末（携帯電話や PDA 等）がシームレスに接続し、利用可能であるための P2P プラットフォーム[6]の提案を行ってきた。

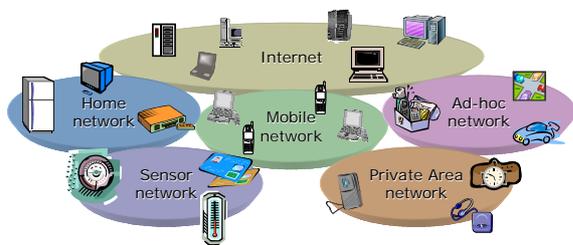


図 1：ユビキタス通信環境

提案している P2P プラットフォームには、以下のような構成要素がある。

◇**P2P ノード**：P2P ノードは独立した、双方向通信機能を持った構成要素を想定している。携帯電話や PDA、PC など、あらゆる通信端末を想定している。

◇**P2P ネットワーク**：P2P ノードにより構成されたネットワークである。P2P ノード間では、相互信頼を前提として、接続を確立する。各 P2P ノードは、独立した構成要素であり、自由にネットワークに参加したり、離脱したり出来る。

このように想定する P2P ネットワークは、様々な環境に存在するデバイス同士がそれぞれ接続を確立しており、その論理的な接続に基づいて通信を行うアプリケーションネットワークである。

2.2. モバイル P2P プロトコル

これまでに提案している P2P プロトコルは、図 2 のような構造となっている。P2P コアプロトコルはノード間のメッセージ転送の制御を行うプロトコルである。上位の P2P プロトコル群は、アプリケーションに依存しない P2P ネットワークの汎用的な機能を提供するプロトコル群である。これらのプロトコル群は、ノード間の接続情

報を管理するためのメッセージや、ノード間のルーティングの制御を行うためのメッセージを定義し、動的に変化する P2P 通信に対応したルート情報の管理等に用いられる。本提案プロトコルは下位のトランスポートプロトコルと独立に設計し、様々なトランスポートプロトコル上で動作を可能としている。また、全てのプロトコルは XML を用いて設計を行った。

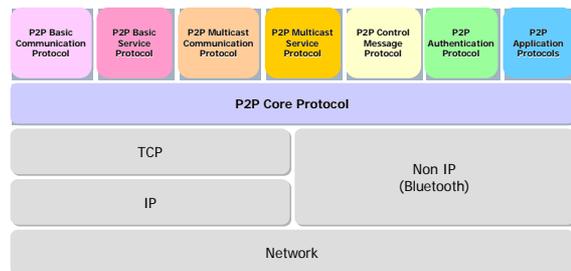


図 2：P2P プロトコルスタック

2.3. プロトタイプ

これまでに、提案している P2P の有効性実証のために、P2P プラットフォームの実装を行ってきた。図 3 に示すように、実装したプラットフォームソフトウェアは Java (J2SE 1.3.1) 及び C++ で、プロトコルは XML を用いて実装している。

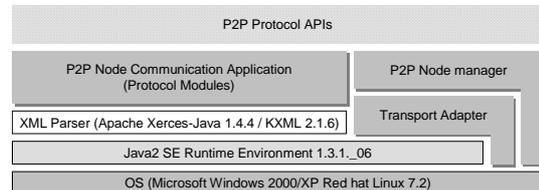


図 3：ソフトウェアアーキテクチャ

3. P2P ストリーミングアーキテクチャ

P2P ストリーミングとは、P2P ネットワーク上でマルチメディアデータを、バケツリレー方式で運ぶ仕組みである。P2P ストリーミングでは、P2P ネットワーク上の全てのノードが、配信・中継・受信の役割を担い、マルチホップでストリーミングデータの転送を行う。

本研究では、上記のような P2P を用いたマルチメディアデータのストリーミングとして、特に、ユビキタスで想定されているような異なった通信環境を跨った P2P ストリーミングを、安定でかつ効率的に実現するための方式を検討する。

3.1. 要求条件

P2P ストリーミングを検討するにあたり、以下の項目を要求条件とした。

■ ユビキタス環境でのストリーミング

ユビキタス環境で想定されているような IP、non-IP ネットワークが混在した環境において、シームレスな P2P ストリーミングを実現することが求められる。

■ 多様な通信形態への対応

様々な通信形態（1 対 1、1 対多）に対応した上で、効率的にストリーミングデータを配信することが求められる。

■ 耐障害性

P2P ノード間の接続、切断が頻繁に起こる環境であっても、中断することなく、ストリーミングを行うことが可能であることが求められる。

3.2. ユビキタス環境でのストリーミング

ユビキタス環境における P2P ストリーミングの利用を想定した場合、そのような環境においても効率的にデータ転送を行うことが可能な方式を検討する必要がある。

インターネットやホームネットワークなどのネットワークには、そのネットワークの特性に合ったマルチメディア転送方式が提案されている。例えば、インターネットを利用する場合であれば、RTP（Real-time Transport Protocol）[7]、ホームネットワークを利用する場合には、IEEE1394 の Isocronous 転送などがある。しかし、これらのネットワークを跨って、経路等の管理を行うためのシグナリング方式は、あまり検討されていない。

そこで本研究では、ユビキタス環境において効率的な P2P ストリーミングを実現するために、ストリーミングデータは各々のネットワーク特性に合ったデータ転送方式を利用する。さらに、様々なネットワークを跨ってストリーミングデータ転送するために、制御メッセージを用いて経路等を管理するためのシグナリングプロトコルについて検討する。

3.3. 多様な通信形態のサポート

ストリーミングには、様々な形態が考えられるが、一般的には、放送型ストリーミングとオンデマンド型ストリーミングの 2 つの配信形態に大別することができる。そこで、これらの配信形態をサポートするために、1 対 1 の通信方式（以下ではユニキャスト方式と呼ぶ）と 1 対多の通信方式（以下ではブロードキャスト方式と呼ぶ）について検討する。

シグナリング処理の流れを以下のステップに分けて検討を行った。

[1] 経路の検索：ストリーミングデータを転送するための経路を検索する

[2] 経路の設定：ストリーミング開始時のストリ

ーミングデータの経路を設定する

[3] 経路の維持：データ転送中の帯域制御、経路分断時の処理を行う

[4] 経路の開放：ストリーミング終了時の経路開放を行う

3.3.1. ユニキャスト方式

[1] 経路の検索

現在の P2P プラットフォームにおけるメッセージ転送のための経路検索は、下位のトランスポート層を意識することなく、送信ノード（もしくは受信ノード）のノード ID をキーとして検索を行い、最も早く経路情報を取得することが出来た経路を選択している。しかし、ストリーミング配信を行う上では、QoS を考慮した経路を選択することも、重要な要素となる。そこで、QoS を考慮した経路検索について検討する。

帯域を保障した経路を検索してストリーミングを行う場合を例として示す。インターネットでは帯域保証を行うことは困難であるが、IEEE1394 などは帯域保証が可能な伝送媒体である。そのことから、下位の伝送媒体を考慮した検索を行うことにより、あらかじめ帯域を確保した上でストリーミングを行うことが可能である。そこで、転送経路上に存在するノードの、下位トランスポート層を指定して検索を行う手法を検討する。図 4 に示すように、送信ノードは任意の帯域（例：10M）が保障可能な経路でストリーミングを行うことを希望しているとする。このような場合、まず現在の P2P プラットフォームと同様に、送信ノードと受信ノード間での最短経路を検索する。その結果、図 4 の経路 1 が最初に得られた転送経路である。しかし、経路 1 では、中継ノード A と B の間で帯域保障が出来ないとする。そのような場合、中継ノード A は受信ノードまでの経路を再検索する。その結果得られた経路 2 で、同様の処理を行う。

このように、要求に応じて複数の検索手法を用いることにより、様々な伝送媒体が混在した P2P ネットワークにおいて、帯域を考慮した経路検索を実現することが可能となる。

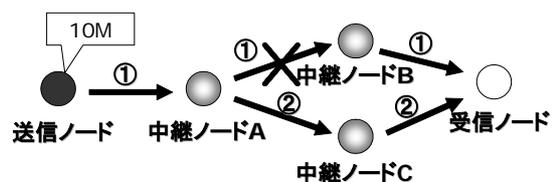


図 4：帯域を考慮した経路検索

[2] 経路の設定

ユニキャスト方式の場合、送信ノード主体でス

トリーミングを開始する放送型と、受信ノード主体でストリーミングを開始するオンデマンド型が考えられる。そこで、上記2つのストリーミングに対応したユニキャスト方式について検討を行った。

まず始めに、放送型ユニキャスト方式について検討を行った。提案する P2P ストリーミングでは、各ノードがどのような経路でどのデータを転送するのかをあらかじめ定める必要がある。そのため、ストリーミングを開始する前に、制御メッセージを用いてルーティングテーブルを作成する。ルーティングテーブルの作成には、以下のような要求条件が考えられる。

- 各ノードは、ストリーミングフローを一意識別出来なければならない。
- 各ノード間のトランスポートプロトコルは、ノード間毎に決定することが可能とする。
- 各ノードは、転送先のトランスポート層に応じた待受けアドレスを知っていなければならない。

要求条件より検討した経路設定フローを、図5に示す。

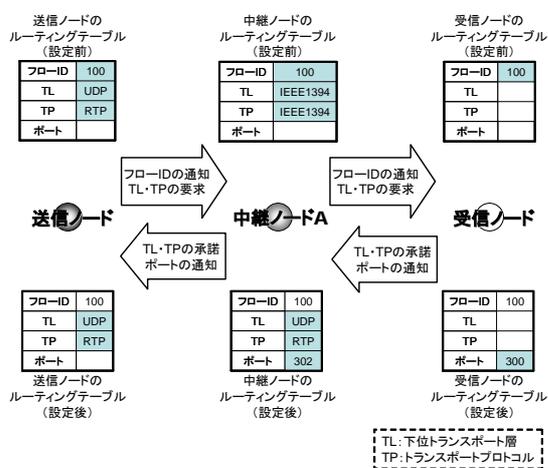


図5：経路設定フロー

まず、ストリーミングフローを一意識別するために、ID (フローID) を定義する。このフローID は、送信ノードがストリーミングを開始する際に、ストリーミングフロー毎に生成し、他のノードに対して通知する。その結果、転送経路上の各ノードも、ストリーミングフローを一意識別することが可能となる。

さらに、ストリーミングデータの送信には、送信先ノードのアドレスが必要である。伝送媒体がUDP/IPの場合、アドレスはIPアドレス+ポート番号となる。しかし、ポート番号の範囲は制限さ

れているため、送信ノードがポート番号を一意に決定すると他のストリーミングフローのポート番号と重複してしまう可能性がある。そこで、制御メッセージを受信した経路上の各ノードは、ノード内で一意となるポート番号を自ら決定する。決定したアドレスと転送プロトコルを各ノード間で設定するために、制御メッセージは、シングルホップ転送 (送信ノード→中継ノードA、中継ノードA→受信ノード) を用いて行うこととする。

図5に示す一連の流れを行うことにより、各ノードは転送されてきたストリーミングデータをどのノードに対して送信するかを定めることができ、データ転送が可能となる。また、オンデマンド型ユニキャスト方式に関しては、受信ノードが送信ノードに対してストリーミングの開始要求を行う。続く経路設定処理は、放送型ユニキャスト方式と同様である。

[3] 経路の維持

P2P ネットワークはノードの出入りが頻繁に発生するため、ストリーミング実施中に転送経路が切断されていないかを確認する必要がある。そこで、提案する P2P ストリーミングでは、経路の切断をすばやく検知するために、ソフトステート管理方式を用いる。しかし、経路維持メッセージを定期的に送信しているだけでは、切断してしまったことは検知出来ても、予約資源の解放を行うことは出来ない。

そこで、各ノードは経路予約の状態を、タイマを用いて維持する。経路を設定したノードは、タイマが切れる前に転送経路上のノードに対して状態維持を要求する。この要求を受けた転送経路上の各ノードは、タイマをリセットする。その結果、転送経路の切断や P2P ネットワークの輻輳により維持メッセージが届かなくても、予約資源を解放でき、利用資源を最小限に防ぐことが可能である。

経路の切断を検知した後の経路回復処理については、3.4 で後述する。

[4] 経路の解放

ストリーミングの実施中、もしくは終了時に、終了を要求するノードはフローID を指定して、設定経路を解放する。ストリーミングを終了する状況としては、以下の場合が考えられる。

- ・送信ノードによるストリーミングの終了
- ・受信ノードによるストリーミングの終了

3.3.2. ブロードキャスト方式

[1] 経路の検索

ブロードキャスト方式の場合、送信ノードから

P2P ネットワーク上に存在するノードに対して、P2P ブロードキャストメッセージを送信する。各ノードは、最も早く届いたメッセージのみを受信し、あとから届いたメッセージは破棄する。その結果、送信ノードをルートとした、ツリーを構築することが可能である。

[2] 経路の設定

ブロードキャスト方式の経路設定の場合、ユニキャスト方式の要求条件に加えて、以下のような要求条件が考えられる。

- ストリーミングデータを受信したいノードが複数存在する。
- P2P ネットワーク上の全てのノードが、必ずしもストリーミングデータを受信するかわからない。
- 後から P2P ネットワークに参加したノードも、ストリーミングデータを受信可能とする。

上記の要求条件から検討した、ブロードキャスト方式の経路設定手順を図 6 に示す。

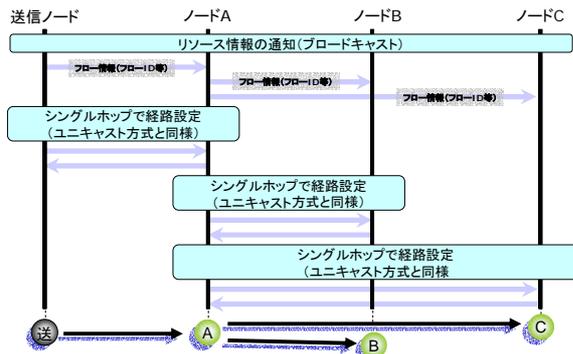


図 6 : ブロードキャスト方式の経路設定手順

要求条件より、送信ノードが全てのノードに対して経路設定を行うよりも、ストリーミングデータを受信したいノードが、送信ノードに対して行うほうが効率が良いと考えられる。そこでブロードキャスト方式では、ユニキャスト方式とは異なり、受信ノードから経路の設定を行う。

しかし、受信ノードは送信ノードがどのようなデータを送信するかを知らない状態にある。そこで、図 6 に示すように、送信ノードはあらかじめストリーミングデータのフロー情報を全ての受信ノードに対してブロードキャストする。このフロー情報のメッセージと経路の検索メッセージは、同様なメッセージとすることも可能である。さらに、送信ノードは定期的にフロー情報を P2P ネットワーク上の全てのノードに通知することにより、P2P ネットワークに参加したタイミングに問わず、全てのノードはストリーミングを受信

することが可能である。フロー情報を受信したノードのなかで、ストリーミングデータの受信を希望するノードは、各自送信ノードの方向に経路の設定処理を行う。但し、利用資源を最小限にするため、経路設定の際には同様のフローID で経路が予約されていないかを確認し、予約されている場合、新たな経路設定は行わない。このときの各ノード間の設定方式は、ユニキャストと同様の方式とする。

[3] 経路の維持

先述したように、経路の設定は受信ノード側から行う。P2P ネットワークでは、ノードの出入りなどが頻繁に起こることが考えられるため、ユニキャスト同様に、定期的にメッセージを送信して、経路の維持を行う。経路の切断を検知した場合、経路の再設定を行うことになる。そこで、経路の設定元から行うことにより、経路の再設定が容易に行うことが可能である。ブロードキャスト方式では、定期的に受信ノードは、上位ノード（送信ノードに近いノード）に対して、経路状況確認メッセージを送信する。

[4] 経路の解放

ユニキャスト方式とは異なり、ブロードキャスト方式の場合、ストリーミングの終了は以下の3つのパターンに分けられる。

- 1) 送信ノードがストリーミングの終了を要求する。
- 2) 受信と転送を行っているノードがストリーミングの終了を要求する。
- 3) 受信のみを行っているノードがストリーミングの終了を要求する。

1、3 の場合は、ユニキャスト方式と同様であり、ストリーミングを終了したい場合は、予約した経路の解放を要求し、ストリーミングの視聴を中断する。しかし、2のノードの場合、経路の解放処理を行ってしまうと、転送を受けている（下流）ノードもストリーミングを中断しなくてはならなくなる。つまり、2のようなノードが、ストリーミングの視聴を中断する場合、経路の解放処理は行なってはならないことになる。そこで、2のノードが受信を終了する場合、視聴のみを中断し、経路の解放を行わない。

3.4. 耐障害性

P2P ストリーミングでは、ノードが出入りするために経路が切れてしまい、ストリーミングデータを転送できなくなる可能性がある。そのため、3.3 で述べたように経路をソフトステートで管理し、経路の切断を検知した際には、他の経路に切

替える必要がある。そこで、ストリーミングデータの欠落を最小限に防ぎ、ストリーミングを継続するための方式を検討する。現在、図7に示すような2つの基本的な方式を検討している。

図7(A)の方式は、あらかじめ主経路と重複しない副経路を用意しておく方式である。中継ノードAが離脱したことにより経路が切れてしまった場合、即座に副経路に副経路に切り替える。この方式では、あらかじめ副経路を検索しておくことにより、すばやく切替えることが可能である。しかし、副経路においても経路設定をしておくことになるため、資源の無駄遣いになってしまう。

図7(B)に示す方式は、経路が切れてしまったノードから、再度経路を検索する方式である。中継ノードBが離脱したことにより経路が切れてしまった場合、上流の隣接ノードは受信ノードまでの経路を再検索する。この方式の場合、最小限の経路切替えを行うことにより、無駄な資源を省くことが可能である。しかし、分断地点の把握が困難なことから、あらかじめ副回線を用意しておくことができない。そのため、経路の切替えに時間がかかってしまう。

本研究では、方式(B)を採用する。モバイル環境などでは、利用資源は最小限に留めることが求められる。方式(B)の場合、現在利用している経路をある区間では利用し続けることが可能であり、あらかじめ多くの資源を予約することも避けられるため、資源の無駄遣いを抑制することが可能である。耐障害性に関しては、様々な方式が提案されており、今後も動的な環境において効果的な方式を検討する。

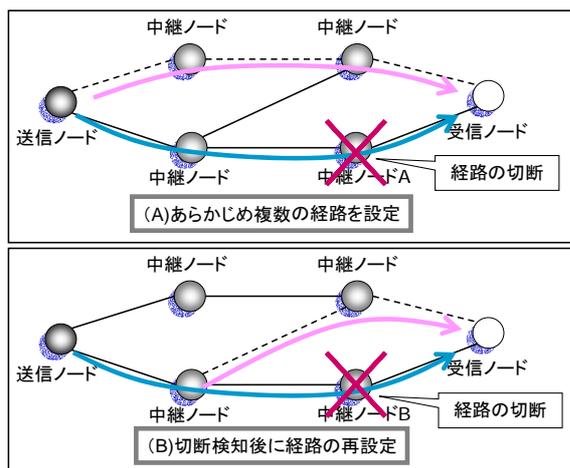


図7：マルチパス

4. P2P プロトコルの設計

4.1. P2P プロトコルの概観

上記で述べた検討に基づき P2P ストリーミン

グプロトコルの設計を行った。提案する P2P ストリーミングプロトコルを、これまでの P2P プラットフォームへ導入するに際して、図8に示すようなプロトコルスタックとした。また、新たに定義した P2P Streaming Control Protocol のメソッドとメッセージの一覧を表1に示す。

■ P2P Streaming Control Protocol

P2P プラットフォームでは、拡張性や伝達性が高いという理由からプロトコルはXMLを用いて実現している。ストリーミングデータの転送制御を行うプロトコルは、P2P メッセージとしてXMLを用いて新たに定義した。

■ Streaming Transmission Protocol

従来のリアルタイムプロトコルを利用。P2P ストリーミングの開始時に、各ノード間で決定して、ストリーミングデータの転送を行う。

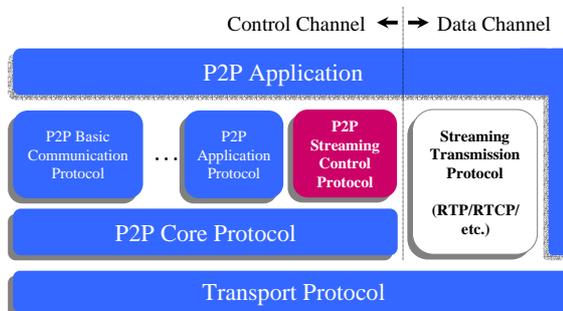


図8：プロトコルスタック

表1：メソッドとメッセージ

No.	Method	Message
1	Setup	Setup
		SetupResponse
2	Release	Release
3	Abort	Abort
4	Prune	Prune
5	Deliver	DeliverRequest
		DeliverResponse
6	Keepalive	Keepalive
		KeepaliveResponse
7	FlowInformationAdvertise	FlowInformationAdvertise
8	BroadcastSetup	BroadcastSetup
		BroadcastSetupResponse
9	BroadcastKeepalive	BroadcastKeepalive
		BroadcastKeepaliveResponse

5. ストリーミングアプリケーション

P2Pストリーミングでは、P2Pプラットフォーム上で用いるストリーミングのため、下位トラン

スポーツ層に依存しないストリーミングデータ配信が可能である。検討しているアプリケーションの1つとして、AV機器ストリーミングアプリケーションがある。ホームネットワーク内に存在するAV機器（ビデオ、テレビ、CDデッキ等）と、モバイル端末（屋外で利用可能な端末）を、P2Pプラットフォームを用いて接続し、P2Pストリーミングを実施する。このアプリケーションにより、宅内の機器に保存している映像や音声データを、屋外でも視聴することが可能となる。

提案するアプリケーションの概観を図9に示す。

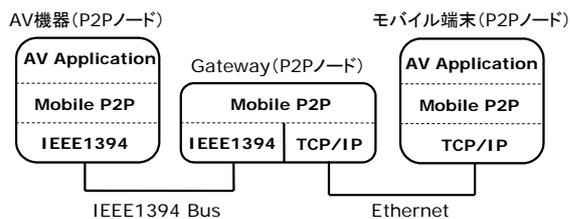


図 9 : AV 機器ストリーミングアプリケーション

しかし、現状ではAV機器にP2Pプラットフォームを導入することは困難なことから、図10に示すような実装を検討している。

図10の実装案に示すように、AV機器の代わりとなる論理P2Pノードを構成するためのプロキシが存在する。このプロキシ内部には、AV機器の代わりとなる論理P2Pノード、ホームネットワーク内のAV機器の接続状況を監視するプロセスとゲートウェイノードが存在する。このような構成にすることにより、最終的な目標となる図9を実現する場合において、AV機器プロキシノードを直接AV機器上で移行させればよいだけになる。

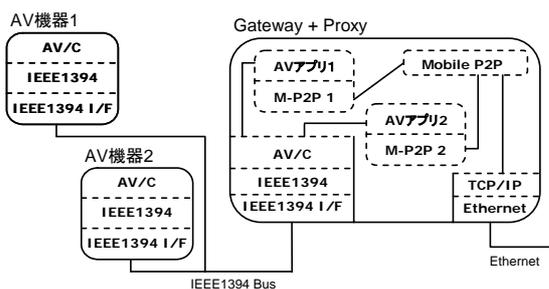


図 10 : 実装案

ゲートウェイ+プロキシ内部では、AV機器がホームネットワークに接続されると、IEEE1394監視プロセスは、プロキシゲートウェイ内に対応するプロキシノード（論理ノード）を生成して、ゲートウェイノードに接続させる。また、機器の接続が外れると、対応するプロキシノード（論理ノード）をゲートウェイノードから接続を外して

消去させる。

現在多くのAV機器に採用されているIEEE1394では、AV制御用のプロトコルとして、AV/Cプロトコルが容易されている。このAV/Cプロトコルを用いることにより、AV機器を外部から操作することが可能である。そこで我々は、このAV/Cコマンドを利用して、Mobile P2P AV Application Command based on AV/C (PAAC)を作成する。PAACは、AV/CメッセージをXML化してP2Pメッセージとする。モバイル端末-AV機器プロキシノード間では、PAACによりメッセージを転送し、AV機器プロキシノード-AV機器間では、AV/Cを利用する。このように、新たなアプリケーションプロトコルを定義することにより、今後AV/Cを利用している機器であれば、IEEE1394を利用していない機器であっても、外部から操作することが可能となる。

6. 関連研究

提案している P2P ブロードキャストストリーミングは、一般的にアプリケーションレイヤマルチキャストとして研究が行われている。

Narada[8]は、コンテンツを視聴するノードのみから構成されるアーキテクチャとなっている。さらに Narada では、P2P ネットワーク上の全てのノードが参加してマルチキャストツリーの構築している。提案する P2P ストリーミングでは、P2P ネットワーク上の希望するノードのみに配信することが可能である。さらに、我々が提案した P2P ストリーミングでは、ユビキタス環境での利用を想定しており、様々なネットワーク環境に対応した P2P ストリーミングを実現することを目的としている。

7. 考察及び今後の課題

1 対多の配信をユニキャストで行うと、送信ノードの負荷が大きくなってしまい、非効率になってしまう。また、P2P ネットワーク上に存在する全てのノードがストリーミングに参加するわけではない場合、配信メッセージをブロードキャストすると非効率である。そこで、任意のグループメンバであるノードに対してのみ、効率的にデータ配信を行うための P2P マルチキャストを用いたストリーミングについて検討を行う。

P2P マルチキャスト方式について、図 11 に示すような検討を行っている。

1. 送信ノード主体で経路設定を行う
 - 方式概容：ユニキャスト方式を応用した経路設定方式
 - メリット：全ての受信ノードを一元的

に把握することが可能

- **デメリット**：P2P ストリーミングの開始契機が、送信ノードに依存してしまう。
2. 受信ノード主体で経路設定を行う
- **方式概要**：ブロードキャスト方式を応用した経路設定方式
 - **メリット**：受信ノードが、自由に P2P ストリーミングを開始することが可能
 - **デメリット**：受信ノードが分散しているため、整合性を取ることが困難

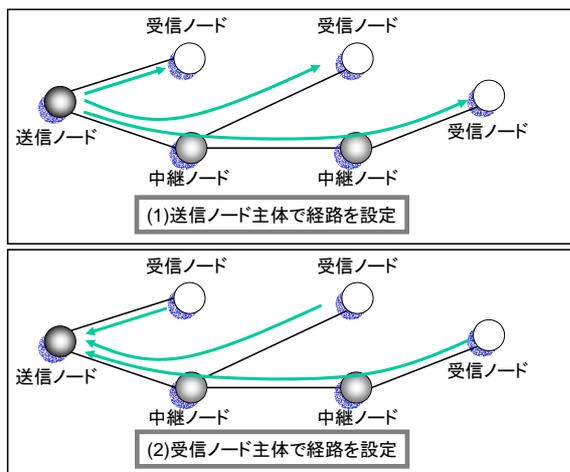


図 11：P2P マルチキャスト方式の検討

効率的なグループ通信を実現するために、今後も検討を続ける。

8. おわりに

本論文では、まずこれまでに我々は提案しているユビキタス環境に対応した P2P プラットフォームについて示した。さらにその P2P プラットフォーム上で効率的で安定した P2P ストリーミングを実現するためのプロトコルについて検討した。

今後の課題としては、P2P ネットワーク上に存在する任意のグループのみで P2P ストリーミングを実現する P2P マルチキャストストリーミングや、帯域や遅延等を考慮した転送を行う方式に関する検討がある。

謝辞

本研究の実施にあたり、京都大学学術メディアセンタ美濃導彦教授、沢田篤史助教授には貴重な助言を頂きまして深く感謝致します。また、ご協力頂いた京都大学中村研究室の皆様にも感謝致します。

参考文献

- [1] GROOVE, <http://www.groove.net>
- [2] Gnutella, <http://gnutella.wego.com>
- [3] Sun Microsystems, "Project JXTA", <http://www.jxta.org>
- [4] Bluetooth, <http://www.bluetooth.com>
- [5] 1394 Trade Association Specifications, <http://www.1394ta.org/>
- [6] N. Ishikawa, et al, "A Platform and Applications for Mobile Peer-to-Peer Communications", www2003 workshop on Emerging Applications for Wireless and Mobile Access, May 2003.
- [7] H. Schulzrinne, "RTP: A Transport Protocol for Real-time Applications", RFC 1889, IETF, January 1996
- [8] Y.Chu, S.Rao and H.Zhang, "A Case for End System Multicast", ACM SIGMETRICS, June 2000.