

テクニカルノート

# P2P ストリーミングにおける NAT 越えを考慮した 配信木構築手法の提案

山場 久昭<sup>1,a)</sup> 鮫島 慎治<sup>1</sup> 池田 匡視<sup>1</sup> 油田 健太郎<sup>2</sup>  
岡崎 直宣<sup>1</sup> 富田 重幸<sup>1</sup> 朴 美娘<sup>3</sup>

受付日 2012年1月28日, 採録日 2012年4月2日

**概要:** P2P ストリーミングによるコンテンツ配信を配信木を構成して行う手法は、高いスケーラビリティが実現できることで知られている。しかし、ネットワークへの接続に NAT を利用しているノードは双方向通信ができないので、このようなノードの割合が多い現状では、配信木への参加成功率が低下するなどの問題が発生してしまうことが懸念される。そこで本論文では、配信木に参加済みのノードから新規参加ノードに向かってセッションを張る手法を併用することによって、この影響を抑える手法を提案する。さらに計算機実験を行うことにより、提案手法は NAT を利用するノードが多い環境でも配信木構築への影響が少なくなることを確認した。

**キーワード:** ピアツーピアシステム, ネットワークアドレス変換, P2P ストリーミング, 配信木

## Proposal of a Method for Distribution Tree Construction on P2P Streaming System in Consideration of NAT Traversal

HISAAKI YAMABA<sup>1,a)</sup> SHINJI SAMESHIMA<sup>1</sup> MASAMI IKEDA<sup>1</sup> KENTARO ABURADA<sup>2</sup>  
NAONOBU OKAZAKI<sup>1</sup> SHIGEYUKI TOMITA<sup>1</sup> MIRANG PARK<sup>3</sup>

Received: January 28, 2012, Accepted: April 2, 2012

**Abstract:** A method for distributing streaming contents on P2P using a distribution tree is known to be suitable because of its high scalability. However, there are several issues such as a fall of tree construction success rate which are caused by a restriction that nodes cannot create sessions to other nodes using NATs. In this paper, a new method of distribution tree construction was proposed by initiating a session from a node in a tree to a new node. It was also confirmed that this new method had higher performance for constructing a distribution tree than classical one through computational experiments.

**Keywords:** Peer-to-Peer systems, network address translator, P2P streaming, distribution tree

### 1. はじめに

近年、マルチメディアコンテンツのストリーミング配信サービスの需要が高まってきており、それにもなって配

信サーバへの負荷も増大してきている。この負荷を軽減する手立てとして期待されているのが、Peer to Peer (以下、P2P) を利用したストリーミング配信法である。具体的には、Application Level Multicast を採用し、受信データを複製して他の複数のノードに送信する能力を各々のノードに付与することにより、配信木と呼ばれる木構造のトポロジを構成してコンテンツを配信する手法が注目されている [1], [2], [3]。この手法の長所は、配信を行う際のノードの負荷を分散できることから、高いスケーラビリティの実現と配信元負荷の減少が可能な点にある。しかしその一方

<sup>1</sup> 宮崎大学  
University of Miyazaki, Miyazaki 889–2192, Japan  
<sup>2</sup> 大分工業高等専門学校  
Oita National College of Technology, Oita 870–0152, Japan  
<sup>3</sup> 神奈川工科大学  
Kanagawa Institute of Technology, Atsugi, Kanagawa 243–0292, Japan  
a) yamaba@cs.miyazaki-u.ac.jp

で、その配信木構築アルゴリズムは参加するノードが双方向通信可能であることを前提として設計されているために、双方向通信ができないノードの割合が増加するにつれ、配信木への参加成功率が低下したり参加の際のオーバーヘッド増加したりしてしまう。実際、ネットワークアドレス変換 (Network Address Translation, 以下 NAT) を利用してネットワークに接続しているために、双方向通信ができないノードが多いのが現状であり、この問題への対策は急務であるといえる。そこで本論文では、NAT を介したネットワーク接続に起因するこの問題を解決できるような配信木構築手法について検討を行う。

## 2. 従来手法

配信木を利用した P2P ストリーミング配信の代表的な手法の 1 つに PeerCast [1], [2], [3], [8] がある。ここでは、まずこの手法での配信木構築の概要を述べてから、この種の P2P ストリーミング配信において、NAT を利用したノードが多い場合に生じる問題点について示す。

### 2.1 配信木構築手法

配信木の親ノードは、自分に接続しているノード (子ノード) のアドレス情報を格納したリスト (子ノードリスト) を持つ。この「子ノードリスト」を再帰的に取得していくことによって、根ノードを開始点として、配信木に参加しているすべてのノードのアドレスが取得可能、すなわち、配信木にすでに参加しているすべてのノードに接続要求を送ることが可能である。配信木に新規に参加しようとするノード (以下、新規参加ノードと呼ぶ) は、まず、根ノードのアドレスを取得し、根ノードから始めて幅優先探索の順序に従って接続要求を送信し、接続を許可したノードを親ノードとして配信木に参加する。

### 2.2 NAT が配信木に与える影響

NAT とは、グローバル IP アドレスとプライベート IP アドレスを変換する技術である。NAT テーブルと呼ばれる対応表で、プライベートネットワークに接続しているノードの「プライベートアドレスとポート番号の組」(ソケットアドレス) と「グローバルアドレスとポート番号の組」(以下、マッピングアドレスと呼ぶ) を 1 対 1 に対応づけ (ポートマッピング) しておく。これにより、ブロードバンドルータ内部の複数のノードが、1 つのグローバル IP アドレスでインターネットに接続することができる [7]。ただし、NAT を利用してインターネットに接続しているノードに対して、インターネットの側から接続することができないので、このことが P2P 通信ネットワークを構築する際の大きな障害となる。すなわち、前述の従来法では、配信木に新規に参加したノードが今度は親ノードとなり、さらに別のノードがそこに接続できることから、配信木は高

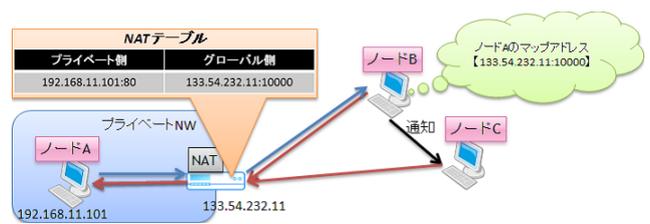


図 1 ホールパンチング

Fig. 1 Hole punching.

いスケラビリティを実現できていたわけであるが、NAT を利用しているために双方向通信ができないノードが配信木に参加した場合、このノードに対しては新たなノードが接続できない。このため、配信木のスケラビリティが損なわれてしまう。

具体的には、以下のような事態が予想される。

- 参加成功率の低下  
配信木に参加しようとするノードの中で NAT を利用しているノードの割合がきわめて高いと、双方向通信可能なノードの帯域を消費し尽くしてしまいやすくなり、それ以上新たなノードが配信木に参加できなくなる。
- 参加オーバーヘッドの増大  
(NAT を利用しているため) 親ノードになれないノードに対しても接続要求を行ってしまうことにより、この種のノードが増えると、参加オーバーヘッドが増大すると考えられる。
- ストリーミング遅延の増大  
同様に、この種の親ノードになれないノードの増加により、構成される配信木が深くなり、末端のノードへ配信されるまでの時間遅れ (ストリーミング遅延) が増大する。これは特に、同期性を求められる生放送などのライブストリーミングにおいて大きな問題になると考えられる。

実際、ファイル共有 P2P アプリケーションを利用しているノードの 70~80% が、NAT を利用してインターネットに接続しているという研究報告 [5] もあり、この問題への対応は必須の状況にあるといえる。

### 2.3 ホールパンチング

NAT を利用しているノードへのインターネットの側からのアクセス (NAT 越え) を実現する代表的な手法にホールパンチング [4], [5] がある (図 1)。

NAT を利用しているノード (図 1 のノード A) から外部のノード (図 1 のノード B) へセッションが張られると (図 1 の青の矢印)、ノード B はノード A のマッピングアドレスを知ることができる。このマッピングアドレスを、ノード A に対して通信要求を送りたいと考えている他のノード (図 1 のノード C) に通知してやれば (図 1 の黒の

矢印), そのマッピングアドレスを利用することによって, ノード B やノード C の側からブロードバンドルータを越えて通信する (図 1 の赤の矢印) ことが可能となるわけである.

ただしこのとき, 内部ノードのソケットアドレスがどの外部ノードとの通信でもつねに同じマッピングアドレスにマッピングされる必要がある. ブロードバンドルータは, それぞれが用いている NAT テーブルのポリシーにより, Full Cone NAT, Restricted Cone NAT, Port Restricted Cone NAT, Symmetric NAT の 4 種類に分類されることが RFC 3489 で規定されている [6] が, この条件を満たし, ホールパンチングを実現できるのは Full Cone NAT のみである.

### 3. 提案手法

#### 3.1 提案手法の概要

前節で述べた問題を解消するために, PUSH セッションと呼ぶ接続方法を導入することにより, NAT を利用してインターネットに接続しているノードであっても, それが親ノードになれる手法を提案する. 本論文では, 特定のサーバに負荷をかけないという P2P の特徴を生かすことを考え, PUSH セッションを実現するうえで新たなサーバをシステムに導入することはしない方針をとる.

基本的なアイデアは次のようなものである. まず, 「実際にストリーミング配信を行うオーバーレイネットワーク」(以下, P2P アプリケーション NW と呼ぶ) とは別に, 「配信木構築に使用するオーバーレイネットワーク」(以下, P2P 制御 NW) を導入する. NAT を利用しているノード X は, 配信元ノードを検索するために, まず P2P 制御 NW に参加するが, このとき, NAT を利用していないノード Y との間にセッションを確立しておく. これにより, ノード X が親候補となった場合, 新規参加ノード Z は接続要求をノード Y を介してノード X に伝えることができるようになる. ノード X がこの接続を許可する場合, ノード X の側からノード Z との間にセッションを確立し, ストリーミング配信を開始するわけである. 既存手法で用いられている, 新規参加ノードから親ノード候補の方向に確立するセッションを PULL セッション, 親ノードの側から新規参加ノードの方向に確立するセッションを PUSH セッションとそれぞれ呼ぶことにする (図 2).

またここで, 新規参加ノードと親候補ノードのそれぞれが, NAT を利用しているか否か, 利用しているのであればどのような種類の NAT を利用しているのかに応じて, セッションを適切に選択するプロトコルを提案する. すなわち, (1) 新規参加ノードが Full cone NAT を利用しているノードであれば, ホールパンチングを用いることにより既存手法と同様の PULL セッションが可能となること, (2) PUSH セッション用いる場合, P2P 制御 NW を介して接続要求を送る際のオーバーヘッドがかかること, の 2 つ

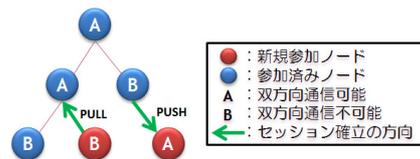


図 2 セッション確立の種類  
Fig. 2 Types of sessions initiation.

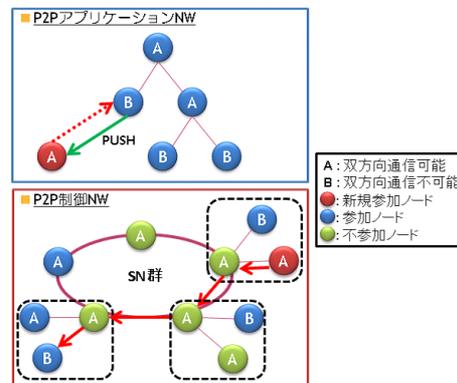


図 3 2 種類のネットワークの構成と基本動作  
Fig. 3 Compositions and basic activities of 2 types of network.

を考えると, この種のノードの組合せでは PUSH セッションを用いず, ホールパンチングを用いる方が良いと考えられる. このように, 親子双方のノードの接続環境に応じて接続に用いるセッションを適切に使い分けることが好ましい. そこで, ノードの接続環境に応じたクラス分類と, 親子双方のノードのクラスの組合せに応じたセッションの選択の手続きを導入する.

なお, 本論文はライブストリーミング配信を対象としているので, すべての通信は UDP によるものとする.

#### 3.2 P2P 制御 NW を活用した NAT 越えの実現

本論文では, ノードの接続環境に応じてノードを 3 つのクラスに分類する. すなわち, NAT を利用していないノードを class 1, Full cone NAT を利用しているノードを class 2, それ以外の NAT を利用しているノードを class 3 と呼ぶ. class 2 のノードは, ホールパンチングを利用することにより, 双方向通信が可能となるので, class 1 と class 2 のノードが双方向通信が可能なノード, class 3 のノードは双方向通信ができないノードである.

本手法で用いる, 2 つのオーバーレイネットワークとそれを用いた PUSH セッション確立の概要を図 3 に示す. 以下では, スーパーノード型 P2P (スーパーノードと呼ばれる選択されたノードがインデックス情報を持つような P2P) を用いて P2P 制御 NW を構築するものとして説明するが, 本手法を実現するうえで P2P 制御 NW の構成方法をスーパーノード型 P2P にすることは必須ではなく, 種々の構築法の採用が可能である.

本手法では既存手法と同様に, まず配信木の根となるコ

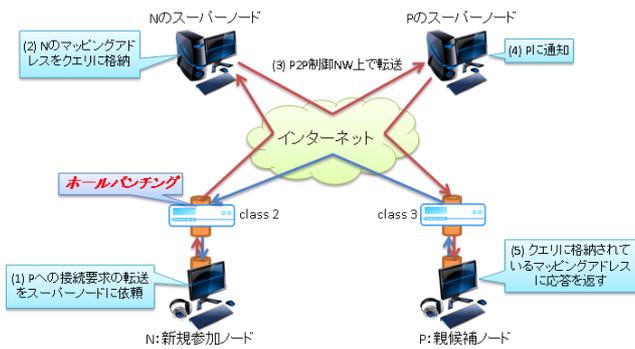


図 4 SCMT の下でのセッション確立  
Fig. 4 Session creation under SCMT.

コンテンツの配信元ノードを探索するが、これは P2P 制御 NW を用いて行っておく。すなわち、新規参加ノードはインデックスサーバから近隣のスーパーノード (class 1 であるものとする) のアドレスを取得し、そこへ参加要求を送り、P2P 制御 NW に参加する。これにより、この新規参加ノードが双方向通信ができない場合であっても、そのスーパーノードはこのときに用いられたマッピングアドレスを利用して、このノードにアクセスすることが可能となる。

またこのとき、新規参加ノードはスーパーノードからユニークな ID を与えられる。ここで、この ID は、そのスーパーノードを特定することもできるような形にしておくものとする。したがって、この ID さえ分かれば、スーパーノードに仲介してもらうことにより、class 1 や class 2 のノードから class 3 のノードへメッセージを送ることが可能となる。

このメッセージ送信時に使用する通信プロトコルを Session Control Message Transfer (以下 SCMT) と呼ぶ。このプロトコルの下でのセッション確立までの動作を、簡単に説明する (図 4 を参照)。

- (1) 新規参加ノード N は、接続要求のクエリに親候補ノード P の ID を格納し、N のスーパーノードに転送を依頼する。
- (2) N のスーパーノードは、N からの転送依頼のパケットを解析し、N に接続するためのアドレス (N のクラスが class 1 であればソケットアドレス、class 2 の場合はマッピングアドレス) を取得し、それを N から受け取ったクエリに格納する。
- (3) クエリに格納されている ID を基に、クエリは P2P 制御 NW 上で P のスーパーノードへ転送される。
- (4) P のスーパーノードは、転送されてきたクエリを P に通知する。
- (5) P は、クエリに格納されているアドレスを取り出し、そこへ応答を返す。

さて、根のノードを発見した新規参加ノードは、既存手法と同様に、根のノードを起点として配信木をたどりながら、配信木を構成するノードに接続要求を送っていく。既

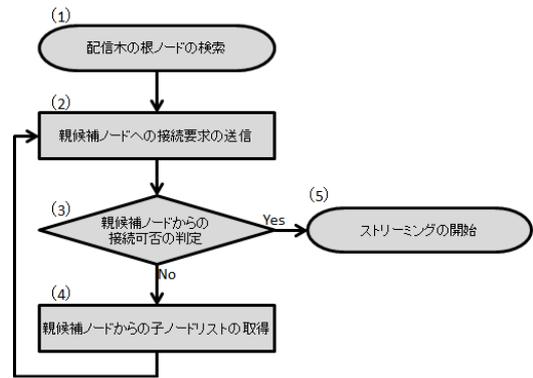


図 5 配信木参加手順  
Fig. 5 Join procedure to a distribution tree.

存手法では、各ノードが持つ子ノードリストにはそれぞれのノードのアドレスが格納されているが、本手法では、各子ノードのクラスに応じて、class 1 ならそのソケットアドレス、class 2 であればマッピングアドレス、class 3 であれば P2P 制御 NW の ID を、それぞれ格納しておくことにする。ただしこのとき、各子ノードのクラスが 1~3 のいずれであるのかについてもあわせて記録し、記録されているデータがアドレスであるのか ID であるのかを識別できるようにしておかなければならない。アドレス、ID の値そのものは、各ノードが接続要求を行う際に親ノードに伝えることにしておけばよいが、各ノードは、一般に自らが使用しているブロードバンドルータのポリシーは知らされないで、そもそも自分のクラスがなんであるかのデータを持たない。そこで本論文では、P2P 制御 NW に参加する際に STUN (Simple Traversal of UDP through NATs) アルゴリズム [6] を適用し、自分が NAT を利用しているのか、使用している場合にどの種類の NAT を利用しているのかの情報を取得し、それに基づいて自らのクラスを特定しておくという方法をとる。こうして特定されたクラスの情報を、接続要求に含めて送信するものとする。

以上のようにしておくことにより、新規参加ノードは子ノードリストに含まれる情報から、各親候補ノードのクラスを、そのアドレスまたは ID とともに知ることができる。ある親候補ノードのクラスが class 1 または 2 (双方向通信可能) であれば、記録されているソケットアドレスまたはマッピングアドレス宛に、接続要求を直接送る。親候補がこれに応じれば、PULL セッションが確立される。一方 class 3 であった場合は、SCMT により、記録されている ID を利用して、接続要求を P2P 制御 NW を介して親候補ノード宛に送信する。親候補ノードがこれに応じれば、親候補の方から PUSH セッションが確立される。

### 3.3 配信木構築アルゴリズム

本手法における配信木参加手順を図 5 に示す。

本アルゴリズムでは、既存手法と同様に、配信木を構成

するノード群に対し、配信木の根ノードから始めて基本的には幅優先探索の順序に従って、接続要求を行う。探索は、標準的な *open* と *closed* のリストを用いる方法で行う。幅優先を採用した理由は、ストリーミング配信の遅延を抑制することを考え、配信木の浅い位置のノードを優先的に選択したいからである。

一方、*open* (親候補ノードリスト) にノードを加える際の、同じノードの子ノード間での優先順位は以下のとおりとする。

- (1) 新規参加ノードが双方向通信できない場合は双方向可能なノードを、双方向通信できる場合はできないノードを優先する。
- (2) 上で同じ順位の場合は、配信木への参加時刻の早いものほど優先する。

前者の優先順位を採用したのは、(a) 双方向通信できないノードの親は、双方向通信できるノードにせざるをえないことと、(b) 一方、双方向通信できるノードの親ノードは双方向通信できてもできなくてもかまわないが、双方向通信できないノードが親を選べる可能性が高くなるように、双方向通信できないノードを優先したいことからである。後者を採用したのは、より早く配信木に参加したノードは、より長くサービスを受けており、そのようなノードが優先的に他ノードへのサービスを行うのが公平であると考えられたからである。

(1) 配信木の根ノードの検索

ストリーミングデータの配信元である根ノードのアドレスは、DHT (Distributed Hash Table) などを利用して取得できるようにしておく。新規参加ノードはこれを利用して根ノードのアドレスの取得する。根ノードは *open* に格納される。

(2) 親候補ノードへの接続要求の送信

*open* の先頭のノードを親候補として選択し、接続要求を送信する。

(3) 親候補ノードからの接続可否の判定

接続が許可された場合は (5) へ、拒否された場合は (4) へ進む。

(4) 親候補ノードからの子ノードリストの取得

接続が許可されなかった場合、新規参加ノードは親候補ノードが持つ子ノードリストを取得し、それを前述の優先度を用いて並べ替えたうえで、*open* の末尾に加える。さらに、親候補ノード自身を *closed* に追加したうえで、(2) に戻る。

(5) ストリーミングの開始

接続が許可された場合、新規参加ノードは、当該の親候補ノードを親ノードとして選択し、ストリーミングデータの受信を開始する。またこの親ノードは、当該の新規参加ノードを自身の子ノードリストに追加する。

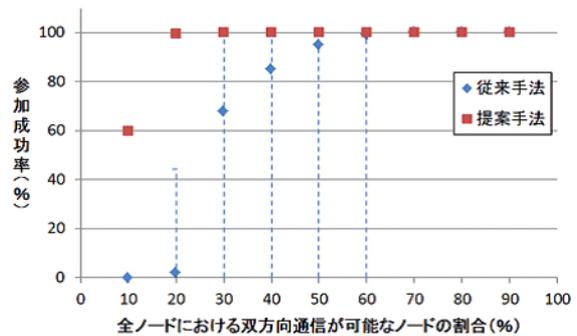


図 6  $N_c=5$  のときの参加成功率  
Fig. 6 Join succession rates on  $N_c=5$ .

## 4. 実験

従来手法と比較して、本手法が参加成功率の向上や配信木の深さを浅くすることが可能であるか調べるための実験を行う。一般に、配信木の参加成功率や配信木の深さは新規参加ノードが参加する順序に依存するため、様々な参加順序で評価する。

### 4.1 方法

双方向通信可能な根ノードを配置し、異なる接続条件を持つ 10,000 個の新規参加ノードをランダムに参加させる。各ノードの帯域は同じと仮定し、ノードが接続できる子ノードの数を  $N_c$  とし、すべての新規参加ノードのうち双方向通信が可能なノードの割合 ( $R_b$ ) を 10~90% と変化させ、各方式の参加成功率と配信木の深さを評価する。なお、実験は 100 回行いその平均値を示す。

### 4.2 結果

#### 4.2.1 参加成功率

図 6 は  $N_c = 5$  のときの参加成功率の平均と最大、最小を示している。従来手法は 60% の双方向通信可能なノードがいなければ参加成功率を 100% にすることができないのに対して、本手法は 20% でほとんどのノードが参加できている。さらに、本手法は試行した 100 回の参加成功率のばらつきがほとんどないことが分かった。これは従来手法に比べ、本手法は新規参加ノードの参加成功率が参加順序に依存しにくいことを意味している。以上のことから、本手法は従来手法と比較して参加成功率の向上が見込まれると考えられる。

#### 4.2.2 配信木の深さ

それぞれの  $R_b$  の下で、 $N_c$  の値を従来手法でも参加成功率が 100% となるような最小の値に設定した際の実験結果を表 1 に示す。具体的には、各回の実験で構築された配信木の深さの平均値について、100 回分の実験の平均と最大、最小の値を示す。この結果から、配信木の深さは双方向通信可能なノードがいかなる割合においても浅くなるこ

表 1 配信木の深さの平均値  
Table 1 Averages of distribution tree depth.

	$R_b(\%)$	10	20	30	40	50	60	70	80	90
平均	従来法	16.8	17.6	19.4	20.5	15.3	23.8	18.2	15.3	13.1
	提案法	3.70	4.63	5.54	6.56	7.15	9.49	10.4	11.1	11.4
最大	従来法	40.8	38.0	42.6	42.7	27.2	47.3	30.6	19.5	15.1
	提案法	4.46	5.28	6.35	7.67	7.65	11.4	11.4	11.4	11.4
最小	従来法	4.88	6.38	6.02	5.89	6.36	6.63	8.55	8.39	12.1
	提案法	2.89	3.45	3.74	3.84	4.47	4.67	6.27	6.27	11.4

とを確認した。これにより参加オーバーヘッドやストリーミング遅延の減少が見込まれる。ただし、この実験において PUSH セッションを確立するためのオーバーヘッドは考慮していない。

### 5. まとめ

本論文では、配信木を利用した P2P ストリーミング配信における NAT 越えを考慮した配信システムの検討を行った。まず、従来手法である PeerCast の配信木構築手法と、NAT を考慮した場合の配信木への影響について示した。次に提案手法として、セッションの張り方の工夫により配信木の最適化を図る手法を示し、実験を通して従来手法より NAT の影響を軽減する特徴を持つことを確認した。今後の課題としては、ネットワークシミュレータを用いた提案プロトコルの参加オーバーヘッドの評価が考えられる。

### 参考文献

- [1] 安原健介, 日下卓也: PeerCast におけるデータロスのない動的な自律分散型経路変更機能の設計と実装, 信学論 D, Vol.J92-D, No.9, pp.1677-1681 (2009).
- [2] 安原健介, 甲本卓也: PeerCast における動的な自律分散型経路変更機能の設計と実装, 信学技報, Vol.108, No.258, pp.7-12 (2008).
- [3] 安原健介, 甲本卓也, 船曳信生, 杉山裕二: PeerCast における経路の動的変更機能の実装, 信学技報, Vol.107, No.36, pp.27-32 (2007).
- [4] 鈴木秀和, 宇佐見庄五, 渡邊 晃: 外部動的マッピングにより NAT 越え通信を実現する NAT-f の提案と実装, 情報処理学会論文誌, Vol.48, No.12, pp.27-32 (2007).
- [5] D'Acunतो, L., Pouwelse, J.A. and Sips, H.J.: A Measurement of NAT & Firewall Characteristics in Peer to Peer Systems, 15th ASCI (Advanced School for Computing and Imaging) Conference, pp.1-5 (2009).
- [6] Rosenberg, J., Weinberger, J., Huitema, C. and Mahy, R.: STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs), RFC 3489IETF (2003).
- [7] 笠野英松: ネットワーク・スーパーテキスト, 技術評論社 (2003).
- [8] PeerCast, available from (<http://peercast.org/>).
- [9] 菅井文郎, 鮫島慎治, 岡崎直宣: P2P ストリーミングにおける NAT 越えを考慮した配信システムの検討, 電気関係学会九州連合大会, No.13-2P-04, pp.1-2 (2011).
- [10] 鮫島慎治, 岡崎直宣: P2P ストリーミングにおける NAT 越えを考慮した配信木構築プロトコルの検討, 電気関係学会九州連合大会, No.13-2P-05, pp.1-2 (2011).



山場 久昭 (正会員)

1988年東京工業大学工学部化学工学科卒業。1990年東京工業大学大学院総合理工学研究科化学環境工学専攻修士課程修了。同年花王(株)入社。1993年より宮崎大学工学部助手。2007年より同助教。2010年宮崎大学大学院博士後期課程システム工学専攻単位取得満期退学。生産システム設計・運用の計算機支援に関する研究に従事。博士(工学)。化学工学会, 人工知能学会, 計測自動制御学会各会員。



鮫島 慎治

2010年宮崎大学工学部情報システム工学科卒業。2012年宮崎大学大学院情報システム工学専攻修士課程修了。アプリケーションネットワークに関する研究に従事。



池田 匡視

2011年宮崎大学工学部情報システム工学科卒業。現在, 宮崎大学大学院情報システム工学専攻修士課程在学中。アプリケーションネットワークに関する研究に従事。



油田 健太郎 (正会員)

2003年宮崎大学工学部情報システム工学科卒業。2005年宮崎大学大学院工学研究科情報工学専攻博士前期課程修了。2006年熊本県立大学総合管理学部助手。2009年宮崎大学大学院工学研究科システム工学専攻博士後期課程修了。同年大分工業高等専門学校助教。2012年より同講師。モバイルネットワークに関する研究に従事。博士(工学)。電子情報通信学会会員。



岡崎 直宣 (正会員)

1986年東北大学工学部通信工学科卒業。1991年東北大学大学院工学研究科電気及び通信工学専攻博士後期課程修了。同年三菱電機株式会社入社。2002年より宮崎大学工学部助教授。2007年同准教授を経て2011年同教授。通信プロトコル設計, ネットワーク管理, 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。博士(工学)。電子情報通信学会, 電気学会, IEEE各会員。



富田 重幸 (正会員)

1971年京都大学工学部衛生工学科卒業。1973年京都大学大学院修士課程修了。1977年同大学院博士課程化学工学専攻修了。同年東京工業大学助手を経て, 1993年宮崎大学工学部教授, 現在に至る。主として, 生産情報システム, 離散事象システム, 知識情報処理に関する研究に従事。工学博士。人工知能学会, 計測自動制御学会, 化学工学会各会員。



朴 美娘 (正会員)

1983年漢陽大学工学部電子工学科卒業。同年漢陽大学工学部助手。1993年東北大学大学院工学研究科情報工学専攻博士後期課程修了。同年東北大学電気通信研究所助手。1994年三菱電機株式会社入社。2010年より神奈川工科大学情報学部教授。通信プロトコル設計, ネットワークセキュリティ等の研究に従事。博士(工学)。電子情報通信学会, 日本セキュリティ・マネジメント学会各会員。