

## 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検討

鮫島 慎治<sup>†1</sup> 菅井 文郎<sup>†1</sup> 池田 匡視<sup>†1</sup>  
岡崎 直宣<sup>†1</sup> 朴 美娘<sup>†2</sup>

P2P ストリーミングにおいて、配信木と呼ばれる木構造のトポロジーを構成しコンテンツを配信する手法がある。この手法は、ノードの負荷を分散し配信することで、高いスケーラビリティをもつ。しかし、現状では多くのノードが NAT を介して接続しているため、双方向通信不可能なノードが存在する。このようなノードの割合が多いほど、配信木への参加成功率やオーバーヘッドへの影響がある。本稿では、配信木に参加済みノードから新規参加ノードの方向にセッションを張ることによって、この影響を抑える手法についての検討を行う。配信木を比較する実験により、提案手法は NAT を考慮した環境でも配信木への影響が少なくなることを確認した。

### A method for distribution tree construction on P2P streaming system in consideration of NAT traversal

SHINJI SAMESHIMA,<sup>†1</sup> FUMIO SUGAI,<sup>†1</sup> MASAMI IKEDA,<sup>†1</sup>  
NAONOBU OKAZAKI<sup>†1</sup> and MIRANG PARK<sup>†2</sup>

A method for distributing streaming contents on P2P using application level multicasting by the distribution tree is known to be suitable because of its high scalability. However, because the node behind NATs cannot create a session to other nodes, in case many P2P nodes are behind NATs, the tree construction success rate falls and an overhead of tree construction procedure increases. In this paper, we propose a new distribution tree construction method in consideration of NAT for distributing streaming contents on P2P. In this method, in addition to “pull-session” that is usually used, according to the type of NAT we introduce a “push-session” to create session from nodes behind NATs to participate in the distribution tree.

#### 1. はじめに

近年のメディアコンテンツのストリーミングサービスの需要の高まりは、配信サーバへの負荷の増大をもたらしている。この負荷を軽減する技術として、P2P(Peer to Peer) を利用したストリーミング配信がある。P2P にはノードで受信データを複製して他の複数のノードに送信する ALM(Application Level Multicast) と呼ばれる技術があり、ALM を利用した P2P ストリーミングには、配信木と呼ばれる木構造のトポロジーを構成してコンテンツを配信する手法がある<sup>1)-3)</sup>。この手法はノードの負荷を分散し配信することで高いスケーラビリティをもち、さらに配信元の負荷を減少することが可能である。しかし、現状では多くのノードが NAT(Network Address Translation) を介して接続しているため、双方向通信不可能なノードが存在する。配信木の構築アルゴリズムは双方向通信可能なノードが参加することを前提として設計されているため、このようなノードの割合が多いほど、配信木への参加成功率やオーバーヘッドへの影響がある。本稿では、NAT によるこのような影響を抑える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従来手法として PeerCast<sup>1)-3)8)</sup> がある。ここでは、その概要と NAT を利用したノードが多い場合の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接続している子ノードのアドレス情報を格納した子ノードリストを持っている。これにより、配信木の配信元となる根ノードを探索し、子ノードリストを取得し続けることで参加している全てのノードに接続要求を送ることができる。まず、新規参加ノードは根ノードを探索する。その後、根ノードから幅優先探索を行い、ノードを選択して接続要求を送信する。許可された場合はそのノードから配信木に参加する。拒否された場合は子ノードリストをもらい、次に幅優先探索で優先度が高いノードを選択し、接続要求を送信する。以上を繰り返して新規参加ノードは配信木の親ノードを決定し、配信木に参加する。

<sup>†1</sup> 宮崎大学  
University of Miyazaki

<sup>†2</sup> 神奈川工科大学  
Kanagawa Institute of Technology

## 2.2 NAT が配信木に与える影響

NAT(Network Address Translation) 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ものであり、NAT テーブルでポートマッピングを行うことで、NAT 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できる技術である<sup>7)</sup>。ただし、この NAT を利用したノードはインターネット側から NAT 内部のノードに接続できない特徴があり、これは P2P 通信を行う上で大きな障害となる。実際にファイル共有 P2P アプリケーションでは、ノードの 70%~80% が NAT を利用しているという研究報告<sup>5)</sup> もあり、P2P 通信を前提とした配信木構築手法にとって大きな影響が出ると考えられる。

配信木構築手法は根ノードから配信木の下流に向かって接続可能なノードを探索する。探索終了後、新規参加ノードは配信木に参加する。以後このノードへも新たなノードが接続できるため、配信木は高いスケーラビリティを持つ。しかし、NAT を利用する新規参加ノードが配信木に参加した場合、このノードへ新たなノードが接続できなくなるため、配信木のスケーラビリティが損なわれると考えられる。スケーラビリティを損なう要因として、以下のことが考えられる。

### ● 参加成功率の低下

配信木に参加しているノードの中で NAT を利用しているノードの割合が多い場合には、双方向通信可能なノードの帯域が全て NAT を利用しているノードに占有され、その後の新規参加ノードは配信木に参加できなくなる。

### ● 参加オーバーヘッドの増大

配信木に参加するノードが NAT を利用していた場合、新規参加ノードは接続できるノードに偏って接続されるため、配信木が深くなる傾向がある。これにより、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

配信木が深くなる傾向は、根ノードからのストリーミング遅延の増大をもたらす。これは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では大きな問題になると考えられる。

## 3. 提案手法

上記の問題を解消するために、ノード間の接続環境を考慮しセッションを使い分け、新規参加ノードの配信木への参加機会を増やす手法について提案する。まず NAT の特徴として、

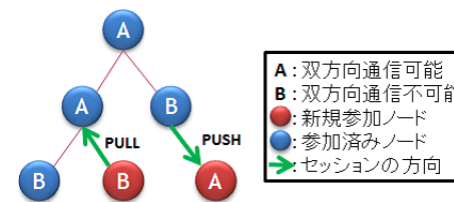


図 1 セッションの種類

NAT は内部ノードから外部ノード方向に接続を行うことはできる。このため、新規参加ノードが双方向通信可能ならば親ノードが NAT を利用するノードでも通信できる。通常のセッションを PULL セッションというのに対して、これを PUSH セッションという(図 1)。通信するノード毎にこの 2 つのセッションを使い分け、さらに配信木の最適化を目指す配信木構築手法について提案する。また、NAT の有無を区別する単純な手法だけではなく、既存の NAT 越え手法であるホールパンチング<sup>4)5)</sup> について考慮した手法も合わせて提案する。これにより、配信木における双方向通信可能なノードの割合を増やし、参加成功率と配信木の深さの改善を行う。

### 3.1 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 2 に示す。なお、本手法は前提として配信木の親ノードは接続している子ノードのアドレス情報と双方向通信の可否情報を格納した子ノードリストを持っているものとする。

#### (1) 配信木の根ノードの検索

新規参加ノードは DHT(Distributed Hash Table) 等を利用し、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を取得する。アドレスを取得後、根ノードを親候補リストに格納する。

#### (2) 親候補ノードへの接続要求の送信

本手法の親候補ノードの選定基準は、通信可能かつ出来るだけ配信木に浅いところに接続できるノードとしており、このための親候補リストの操作を(4)で行っている。ここでは、親候補リストの dead フラグ(一度親候補ノードとして選択済みであることを示す)の立っていないノードで一番先頭に記述されているノードを親候補ノードとする。この親候補ノードに対して、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

接続要求を送った親候補ノードから接続を許可された場合は(5)を実行し、拒否され

た場合は (4) を実行する。

- (4) 親候補ノードからの子ノードリストの取得  
 接続を許可されない親候補ノードが持つ、配信木の参加時刻でソートされた子ノードリストを取得する。新規参加ノードは子ノードリストを表 1 の優先度でソートし、親候補リストに追記する。その後、前処理で親候補になったノードは親候補リスト上で dead フラグを立て、再び親候補にならないようにし、(2) を繰り返す。
- (5) ストリーミングの開始  
 新規参加ノードは、親候補ノードを配信木の親ノードと決定して、ストリーミングデータの受信を開始する。この際、親ノードは新規参加ノードを自身のアドレスリストに追加する。

上記 (2) において新規参加ノードが PUSH セッションを選択する場合、既存手法と同じようなオーバーレイネットワーク構成や通信プロトコルでは、NAT の制約により PUSH セッションを確立することができない。このため PUSH セッションを実現するための通信プロ

トコルが必要になる。このプロトコルについて 3.2 で述べる。

### 3.2 セッション制御プロトコル

本手法は、新規参加ノードと親候補ノードの接続環境を考慮しセッションの選択を行う。PULL または PUSH セッションの制御を適切に行い、本手法を実現するためのプロトコルについて述べる。

### 3.3 前提条件

本プロトコルは、ライブストリーミング配信を想定してプロトコルを設計した。そのため通信は、全て UDP によるものとした。また、PUSH セッションを実現するために新たなサーバをシステムに導入しないものとする。これにより P2P の耐故障性を維持することができる。

#### 3.3.1 オーバレイネットワークの構成

本手法は、配信木を構成しストリーミングサービスを提供する P2P アプリケーションネットワーク (NW) とコンテンツ検索とセッション制御を行う P2P 制御 NW と呼ぶ、2 つのオーバーレイネットワークを利用して動作する (図 3)。本手法が動作するには、ノードがあらかじめ P2P 制御 NW に参加する必要がある。この際にノードはユニークな ID を取得する。ノードからストリーミングサービスの要求が起きた時は、初めに P2P アプリケーション NW 上に P2P 制御 NW と同じ ID を持つノードを新たに作成する。このノードは自身と配信木に参加しているノードの接続環境を考慮し、配信木の接続するノードを決定する。接続するノードが NAT を利用している環境であるならば、セッション制御メッセージを P2P 制御 NW のノードを介して転送し、自身と接続するノードでの適切なセッションを確立する。

なお、同図で示すように本稿では、P2P 制御 NW をスーパーノード型 P2P で構築する場合のプロトコルについて示している。しかしながら、その他の P2P 制御 NW の構成方法でも提案手法は実現可能だと考える。以降、P2P 制御 NW がスーパーノード型 P2P である場合のプロトコルの動作について示す。

本手法は、配信木の根ノードから配信木への参加時刻、新規参加ノードと参加済みノードの接続環境を考慮し、幅優先探索で親候補ノードを選出する。親候補ノードの選出後、新規参加ノードは自身と親候補ノードの接続環境を考慮し、セッションの選択を行う。PULL セッションを選択した場合、通信プロトコルは P2P アプリケーション NW のみで処理が終了し、親候補ノードに接続要求の送信を行う。また、PUSH セッションを選択した場合、通信プロトコルは P2P 制御 NW を介して、親候補ノードにセッション制御メッセージを転

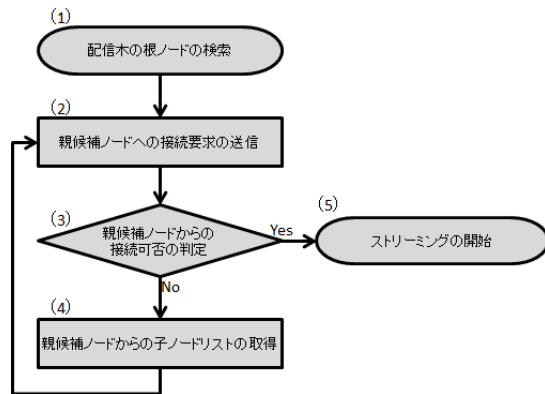


図 2 配信木参加手順

新規参加ノード	親候補ノード	優先度
可	否	高
否	可	
可	可	低

表 1 双方向通信の可否による接続優先度

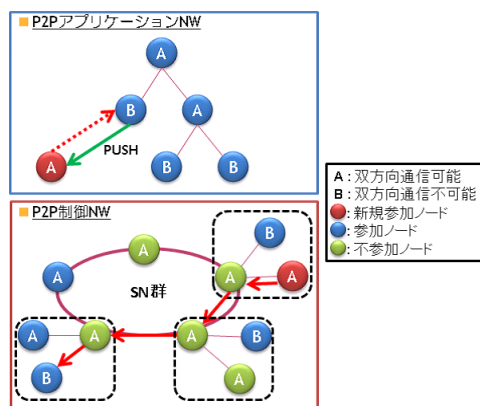


図 3 NW の構成と基本動作

送する。セッション制御メッセージを受信した親候補ノードは、新規参加ノードに対して PUSH セッションを行い処理が終了する。

### 3.3.2 セッション制御

プロトコルが動作するためには、新規参加ノードと親候補ノードの接続環境を考慮し、適切にセッション制御する必要がある。本手法ではセッション制御のために、ノードの接続環境を接続特性と呼ばれる形で定義し、その接続特性に基づいたセッションの選択を行っている。本節では、この接続特性の定義とセッションの選択について述べる。

#### 3.3.2.1 接続特性の定義

NAT を利用したノードや利用していないノードが混在する環境において、本手法がセッション制御を適切に行うためにノードの接続特性を定義する。またこれに加えて、ここで既存 NAT 越え手法であるホールパンチング<sup>4)5)</sup>を考慮した定義についても合わせて考える。

まず、ホールパンチングについて図 4 で示し、その説明について述べる。NAT はノード A からノード B にアクセスするとき、NAT 内部でソケットアドレスが 1 対 1 で対応付けられる。この対応表を NAT テーブルと言い、これにより NAT はアドレス変換を行い、ノード A とノード B は互いのノードと 1 対 1 で通信することができる。ホールパンチングは、この NAT テーブルのマッピングされたアドレスを何らかの方法で通信要求のあるノードに通知し、そのマッピングアドレスで NAT を越えて通信することが可能になる手法である。

ただし、現在普及している全てのブロードバンドルータがホールパンチングを許可してい

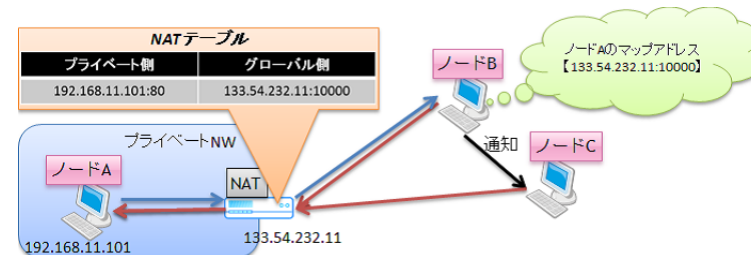


図 4 ホールパンチング

るわけでない。ノードが利用するブロードバンドルータは、NAT テーブルのポリシーにより、RFC 3489 で定義される 4 種類の NAT が存在する<sup>6)</sup>。この 4 つの NAT と NAT テーブルのポリシーについて下記に示す。

- Full Cone NAT

内部ノードからルータの外部にパケットを送る時に、内部ノードのポートとルータのポートが NAT テーブルに 1 : 1 でマッピングされる方式。マッピングアドレスと通信することで、どのような外部ノードでも内部ノードと通信することが可能である。

- Restricted Cone NAT

マッピング方式は Full Cone NAT と同様だが、内部ノードとの通信が可能な外部ノードは、マッピングの際に送信したパケットの宛先 IP アドレスのノードのみに制限するもの。

- Port Restricted Cone NAT

基本的には Restricted Cone NAT と同様に振舞うが、内部ノードとの通信制約にポート番号を含むもの。

- Symmetric NAT

他の 3 つのマッピング方式とは全く異なり、内部ノードの同一のポートからパケットが送信されたとしても、異なるルータのポートにマッピングされるもの。このため、一度送信したパケットの宛先ノード以外からは通信を行うことができない。

これらのうち、ホールパンチングを適用できる NAT は、Full Cone NAT だけである。その他の NAT は、P2P 制御 NW を利用しなければ不特定多数と通信することができない。このため、ここではノードの接続特性の class を表 2 のように定義する。

なお、これらの接続特性は、ノードが P2P 制御 NW に参加する時に STUN(Simple Traver-

接続特性	NAT タイプ
class 1	NAT 無し
class 2	Full Cone NAT
class 3	Restricted Cone NAT , Port Restricted Cone NAT , Symmetric NAT のいずれか

表 2 ノードの接続特性

		親候補ノードの接続特性		
		class 1	class 2	class 3
新規参加ノードの接続特性	class 1	PULL	PULL	PUSH
	class 2	PULL	PULL	PUSH
	class 3	PULL	PULL	×

図 5 セッションの選択

sal of UDP through NATs) アルゴリズム<sup>6)</sup> を適用し、取得するものとする。

### 3.3.2.2 セッションの選択

新規参加ノードと親ノードの接続特性を考慮したセッションの選択について図 5 に示す。ここで、例えば新規参加ノードの接続特性が class 1 または class 2 で親ノードが class 3 の時に PUSH セッションを選択し、それ以外は PULL セッションを選択する。ただし、同図に記されているセッションは、最終的なセッションの方向であって、同じセッションであっても処理内容が異なっている場合がある。これについて、それぞれのセッションの意味を下記に示す。

- 無処理の PULL セッション  
新規参加ノードと親候補ノードの接続特性の組み合わせが (class 1, class 1) または (class 3, class 1) の時、親候補ノードのソケットアドレスを用いて、PULL セッションで通信することが可能。
- ホールパンチングを適用するための PULL セッション  
新規参加ノードと親候補ノードの接続特性の組み合わせが (class 2, class 1) の時、親候補ノードのソケットアドレスを用いて、PULL セッションで通信することが可能。な

お、親候補ノードが通信を許可した場合、親候補ノードの子ノードリストに格納されるソケットアドレスは、新規参加ノードにマッピングアドレスである。これにより、class 2 の新規参加ノードにホールパンチングを適用できるようになる。

- ホールパンチングを利用した PULL セッション  
新規参加ノードと親候補ノードの接続特性の組み合わせが (class 1, class 2), (class 2, class 2) または (class 3, class 2) の時、親候補ノードのソケットアドレスを用いてホールパンチングを行い、PULL セッションで通信することが可能。
- 通常処理の PUSH セッション  
新規参加ノードと親候補ノードの接続特性の組み合わせが (class 1, class 3) の時、親候補ノードの ID を用いて親候補 P2P 制御 NW を介してセッション制御メッセージを転送し、PUSH セッションにより通信することが可能。
- ホールパンチングを利用した PUSH セッション  
新規参加ノードと親候補ノードの接続特性の組み合わせが (class 2, class 3) の時、親候補ノードの ID を用いて親候補 P2P 制御 NW を介してセッション制御メッセージを転送し、PUSH セッションにより通信することが可能。なお、セッション制御メッセージには、P2P 制御 NW を介している最中に新規参加ノードのマッピングアドレスが格納される。親候補ノードは、このマッピングアドレスでホールパンチングにより通信することができる。加えて、親候補ノードが通信を許可した場合、親候補ノードの子ノードリストに格納されるソケットアドレスは、新規参加ノードにマッピングアドレスである。これにより、class 2 の新規参加ノードにホールパンチングを適用できるようになる。

### 3.3.2.3 SCMT(Session Control Message Transfer)

セッションの選択で PUSH セッションが選択された場合、セッション制御メッセージを P2P 制御 NW を介して転送する。この通信プロトコルが SCMT であり、ホールパンチングを利用した PUSH セッション時の動作について図 6 で示し、下記に説明する。

- (1) 接続要求の転送をスーパーノードに依頼  
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) マッピングアドレスをクエリに格納  
N のスーパーノードは、N と直接通信することで N 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 N から受け取ったクエリに格納する。



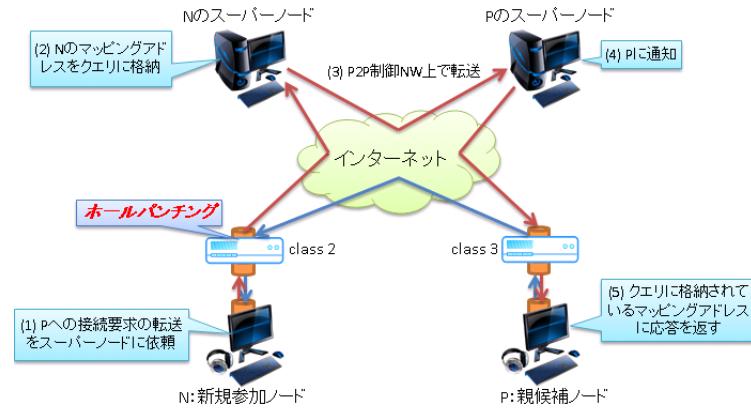


図 6 SCMT の動作例

(3) P2P 制御 NW 上でクエリの転送

クエリに格納されている ID を基に、P2P 制御 NW 上でクエリを転送する。

(4) P に通知

P のスーパーノードは、P の P2P 制御 NW 参加時の処理により、双方向通信可能である。転送されてきたクエリを P に通知する。

(5) マッピングアドレスへの応答

P は、クエリに格納されているマッピングアドレスを取り出し、そのマッピングアドレスに応答を返す。

なお、もう一つの PUSH セッション処理である通常処理の PUSH セッションの場合、処理番号 (2) で格納するアドレスが N のソケットアドレスがマッピングアドレスの違いのみである。

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 10000 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  $N$  とし、全ての新規参加ノードのうち双方向通信可能な割合を 1 割 ~ 9 割と変化させ、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 7 は  $N=5$  の時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無いことが分かった。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

図 8 は  $N$  を従来手法で参加成功率が 100 % となる時の値を用い、配信木の深さの平均値の平均と最大、最小を示している。なお、横軸の値はそれぞれ異なる  $N$  を用いて比較しているので相関はない。図 8 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。

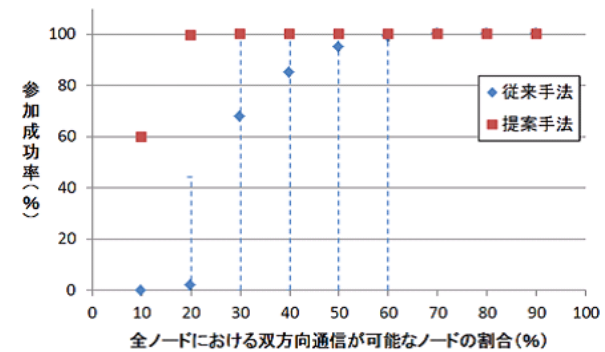


図 7  $N=5$  の時の参加成功率

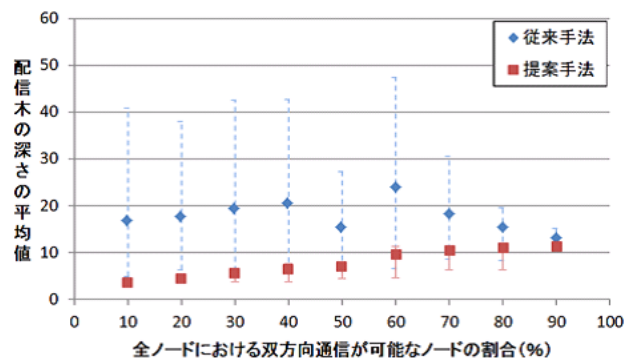


図 8 配信木の深さの平均値

## 5. ま と め

本稿では、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 の配信木構築手法を示し、NAT を考慮した場合の配信木への影響について示した。提案手法ではセッション制御により配信木の最適化を手法について示し、実験において従来手法より NAT の影響を軽減する特徴を確認した。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, 情処論, Vol.48, No.12, pp.27-32(2007) .
- 5) L.D'Acunto, J.A.Pouwelse, H.J.Sips: 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. andR.Mahy: STUN - Simple Traversal

of User Datagram Protocol(UDP) Through Network AddressTranslators(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, <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) .