

push型セッションを用いた P2Pストリーミング配信手法に関する研究

鮫島 慎治^{a)}・岡崎 直宣^{b)}

A Study of a Streaming Distribution System on P2P using a Push-session

Shinji SAMEISHIMA, Naonobu OKAZAKI

Abstract

A method for distributing streaming contents on P2P using over-layer 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.

Keywords: P2P, video streaming, NAT

1. はじめに

近年、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、それに伴って配信サーバ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)¹⁾を利用したストリーミング配信法である。具体的には、Over-Lay Multicast (以下、OLM)方式を採用し、受信データを複製してほかの複数のノードに送信する機能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ーを構成してコンテンツを配信する方法が注目されている。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軽減が可能である点にある。しかし、その一方で、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が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 Translator, 以下NAT)²⁾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必須の要件であると考えられる。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提案を行う。

2. 従来手法

配信木を利用したP2P型ストリーミングの代表的な手法の一つとしてPeerCast^{3,4,5)}がある。また、NATを考慮したP2Pアプリケーションで適用されるNATを利用した端末へのインターネット側からのアクセス (以下、NAT越え)を実現する代表的な手法としてホールパンチング²⁾がある。

ここでは、まず、配信木構築手法の概要、およびNATを利用している端末が多い場合に生じる問題点について示す。その後、ホールパンチング^{6,7)}の概要について示す。

2.1 配信木構築手法

従来手法の配信木構築手法では、親ノードは「自身に接続している子ノードのソケットアドレスのリスト」(子ノードリスト)を必ず持つ。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全てのノードのソケットアドレスが取得可能、すなわち、配信木に既に参加している全てのノードに接続要求を送ることが可能である。

新規参加ノードは配信木に参加する際の手順として、まず根ノードのソケットアドレスを取得する。根ノードのソケットアドレスを取得後、根ノードから始めて配信木の上流から下流に辿って親候補ノードを選択し、接続要求を送信する。新規参加ノードは自身に対する接続を許可した親候補ノードを自身の親ノードとして配信木に参加する。

2.1.1 NATが配信木に与える影響

NATを利用している端末は双方向通信ができなくなる。これは、双方向通信を前提とするP2Pでは大きな障害となると

a) 情報システム工学専攻大学院生

b) 情報システム工学科教授

考えられる。従来手法では、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してきたが、NATにより双方向通信ができない新規参加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新規参加ノードが接続できない。これに起因する問題点として、具体的には次のようなことが予想される。

- **参加成功率の低下**

配信木に参加するノードの中でNATを利用しているノードの割合が高い場合、双方向通信可能なノードの帯域を消費し尽してしまうと、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に親ノードになれないノードが増えることにより、不要な接続要求が発生し、参加オーバーヘッドが増大する。

- **ストリーミング遅延の増大**

NATを利用しているために親ノードになれないノードが増えることにより、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）が増大すると考えられる。これはリアルタイム性を求められる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。

実際、ファイル共有 P2P アプリケーションを利用している端末の 70%~80%が NAT を介してインターネットに接続しているという研究報告⁷⁾もあり、この問題への対応は必須の要件と考える。

2.2 ホールパンチング

NATがアドレス変換を行うためには、プライベートネットワーク内の端末（以下、内部端末）の「プライベートIPアドレスとポート番号の組」（ソケットアドレス）と、NAT機能付きブロードバンドルータ（以下、ルータ）の「グローバルIPアドレスとポート番号の組」（マッピングアドレス）との対応付けが必要になる。NATはこの対応付けをNATテーブルと呼ばれる対応表で管理している。

ホールパンチングとは、ある内部端末のソケットアドレスに対応付けられたマッピングアドレスを当該の端末に対して通信要求を送りたいと考えている端末（以下、外部端末）に通知し、そのマッピングアドレスを利用することによって、外部端末が内部端末とNATを越えて通信することを可能にする手法である（図1）。

ただし、現在普及している全てのルータがホールパンチングを許可しているわけではない。NATはそれぞれが用いているNATテーブルのポリシーにより、四種類に分類されること

がRFC 3489⁸⁾で規定されている。それらのうち、ホールパンチングを適用できるNATは内部端末のソケットアドレスと対応付けられたマッピングアドレスを用いて通信することに制約を持たないFull Cone NATだけである。

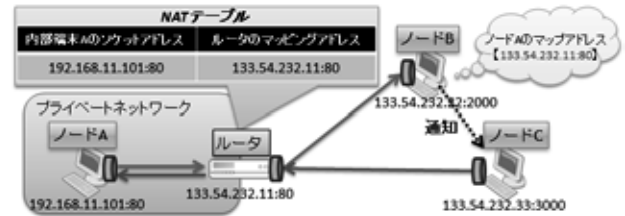


図1. ホールパンチング。

3. 提案手法

2.1.1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を利用したノードであっても、それが親ノードになれる手法を提案する。なお、本論文ではP2Pの耐故障性を維持するためにPUSHセッションを実現する上で新たなサーバをシステムに導入しない方針をとる。また、従来方式がライブストリーミングを対象としているため、本手法も対象とするストリーミングはライブストリーミングとした。

3.1 提案手法の概要

本手法は「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「配信木構築に利用するオーバーレイネットワーク」（以下、P2P制御NW）で動作する。以上を踏まえ、本手法の基本的なアイデアについて図2を用いて説明する。

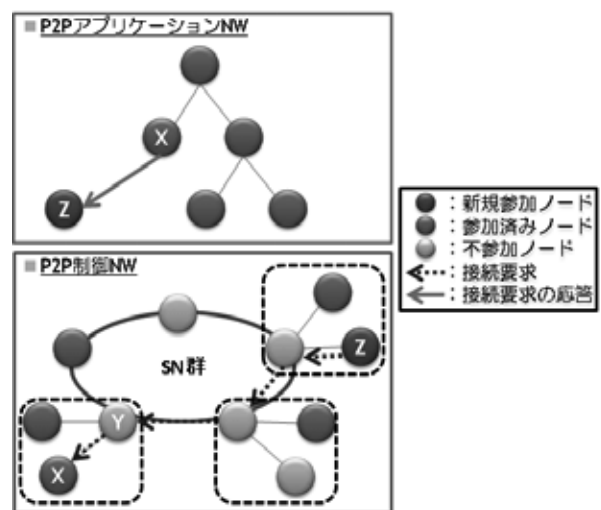


図2. 基本的なアイデア。

まず、NATを利用しているノードXは動画を視聴する前に、

あらかじめノードYに接続し、P2P制御NWに参加する。この参加処理の通信からノードYはノードXのマッピングアドレスを取得し、ノードXへいつでも通信可能な状態にしておく。これにより、ノードXが親候補ノードとなったとき、新規参加ノードZが接続要求をP2P制御NWを介してノードYへ、ノードYを介してノードXへ転送す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とのセッションを確立し、ストリーミング配信を開始する。従来手法で用いられている新規参加ノードから親候補ノードに向かってアクセスするセッションをPULLセッションと呼ぶのに対して、親ノード側から新規参加ノードに接続するセッションをPUSHセッションと呼ぶことにする(図3)。

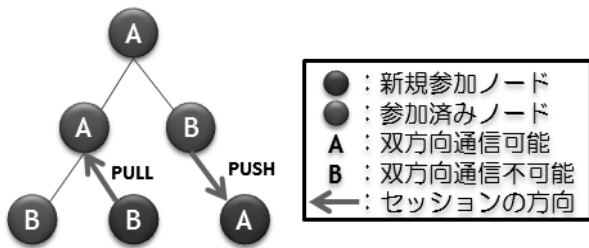


図3. セッションの種類

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。例えば、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、従来手法と同様にPULLセッションが可能になることPUSHセッションを用いる場合、P2P制御NWを介して接続要求を送る際にオーバーヘッドがかかることの二つを考えると、この種のノードでは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接続環境に応じて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の分類と親子双方のノードのクラスの組み合わせに応じたセッションの選択の手続きを導入する。その詳細については3.4節で述べる。なお、本手法でのホールパンチングはUDPホールパンチングを用いるため、通信は全てUDPによるものとした。

3.2 ノードの分類

本手法では、ノードの接続環境に応じてノードを三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class1、Full Cone NATを利用しているノードをclass2、それ以外のNATを利用しているノードをclass3と呼ぶ。class2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、class1とclass2のノードが双方向通信可能なノード、class3のノードは双方向通信不可能なノードである。

ノードがP2P制御NWに参加する際に、STUNアルゴリズム[5]を適用し、NATを利用しているのか、利用している場合にどの種類のNATを使用しているのかの情報を取得し、それに基づいて当該ノードのクラスが特定される。

3.3 SCMT プロトコル

PUSHセッションを確立するためには、NATを利用しているために双方向通信ができない親候補ノードへいかかにして接続要求を送るかが問題である。本論文では、セッション制御メッセージをP2P制御NWを介して親候補ノードへ転送することによりこの問題を解決する。この時に使用するプロトコルをSession Control Message Transfer(以下、SCMT)プロトコルと呼ぶ。このプロトコルのもとでのセッション確立までの手順を図4を用いて簡単に説明する。

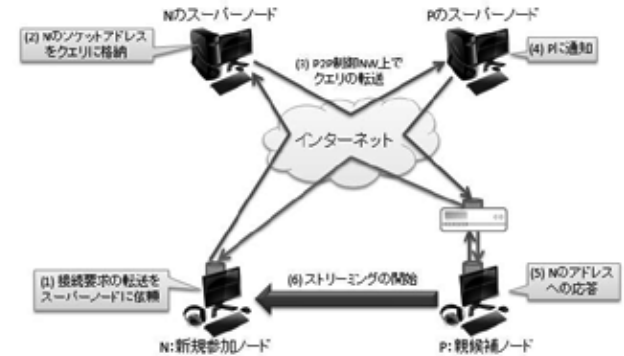


図4. SCMT プロトコルの手順

(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に接続するためのアドレス(NがNATを利用していないのであればソケットアドレス、NATを利用しているのであればマッピングアドレス)を知ることができる。このアドレスをNから受け取ったクエリに格納する。

(3) P2P制御NW上でクエリの転送

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、Pとの間で双方向通信が可能となっている。そ

ここで、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アドレスを取り出し、そのアドレスに応答を返す。

3.4 接続環境に応じたセッションの選択

ここでは、新規参加ノードと親候補ノードそれぞれの接続環境の組み合わせに応じてPULLセッションとPUSHセッションのどちらを選択するのかの選択基準と新規参加ノードと親候補ノードとの間でセッションを確立するまでの手順について述べる。親候補ノードがNATを利用している場合、基本的にはFull Cone NAT以外はホールパンチング、その他のNATはP2P制御NWを介した通信を行うという方針をとる。本手法では新規参加ノードと親候補ノードとのクラスの組み合わせを六つのtypeに分類(表1)し、そのtypeに応じて接続時には異なる処理を行う。具体的には次の通りである。

ノードのクラス	親候補ノード			
	class 1	class 2	class 3	
新規参加ノード	class 1	type 1	type 3	type 5
	class 2	type 2	type 4	type 6
	class 3	type 1	type 3	N/A

表1. セッション確立手段.

- **type 1**: 基本PULLセッション
親候補ノードが双方向通信可能なので、そのソケットアドレスに対して新規参加ノードが接続要求を直接送信し、PULLセッションの確立を試みる。親候補ノードが通信を許可し、親ノードになった場合はPULLセッションが確立される。この時、親ノードの子ノードリストには新規参加ノードがclass 1の場合はソケットアドレスが格納され、class 3の場合はIDが格納される。
- **type 2**: ホールパンチングの準備を伴うPULLセッション
基本的にtype 1と同様である。親候補ノードの子ノードリストに格納されるアドレスが新規参加ノードのマッピングアドレスである点で異なる。
- **type 3**: ホールパンチングを利用したPULLセッション
基本的にtype 1と同様である。親候補ノードのアドレスがマッピングアドレスであり、ホールパンチングを利用して接続が行われる点で異なる。
- **type 4**: ホールパンチングの準備を伴う、ホールパンチングを利用したPULLセッション
基本的にtype 3と同様である。親候補ノードの子ノードリストに格納されるアドレスが新規参加ノードのマッピングアドレスである点で異なる。

- **type 5**: 基本PUSHセッション
P2P制御NWを介して親候補ノードのID宛にセッション制御メッセージを送る。このセッション制御メッセージには親候補ノードがPUSHセッションを確立できるように新規参加ノードのソケットアドレスを格納する。セッション制御メッセージを受け取った親候補ノードは通信を許可する場合にはセッション制御メッセージに格納されているアドレスとの間にPUSHセッションを確立する。この時、親候補ノードの子ノードリストに新規参加ノードのソケットアドレスが格納される。
- **type 6**: ホールパンチングの準備を伴うPUSHセッション
基本的にtype 5と同様である。セッション制御メッセージに格納される新規参加ノードのアドレスと親ノードの子ノードリストに格納されるアドレスがマッピングアドレスである点で異なる。

3.5 配信木構築アルゴリズム

ここでは、本手法における配信木構築アルゴリズムについて述べる。なお、アルゴリズムを区別するために、3.5.1節で述べるものを「基本選択法」と呼び、3.5.2節で述べるものを「最適選択法」と呼ぶことにする。

3.5.1 基本選択法

基本選択法における配信木参加手順を図5を用いて簡単に述べる。

本手法では、配信木の親ノードは「自身に接続している子ノードのIDとアドレス、双方向通信の可否の三つの情報を格納したリスト」(子ノードリスト)を必ず持つものとする。そして、本手法も従来手法と同様に、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に従って接続要求の送信を行う。探索は標準的なopenとclosedのリストを用いる方法で行う。幅優先探索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するためである。一方、同じノードの子ノード間での親候補ノードを加える際の優先順位は次の通りである。

- ① 新規参加ノードが双方向通信できない場合は双方向通信可能なノードを優先し、双方向通信できる場合は双方向通信不可能なノードを優先する。
- ② ①で同じ順位場合は配信木の参加時刻の早いものほど優先する。
①の採用理由は、(a) 双方向通信できないノードの親ノードは双方向通信できる親ノードにせざるを得ないことと、(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なくてもできなくても構わないが、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である。②の採用理由は、文献⁹⁾で述べられているように、早く配信木に参加することにより、より長くサービスを受けているノードは今後も配信木から離脱しにくく、配信木における離脱のピアの影響範囲や再参加回数を減らすことが可能であることから、参加時刻の早いノードほど優先的に他ノードへサービスを行うことが適当であると考えたためである。

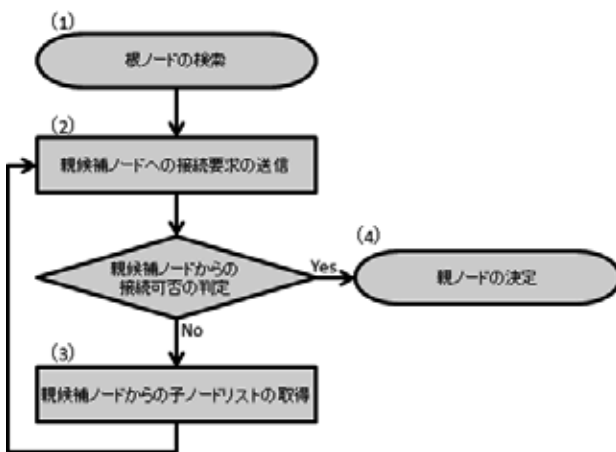


図5. 配信木への参加手順。

(1) 根ノードの検索

根ノードのアドレスは5.3節で述べたように、P2P制御NWの構成に基づいた取得方法で入手する。その後、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の先頭ノードを親候補ノードとして選択し、接続要求を送信する。接続が許可された場合は(4)へ、拒否された場合は(3)へ進む。

(3) 親候補ノードからの子ノードリストの取得

接続を拒否された場合、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、それを前述する優先度を用いて並び替えた上でopenの末尾に加える。さらに、親候補ノードをopenから削除し、closedに追加した上で(2)に戻る。

(4) 親ノードの決定

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、動画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。

3.5.2 最適選択法

ここでは、3.5.1節で述べた基本選択法により、さらに多く

のノードが配信木に参加可能な配信木構築アルゴリズムについて述べる。なお、最適選択法は基本選択法の親候補ノードの選出方法のみが異なっている。

基本選択法では、新規参加ノードが配信木の親を探索するとき、openの先頭ノードを親候補ノードとして選択し、接続要求の送信を行う。このopenは接続を拒否された親候補ノードの子ノードリストを5.7節で述べた優先度を用いて並び替えたものが次々に末尾に加えられる。こうすることで幅優先探索が機能し、配信木の深さを抑えることが可能になる。しかし、配信木にさらに多くのノードが参加できるようにする場合にはリストを加えた後でその都度openを優先度で並び替える方が良い。だが、これでは幅優先探索が機能しなくなり、配信木の深さが深くなってしまいう可能性がある。

そこで、最適選択法は「探索中の深さが変わるとき」にopenを3.5.2節で述べた優先度で並び替える処理を追加する。幅優先探索を行うとき、openに入っているノードの深さは「探索中の深さ」と「その深さ+1」のみである。つまり、探索中の深さが変わるときには、同じ深さのノードしかいない。このタイミングにおいて優先度で並び替える場合、幅優先探索を機能させ続け、同時により多くのノードが接続可能な配信木を構築可能である。

4. 評価

ここでは第2章で述べた従来手法と第3章で述べた基本選択法と最適選択法について評価実験を行う。評価基準はNATにより影響を受けると思われる配信木への参加成功率、オーバーヘッドとストリーミング遅延の四つとする。

4.1 方法

まず、P2P制御NWに2000個のノードを参加させる。次に、このノードの中からclass 1の根ノードを設置し、異なる接続条件を持つ999個の新規参加ノードをランダムな順序で参加させる。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5とし、表2のようにP2P制御NWにおけるクラスの分布を変化させて実験を行った。なお、同表は文献を参考に作成した。最後に、構築した配信木にストリーミングデータを流す。なお、実験は3回行い、その平均値を示す。

クラス	分布 (%)									
	10	20	30	40	50	60	70	80	90	100
class 1	10	20	30	40	50	60	70	80	90	100
class 2	10	9	8	7	6	4	3	2	1	0
class 3	80	71	62	53	44	36	27	18	9	0

表2. P2P制御NWにおけるクラスの分布。

4.2 結果

図6は参加成功率, 図7は制御オーバーヘッド, 図8はストリーミング遅延の実験結果を示している.

図6から提案手法は配信木への参加成功率を改善していることが確認できる. ただし, 基本選択法と最適選択法の優劣を確認することができなかった. しかし, 本来は最適選択法がより優れた性能を得られると考えられる.

図7から提案手法はオーバーヘッドが増加していることが確認できる. ただし, (a) ノード当りのオーバーヘッドが61Kバイト程度であること, (b) 配信木構築後のオーバーヘッドの差がないこと, (c) 一般的に, ストリーミングを行うコンテンツが数十から数百Mバイトであることを考慮すれば, この値は十分に許容できる値だと考えられる.

図8から提案手法はストリーミング遅延を改善していることが確認できる. これは, 配信木構築アルゴリズムにより, 配信木の深さを改善していたことが要因であると考えられる.

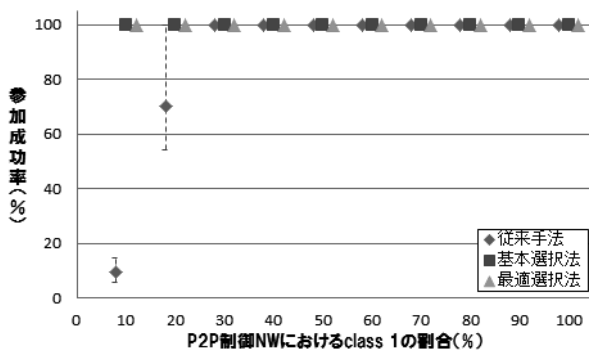


図6. 配信木への参加成功率.

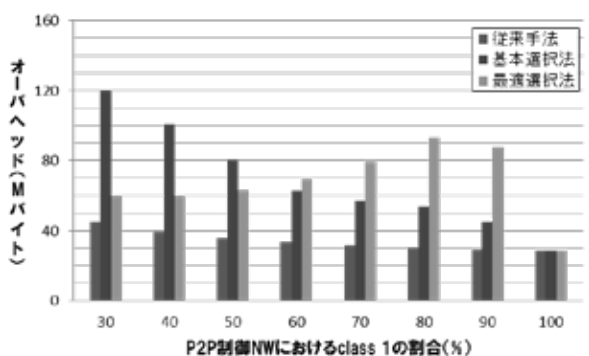


図7. オーバーヘッド.

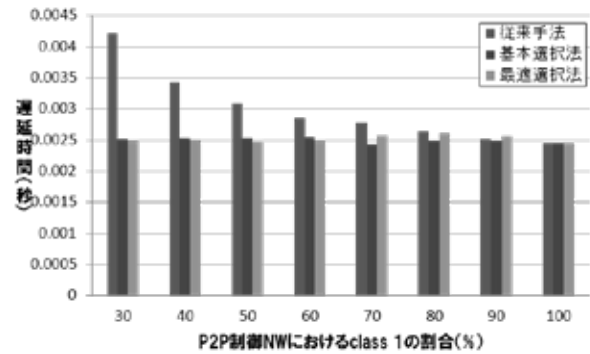


図8. ストリーミング遅延.

5. まとめ

本論文では, 配信木を利用した P2P ストリーミングにおける NAT 越えを考慮した配信システムの提案を行った. まず, 従来手法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, PUSH セッションと呼ぶ接続方法を導入することにより, NAT を利用したノードであっても, それが親ノードになることが可能な手法について示した. 実験において, 従来手法より NAT の影響を軽減することを確認した.

今後の課題としては, まず最適選択法が基本選択法と同様に大規模なパラメータにおいても NAT の影響を軽減できるかどうか調べることである. この軽減できることを確認した上で, 配信木に参加している NAT を利用しているノードの割合を考慮することで基本選択法と最適選択法を状況に応じて切り替える手法について評価することが挙げられる.

参考文献

- 1) 江崎 浩: P2P(ピア・ツー・ピア)教科書, インプレス R&D, 2008.
- 2) 笠野英松: ネットワーク・スーパーテキスト, 技術評論社, 2003.
- 3) 安原 健介, 甲本 卓也, 舩曳 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) 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.
- 5) 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.
- 6) 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.
- 7) L.D'Acunto, J.A.Pouwelse and H.J.Sips: A Measurement of NAT & Fire-wall Characteristics in Peer to Peer Systems, 15th

ASCI(Advanced School for Computing and Imaging)
Conference, pp.1-5, 2009.

- 8) Rosenberg,J., Weinberger,J., Huitema,C. and R.Mahy : STUN
- Simple Traversal of User Datagram Protocol Through
Network Address Translators, RFC 3489, 2003.
- 9) 杉野 博徳, 上野 貴之, 森野 博章 : ピアの滞在履歴
を考慮した P2P ストリーミング配信ツリーの高信頼化の
検討, 信学技報, Vol.107, No.221, pp.47-50, 2007.