

実ネットワークの特性を考慮した P2P リレーによる多チャンネル IP 映像配信の実現

勝 間 亮† 村 岡 洋 一†

本研究では、ユーザが複数の映像コンテンツを同時に受信し、テレビのようにザッピングを行いながらコンテンツの視聴を行うための効率的なコンテンツの配信方法、ならびに配信トポロジの構成方法を提案する。あらかじめ千人規模の配信実験を行うことで、一般的なインターネットユーザの上り帯域スループット値、接続方法といったネットワーク上の特性を取得した。この特性を用いることで、10000 人程度のユーザや、NAT 配下のユーザが半数以上存在するときも安定した配信を行えることを実現した。

Realization of multi channel IP media streaming using a P2P relay technology that considers properties of the real network

RYO KATSUMA† and YOICHI MURAOKA†

In this paper, we propose the technique to deliver the contents efficiently and to construct the topology while users can watch multi channel contents and surf channels. At first we conducted verification experiments for P2P Relay streaming of 1000 people scale and we got the properties of the real network, for example, the bandwidth and the way to connect to network. As the result of using this properties, we realized the stable streaming delivery to about 10000 users and to upwards of 50% of NAT users.

1. はじめに

ユーザが複数の映像コンテンツを同時に受信し、テレビのようにザッピングを行いながらコンテンツの視聴を行う IP 上の映像配信方法を「多チャンネル IP 映像配信」と呼ぶ。現在の IP 映像配信の多くはコンテンツ切り替えにおいてコンテンツの再生、確認、停止、を繰り返す「単一チャンネル IP 映像配信」方式であるが、ユーザの利便性を考えると今後は多チャンネル映像配信方式の需要が高まることが考えられる。本研究では、多チャンネル IP 映像配信を実現するための P2P リレーを用いた効率的なコンテンツの配信方法、ならびに配信トポロジの構成方法を提案する。

1.1 従来研究と問題点

大規模な映像配信を実現するための手段としては、IP マルチキャスト、CDN などが考えられる。しかし、インフラストラクチャの問題、キャッシュサーバの設置費用投資などの問題¹⁾ がそれぞれ挙げられている。そこ

で、本研究では ALM(Application Layer Multicast) を用いた実現を目指す。

ALM はこれまでに多くの研究が成されてきたが^{2)~4)}、実際のインターネット上での ALM の運用を試みると、十分に考慮されていない様々な現実的な問題が挙げられる。例えば、1. 上り帯域幅、2. エンドユーザのネットワークモデル、3. ホストのログ取得などの問題がある。

1. は、配信能力に関わる問題である。ALM の場合、他のホストから受信したデータを他のホストに再配信(リレー)することでオーバーレイネットワーク上に全ホストにデータを配信していく。しかし、ADSL に代表される上り帯域の狭い環境のホストは、FTTH 環境など広帯域環境のホストと比較するとその配信能力は小さいものとなり、コンテンツのビットレートによってはリレー自体が不可能であることもある。

2. は NAT/Firewall 越え問題である。P2P によるリレーを行う際は、ALM に参加する各ホストがお互いにルータを越えて通信を行う必要があるが、オーバーレイネットワークを構築する場合は、配信トポロジを構築する段階で NAT/Firewall 越えを意識したトポロ

† 早稲田大学
Waseda University

ジを構築する必要がある。

3. は商用利用を考慮した際の問題である。トポロジーの構成を分散処理で行うアプローチ^{2),6)}は多く研究されてきたが、商用利用を考慮した際は分散処理を行うことでユーザの視聴履歴についてのログを効率的に取得することが困難になる。課金や認証などを行う際、ユーザのログは有益であり、また必要となる情報である。

本研究では、実際のインターネット上での多チャンネル映像配信を念頭においてあり、実ネットワーク上でALMを実現するためにこれらの問題点の解決も同時に図る。

1.2 要求事項

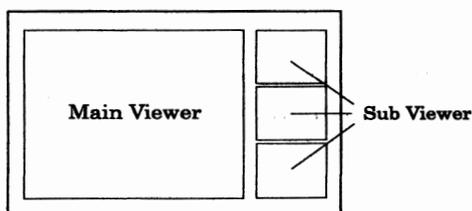


図1 コンテンツビューア

本研究では「ザッピング」の方法を次のような手順で定義する。図1は本研究で想定しているコンテンツビューアである。ビューアの中央にメイン映像用のビューア (Main Viewer) を配置し、その横に同時刻において配信されている他のコンテンツを確認するための小さなビューア (Sub Viewer) を複数個配置する。ユーザは Sub Viewer をマウス操作で視聴したいコンテンツを選択し、コンテンツの表示を切り替えることで、映像チャンネルのザッピングを実現する。

数種類のデータレートをを用いて配信実験を行った結果、Sub Viewer 用のコンテンツはキャプチャサイズ 172×132px, 10~20kbps, 5~10fps 程度のデータ、Main Viewer 用のコンテンツは 640×480px, 700~800kbps, 30fps 程度のデータが要件を満たす最低限の配信設定であることが分かった。

また、コンテンツのザッピングを行う毎に毎回コンテンツデータの受信を開始していればTVのような違和感なくザッピングが行えない。そこで、違和感なくザッピングを行うためにデータの先読みを行わせておくとすると Sub Viewer 用データは常に 5~10(表示されてある Sub Viewer データの数+α) 程度のコンテンツ分のデータを受信しておく必要がある(図2)。

つまり、先に述べた Main Viewer, Sub Viewer での

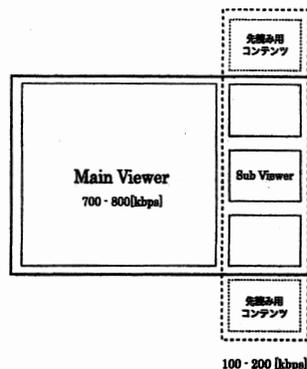


図2 各コンテンツのデータレート

配信設定のデータレートから計算すると、ALM 上で多チャンネル映像配信サービスを受けるためには少なくとも 1Mbps 程度の下り帯域が必要であり、同時に他のホストにデータを再配信 (リレー) するためには 1Mbps × 配信ホスト分 の上り帯域が要求されることが言える。以降、本論文では下り帯域に 1Mbps の実測値が確保されるホストを対象として議論を進める。

2. 解決手法

本研究では次の3つの機構を用いて多チャンネル映像配信を実現する。

2.1 上り帯域測定機構

P2P によるリレー配信モデルを構築する際に、あらかじめホストの上り帯域の実測値を測定、取得しておくことは非常に有効な手法である。実測値が視聴するコンテンツデータのビットレートから算出されるある閾値を越えたホストに関しては、越えていないホストと比較してリレーを行う数を増やす措置とすることで、ALM に参加できるホスト数を増やすことが可能であり、その措置は結果的にオーバーレイネットワーク全体の品質向上につながる。

本研究では、閾値を越えたホスト h については、ホスト h がリレーを行う最大数 $N_R(h)$ の値を、配信数決定関数 F を用いて

$$N_R(h) = F(TP(h) - \alpha)$$

として、決定している。 F は ALM で扱うデータレートの総数に依存する。また、 $TP(h)$ は、ホスト h の実測スループット値、 α はゆとり値である。つまり、リレー数を決定する際は、実測スループット値の最大限までリレーを行わせるように決定するのではなく、トラフィックの変化が起きたときにトラフィックの揺らぎを吸収することができるように、ゆとりを考慮させた値を用いて決定させる。帯域の測定においては、コンテ

ンツデータのリレーで用いるプロトコル (UDP/TCP) を利用している。

2.2 ネットワークモデル測定機構

ホストが ALM に参加する時において、各ホストのネットワークモデルを測定する。ここでの「ネットワークモデル」とは 1.GIP, 2.UPnP, 3.NAT の 3 種類である。それぞれ Global IP Address が割り当てられているホスト、Private な IP アドレスを持ちながら、UPnP 機能を持ったルータに接続されているホスト、それ以外のホストを指す。

つまり、リレーを行う際、GIP モデルのホストと UPnP モデルのホストはリレーの接続要求をリレープロトコルに依存せず受け付けることは可能であるが、NAT モデルのホストは接続要求の受け付け処理が不安定であり、場合によっては全く不可能であることがある。

2.3 トポロジ管理サーバ

トポロジ管理サーバは、ALM に参加している全ノードの接続状態を管理する。管理サーバはコンテンツ毎に独立したプロセスで稼動し、コンテンツの視聴状況の情報を逐一 DB に格納させる。たとえば、「ch.100」のコンテンツを視聴していたユーザ A がチャンネルの切り替えを行い、「ch.200」に切り替えるときは、
1. ch.100 用の管理サーバに自分自身が ch.100 用のトポロジから脱退する旨を宣言 (Leave 命令の送信)、
2. ch.200 用の管理サーバに ch.200 用のトポロジに新規接続要求を宣言 (Join 命令の送信)、という手順を踏むこととなる。ch.100, ch.200 用の管理サーバはそれぞれ「ユーザ A : Leave(ch.100)」「ユーザ A : Join(ch.200)」というログを記録する。また、全トポロジを把握していることからホストの接続場所を高速に提供することができ、同時にログを逐一保存していることから商用配信にも応用が可能である。実際の配信トポロジの構成方法については次章以降で述べる。

ホストはシステム稼動後、コンテンツの視聴を開始する。ホストはまず、SubViewer に表示するコンテンツとザッピングを行うための先読みデータ用コンテンツを取得する。このときホストは n チャンネル分の配信トポロジに同時に参加していることとなる。ザッピングの結果、マウス操作によってあるコンテンツを Main Viewer で視聴を行おうとする時は、更に新たに高レートデータの配信トポロジに参加することになる。ただし、ホストの利用帯域を可能な限り小さく押さえるためにある時刻 t において、高レート用のトポロジに参加している数は最大で 1 とする。つまり、ザッピングの結果、Main Viewer で視聴するコンテンツを切り替える際は、実際に高レートデータを受信するため

のトポロジにも参加・脱退を繰り返すこととなる。

一般的にトポロジの集中管理は大規模なグループについて扱う際にはスケールしないと言われているが、プロセスの分離、スレッドプールの利用やバックアップサーバの設置など、コードレベルでの対応も不可能では無い。⁴⁾ また、分散管理手法と比較して新規接続場所を ALM 参加要求ホストに対して高速に提供することができ、全トポロジを把握していることからホスト間遅延などメトリックの考慮も可能である優位性を挙げることができる。⁵⁾

3. 検証実験と提案トポロジ

3.1 実験の目的

まず、トポロジ管理サーバによるトポロジ構成で、実際にはどの程度の数のホストに対してコンテンツのリレーが可能であるかどうかを検証を行った。

実験は 2 日間行い、両日共に約 3 時間のスポーツ番組のコンテンツを 1 チャンネル放送する実験を行った。配信されるコンテンツは 320 × 240px のサイズでキャプチャされた 700kbps, 15fps のデータであり、H.264 エンコーディングされたものを扱った。また、Java で実装されたクライアントソフトウェアをインターネット上でフリーウェアとして配布し、約 1800 人程度を対象に実験を行った。つまり実験対象ユーザは特定の環境に強く依存されたものではなく、無作為に選ばれたものである。

配信トポロジは完全 1 分木、つまりリスト構造とした。これは最も簡単なリレーモデルを採用することで、インターネット上でのリレーがどの程度可能なかを明確に検証するためである。今回の実験では配信サーバから 4 本のリスト型配信トポロジを構築しているトポロジとした。コンテンツデータのホスト間におけるリレーに用いるプロトコルは UDP を用いた。

ホスト間の接続相手を選択する ParentSelect は Minimun-Depth と RoundRobin の 2 種類のアルゴリズムを用いた。同時にインターネット上でのホストのネットワークモデルの割合や上り帯域幅の実測値がどのようなものかを計測した。

3.2 実験の結果

ParentSelect において、Minimun-Depth を用いた場合はトポロジ管理サーバ側の Personal Firewall の未検知や NAT ホストの扱い方の不十分さなどに起因することからトポロジ管理サーバが稼動しなくなり、再起動を 2 度行った。RoundRobin を用いた場合は再起動を行うことはなく、ホストの平均接続時間は 1.28 倍に伸びた。

NW Model	GIP	UPnP	NAT
Rate[%]	7	54	39

表 1 ネットワークモデルの割合

Throughput[Mbps]	0.0-1.0	1.0-2.0	2.0-3.0	3.0-
Rate[%]	4.9	8.8	64.6	21.7

表 2 上り帯域幅スループット実測値の割合

また、ネットワークモデルの割合や帯域幅の実測値測定の結果は表 1、表 2 となった。

3.3 実験の考察

2.2 で述べたように、GIP と UPnP モデルのホストをリレー可能なホスト、NAT モデルのホストをリレー不可能なホストと見なすと、実験結果から実ネットワーク上でのホスト比率は 62.1 : 37.9 となる。上り帯域幅の観点から見ると 1.2 で述べたように、1Mbps の実測値が確保できるホストをリレー可能なホストと見なすと、リレー不可能なホストとの比率は 95.1 : 4.9 となる。

ここでネットワークモデルと上り帯域幅の値の結果をまとめて、リレー可能なホストを **Relayable** ホスト、リレー不可能なホストを **Non-Relayable** ホストと定義し、その比率を求めると 52 : 48 と求めることができる。つまり、現在においてインターネット上のホストの約半分が **Relayable** ホスト、残りの半分が **Non-Relayable** ホストである、という特性を得ることができる。ネットワークの広帯域化を考えると、今後は当然 **Relayable** ホストの数が増加していくことが期待できる。

3.4 提案手法

3.3 で述べたホストの特性から、本研究では ALM のトポロジとして **Relayable** ホストがデータの受信とデータリレーを行うが、**Non-Relayable** ホストはデータの受信のみを行い、データリレーを一切行わないという手法を提案する。

Relayable ホストは、接続されている自分の子ホストにリレーを行うのと同時に、1 ホスト分の **Non-Relayable** ホストにデータをリレーする (計 2Mbps の上り帯域利用) と上記の提案モデルは検証実験から得られたホストの割合的には十分に実現することができる。また、表 2 より、帯域幅の実測値が平均的な値である **Relayable** ホストであれば、上記の 2 つのホストにリレーする、ということも十分に実現可能であることであることが分かる。

コンテンツの種類やコンテンツの開始、終了の時間帯によっては、これらの比率には揺らぎが生じることも

予測される。しかし、リレーが可能なホスト群の中で、上り帯域値が十分に大きなホストは **Non-Relayable** ホストへの配信数を増やすことでその対応は可能であると考えられる。

反対に、コンテンツの開始直後など十分な数の **Relayable** ホストがオーバーレイネットワーク上に存在しない時に、多くの **Non-Relayable** ホストが接続を試みることも考えられる。この場合は、**Non-Relayable** ホストは一定時間接続を待機してもらい、その後に再接続を行うものとする。**Relayable** ホストが接続を行えば行うほど **Non-Relayable** ホストが接続できる確率が上がり、待機時間も短くなることとなる。そのため、本論文で提案する手法において **Relayable** ホストと **Non-Relayable** ホストとの比率は大変重要なものとなる。

図 3 は、本研究で提案する 1 チャンネルあたりの配信トポロジを示したものである。多チャンネル配信時は、自ホストが同時に複数のチャンネルのトポロジに参加することとなる。

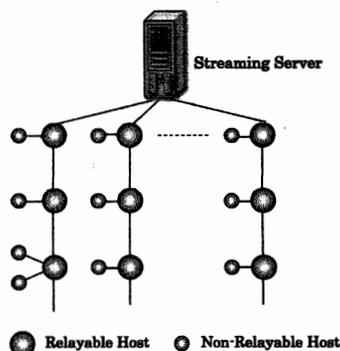


図 3 1 チャンネルあたりの提案配信モデル

4. 評価実験

4.1 シミュレータの作成

本手法の有効性について調べる評価実験のために、オーバーレイネットワークのシミュレータを作成した。参加するホストの総数、ネットワークモデルの割合、上り帯域のスループット実測値の割合、コンテンツの配信時間をパラメータとすることで、擬似的なホスト、疑似参加・脱退クエリを生成する。

生成されたホストは、配信開始時刻から終了時刻まで常に参加、脱退の疑似イベントクエリを起こす。また、トポロジ管理サーバはこの疑似イベントクエリを、実際のクエリと見なして接続ホストを促す。

ここで、ある時刻 t において、総ホスト数における参加クエリを送信するホスト数の割合は 3.3 の結果を用いて次のように求めた。

- (1) 3.3 の結果から各分毎の参加ホスト数 n_t を算出
- (2) コンテンツ配信時間内における参加ホスト総数 n を算出
- (3) (1) と (2) の結果より配信時間内における各分毎の参加ホスト数の割合 r_t を算出

シミュレータでは、参加ホスト総数を変化させ、上で求めた割合を用いることで各分毎の参加クエリを送信するホスト数を決定している。

また、参加要求を送信したホストが離脱要求を送信する時刻を求める方法については、次のように行っている。

- (1) 3.3 の結果から各分毎の参加ホスト数 n_t を算出
- (2) 時刻 t において参加要求を送信したホストにおいて、それぞれ離脱要求を送信した時刻を取得
- (3) (2) の結果より時刻 t において参加要求を出したホスト群の接続時間のリスト $c_1(t), c_2(t) \dots c_i(t)$ を算出
- (4) (3) の結果より時刻 t におけるホスト群の最短接続時間 $c_{min}(t)$ と最長接続時間 $c_{max}(t)$ を算出
- (5) シミュレータ上での時刻 t における参加ホスト数 N_t の接続時間を $random(c_{min}(t), c_{max}(t))$ として定義

つまり、実験結果より時刻 t に参加要求を送信したホストの最短接続時間と最長接続時間を算出し、その範囲内で発生させた乱数の値をシミュレータ上での接続時間として採択している。よって、本シミュレータでは、時刻 t で擬似参加要求を送信したホストは、 $random(c_{min}(t), c_{max}(t))$ 時間後に再び擬似脱退要求を出すものとしている。

実験では、参加ホスト数を 1000, 5000, 10000 と変化させたとき、およびホスト数を 1000 として Non-Relayable ホストの割合を 30, 40, 50, 60% と変化させたときにおいて、Resource Index²⁾、トポロジ管理サーバの CPU クロック周波数、命令実行数における Non-Relayable ホストの接続待ち状態数の変化などの値の変化を調べた。

ここでの Resource Index とは、ネットワーク上におけるリソース使用率を表すもので、(実際に配信を行っているパス数+余剰パス数)/(実際に配信を行っているパス数) の値として表される。値が大きい方がリソース使用率が低いことを表す。また、先に述べたとおりオーバーレイネットワークの状況によっては、Non-Relayable モデルのホストは接続を行うまでに一

上り帯域幅 [bps]	配信数
0.0 - 1.0	0
1.0 - 2.0	1
2.0 - 3.0	2
3.0 -	4

表 3 上り帯域幅における配信数

定時間だけ待機を行う必要がある。ホストの総数や Relayable ホストと Non-Relayable ホストの割合の変化によって全命令実行数中における待ち状態がどれほどの回数あったかを取得することで、オーバーレイネットワークの品質を求めることができる。

擬似ホストは合計 5 チャンネル分のコンテンツが 3 時間に渡って配信されていることを想定し、それぞれのコンテンツに対して参加・脱退の擬似クエリを送信する。擬似ホストにおける Relayable ホストと Non-Relayable ホストの比率は 3.3 で得た結果を利用している。また、Relayable ホストの配信決定関数 F は便宜上次の表 3 のように決定した。

4.2 実験の結果

degree, ResourceIndex, トポロジ管理サーバの CPU クロック数の変化の結果をそれぞれ表 4, 表 5 に、また Non-Relayable ホストの接続待ち状態数の変化を表 4 に示す。結果のグラフや表における N はホスト数, NR は Non-Relayable を指す。

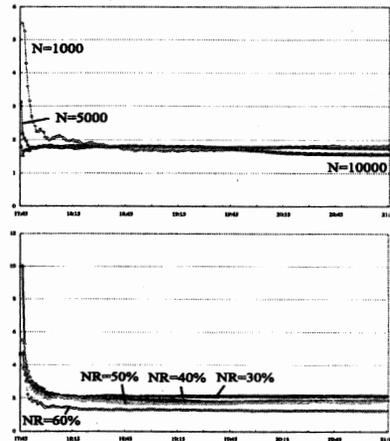


図 4 ResourceIndex の変化

オーバーレイネットワークを構築するホストがインターネット上のホストのネットワークモデル、上り帯域幅の分布に従って存在する場合、本手法を用いることで次のことが導ける。

- 各ホストは常に余剰帯域を持ったままオーバーレイネットワークを構築できる。

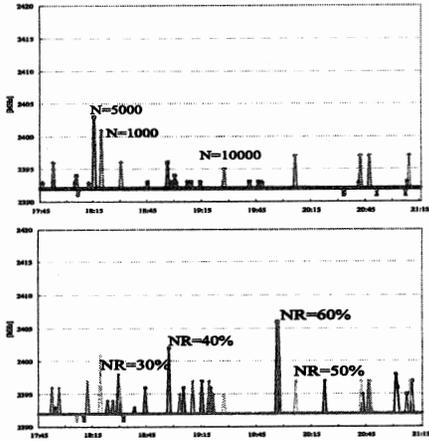


図5 トポロジ管理サーバのCPU実クロック周波数の変化

N	待ち状態数 / 実行命令数
1000	3 / 1616
5000	4 / 9529
10000	4 / 19292
NR	待ち状態数 / 実行命令数
30%	1 / 1609
40%	3 / 1612
50%	4 / 1616
60%	11 / 1614

表4 命令実行数に対する待ち状態数

- 10000 ホスト程度であれば参加ホスト数に関わらず、トポロジ管理サーバは常に安定している。
- データをもらう相手が存在しない場合はほぼありえなく、十分に稼働する。

また、Non-Relayable ホストの割合を変化させたときは次のことが導ける。

- Relayable ホストの割合を増やすことで、同時接続ホスト数を増やすことができる。
- 全体の半数以上が Non-Relayable ホストであるときも、ネットワークへの参加待ちになることはほぼありえなく、十分にスケールする。

以上より、インターネット上のホストの特性を考慮したトポロジを構築する本手法は1万人規模のデータリレーを行うことも十分に可能であり、また、ネットワークモデルや上り帯域幅などユーザの環境の違いを吸収することができる手法であると言える。

5. まとめ

本論文では、多チャンネル映像配信を行おうとする際に、インターネット上のユーザ環境の特性を考慮したP2Pを用いたALMによる配信トポロジを構築する手法を提案した。

まず、多チャンネル映像配信を実現する上で、最低限必要とする利用帯域幅を定義し、チャンネルのザッピング機能、コンテンツホルダのログ管理など、付加すべき必要機能を提示した。更に、参加ホストを帯域幅やネットワークモデルによってRelayableホストとNon-Relayableホストに分類した配信トポロジを構築する手法を提案した。この手法は、インターネット上での検証実験結果から、実現が十分に可能なものであることを示した。

また、評価実験の結果から、提案した手法によってトポロジを構築し、3時間のコンテンツで10000程度のホストやNon-Relayableホストが半数以上存在する場合でも安定してスケーリングできることが確認できた。

参考文献

- 1) 青山 友紀, "CDNで流す大規模コンテンツ配信技術の課題と解決策を探る," テレコミュニケーション5月号, pp.128-131, May 2002.
- 2) Y. Chi, S. Rao, S. Seshan, and H. Zhang, "A Case For End System Multicast" Proc. ACM SIGMETRICS 2000, June 2000
- 3) S. Shi, J. Turner and M. Waldvogel, "Dimensioning Server Access Bandwidth and Multicast Routing in Overlay Networks", Proc. of Network and Operating System Support for Digital Audio and Video 2001, 2001.
- 4) Dimitrios Pendarakis, Sherlia Shi, Dinesh Verma, Marcel Waldvogel, "ALMI: An Application Level Multicast Infrastructure", Proceedings of the 3rd USENIX Symposium on Internet Technologies and Systems, 2001.
- 5) V. N. Padmanabhan, H. J. Wang, P. A. Chou, and K. Sripanidkulchai "Distributing Streaming Media Content Using Cooperative Networking", ACM NOSSDAV, Miami Beach, FL, USA May 2002
- 6) 常村 和史, 山口 弘純, 東野 輝夫, "オーバーレイマルチキャストにおける遅延最小木の自律的再構築アルゴリズムの提案と評価", 情報処理学会研究報告(情処技報), Vol. 2002, No. 111 (2002-DPS-111), pp.7-12, March 2003.