

ザッピング視聴を考慮した オーバーレイ型映像配信プロトコルの一検討

壹岐勇太郎¹⁾ 青木輝勝¹⁾²⁾ 沼澤潤二¹⁾²⁾

¹⁾東北大学 情報科学研究科 ²⁾東北大学 電気通信研究所

1. はじめに

近年、ブロードバンド回線の普及により、インターネット上でのコンテンツ配信が活発となってきている。このようなコンテンツ配信を行うための通信方式としては、主にユニキャスト、ブロードキャスト、マルチキャストなどの通信方式があげられるが、配信サーバの負荷や帯域効率など観点からマルチキャストによる配信に注目が集まっている。また、今後コンテンツの多様化に伴い、インターネット上の放送局は増加していくと予想される。そのような状況では、地上波テレビにおいて一般的に行われている、チャンネルを頻繁に切り替えながらの視聴（ザッピング視聴）への要望が高まると予想される。しかしインターネット上のテレビ放送では、全チャンネルのデータの送信が困難なことや接続までに時間がかかるなどのネットワークの性質のため、チャンネルの切り替え時に遅延が発生してしまい、ユーザにとってストレスとなる。

そこで本研究では、オーバーレイマルチキャストを用いたコンテンツ配信における、チャンネル切り替え時の遅延を最小限に抑えるための解決法を提案する。

2. オーバーレイマルチキャスト

2.1. 概念

マルチキャストとは、配信サーバとマルチキャストグループに属する複数のクライアントが通信を行う方式のことである。マルチキャストでは、サーバが送信するデータは 1 つでよく、ブロードキャストと異なりネットワークの制限は無く、さらにマルチキャストグループに属していないクライアントには負荷がないため、放送型コンテンツ配信には適している。

マルチキャストは、トポロジ管理やデータ転送の管理などの処理をどの層で行うかで、2 つの

種類に分けることが出来る。ネットワーク層でそれらの処理を行う IP マルチキャストとアプリケーション層でそれらの処理を行うオーバーレイマルチキャストである。IP マルチキャストはそれらの処理をネットワーク上のルータが行わなければならない、さらにルータが IP マルチキャスト対応でなければならないという制約があるため、実現に相当な時間を要すると予想される。一方、オーバーレイマルチキャストはノード上のアプリケーションが実際にそれらの処理を行うため、IP マルチキャストと比較して実現性が高い。よって本研究では、オーバーレイマルチキャストを研究対象とする。

2.2. 既存のオーバーレイマルチキャストプロトコル

既存のオーバーレイマルチキャストプロトコルの多くは 2 種類のトポロジをもつ^[1]。各ノードの管理を行なうコントロールトポロジと実際のデータ転送のためのデータトポロジである。また各プロトコルはその 2 つのトポロジの構成手法から 3 種類に大別可能である。メッシュ状の制御トポロジを最初に形成する Mesh-first 型、分散的にデータ転送の為のツリーを最初に形成する Tree-first 型、コントロールトポロジを特定の方法で形成する Implicit 型である。今回はこの中でデータがツリーに沿って流れていくためデータの制御を正確に行うことができる Tree-first 型のアプローチを採用する。

図 1 に Tree-first 型の構成例を示す。Tree-first 型では最初にデータトポロジの形成を行う。次にそれぞれノードが隣接ノード以外と追加的な制御バスを張ることによりコントロールトポロジを形成する。Tree-first 型のプロトコルのとしては Yoid^[2]が挙げられる。

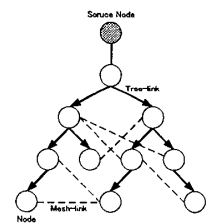


図 1. Tree-first 型

“Improvement of Channel Zapping Time in Overlay Multicast Protocol for Delivering Video”

¹⁾Yutaro Iki · Graduate School of Information Sciences, Tohoku Univ.

¹⁾²⁾Terumasa Aoki · RIEC, Tohoku Univ.

¹⁾²⁾Junji Numazawa · RIEC, Tohoku Univ.

3. LBOVM 方式の提案

本システムは複数の放送局がインターネット

上の異なる場所に存在し、テレビ放送を行っている環境を想定している。そのような環境において各ノードの負荷が分散するようなデータ管理の方式として、LBOVM(Load Balancing for Overlay Video Multicast)方式を提案する。

まず、各クライアントにおいて、視聴しているチャンネルをメインチャンネル、その他のチャンネルをサブチャンネルと定義する(サブチャンネルはどのチャンネルを視聴しているかによってクライアントごとに異なる)。各クライアントはメイングループとサブグループにそれぞれに所属し、メイングループで各クライアントが視聴しているメインチャンネルの元データを受信し、サブグループでサブチャンネルのデータを受信する。そして、ザッピングを行うときはサブグループの階層符号化データで行うことで、遅延を抑制することを狙いとする。ここでいう遅延というのは、チャンネル接続までの遅延のことである。

図2と図3、図4にメイングループとサブグループのそれぞれの構成を示す。

図2のメイングループでは、各放送局から現在放送している番組の元データが流れている。

サブグループの構成としては図3と図4のような方式が考えられる。図3の方式1では、各放送局の階層符号化データをまずどこかのメイングループの1つのクライアント(以降、SPC(Special Client)と呼ぶ)に集め、その後それ以外のメイングループの頂点にあるクライアント(以降、RC(Root Client)と呼ぶ)にデータを送信し階層符号化データを広めていく方式である。図4の方式2では方式1と異なり、各RCが各放送局の階層符号化データを自分以外のRCに送信しあうことで階層符号化データを広めていくという方式である。各方式について比較を行うと、方式1ではSPCに全ての放送局の階層符号化データが集まり流れていくため、データ管理が容易となる。しかし、SPCに負荷が集中し、SPCが故障した場合の影響も大きくなってしまふ。一方、方式2では階層符号化データをRC同士が送信しあい広めていくため、各クライアントに負荷を分散させることができる。

具体的に放送局数を x 、各クライアントの子の数を y とし各方式のデータの送信負荷について比較すると、方式1のSPCは自分以外の各RCと子にデータを送信しなければいけないため、 $(x(x-1)+xy)$ のデータを送信する。その他のRCとクライアントは xy のデータを送信する。一方、方式2では、各RCは自分以外の各RCと子にデータを送信しなければいけないため、 $(x-1+xy)$

のデータを送信する。その他のクライアントは xy のデータを送信する。

以上より、方式1ではSPC以外のクライアントの負荷は線形的であるが、SPCの負荷が指数的に増えていく。一方、方式2では各クライアントの負荷が線形的に増えていくため、その負荷が少ないことが分かる。よって、本システムでは、負荷の分散が行える方式2を採用する。

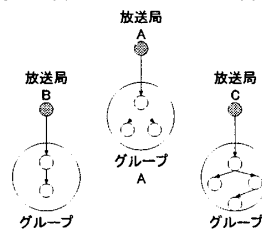


図2. メイングループ構成

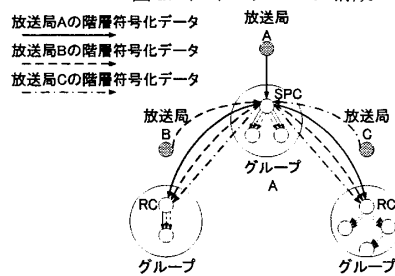


図3. サブグループ構成(方式1)

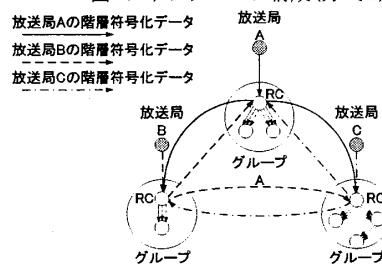


図4. サブグループ構成(方式2)

4. まとめと今後の課題

本稿では、オーバーレイマルチキャストを用いたコンテンツ配信における、チャンネル切り替え時の遅延を最小限に抑えるための解決法を提案し、方式の比較を行った。また、本システムのプロトタイプを試作しシミュレーションを行った結果、有効な結果を得ることが出来た。

今後は、階層符号化データを用いて実験を行い、本システムによる効果の検証を行っていく予定である。

文献

- [1] S. Banerjee and B. Bhattacharjee, "A comparative study of application layer multicast protocols", Unpublished report (2002), available at <http://minoas.di.uoa.gr/>.
- [2] P. Francis, "Void: Extending the Internet Multicast Architecture", Unpublished paper (2000), available at <http://www.aciri.org/void/>.