

オーバーレイ型映像配信プロトコルにおけるザッピング視聴の最適化の一検討 Improvement of Channel Zapping Delay in Overlay Multicast Protocol for Video Delivery

壹岐 勇太郎[†]
Yutaro Iki

青木 輝勝^{†‡}
Terumasa Aoki

沼澤 潤二^{†‡}
Junji Numazawa

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。そのような配信を行うための通信方式としては、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、ネットワークの性質により、ザッピング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、ザッピング用のチャンネルデータを用意することで、オーバーレイ型のテレビ放送におけるザッピング時の遅延を最小限に抑える手法の提案を行う。

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ストリームは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。本研究では、より実現性の高い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、データ転送の為に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 型のプロトコルのとしては Yoid^[2]が挙げられる。

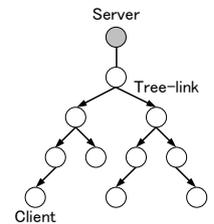


図1 Tree-first型

3. 提案手法

3.1 概要

本研究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、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ザッピングが快適に行えないのは、地上波テレビと異なり、帯域制限のために放送中の全チャンネルのデータを常に受信できないためである。そのためザッピング時に、毎回放送局への接続処理を行わなければならないため、遅延が発生してしまう。

そこで本研究では視聴しているチャンネル以外のデータも常に受信できるようにすることで、ザッピング時の遅延を抑制することを狙いとする。しかし、全てのチャンネルの元データ(視聴用データ)を常に受信することは帯域制限のために不可能であるため、ザッピングのための低ビットレートのチャンネルデータ(ザッピング用データ)を用いることで、それを実現する。また、ザッピング後はザッピング用の映像から視聴用の映像に自動で切り替わるようにする。

3.2 LBOVM 方式の提案

ここでは、3.1 で述べたザッピング用データを各クライアントに配信する方法として、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アザーチャンネルと定義する(アザーチャンネルはクライアントがどのチャンネルを視聴しているかによって異なる)。各クライアントはメイングループとザッピング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視聴用データを受信し、ザッピンググループでアザーチャンネルのザッピング用データを受信する。

図2と図3、図4にメイングループとザッピング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視聴用データが流れている。ザッピング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、各放送局のザッピング用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)にデータを送信しザッ

[†] 東北大学 情報科学研究科 Graduate School of Information Sciences, Tohoku University.

[‡] 東北大学 電気通信研究所 Research Institute of Electrical Communication, Tohoku University.

ピング用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が各放送局のザッピング用データを自分以外のRCに送信しあうことでザッピング用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すると、方式1ではSPC全ての放送局のザッピング用データが集まり転送されていくため、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ではザッピング用データをRC同士が送信しあいザッピング用データを広めていくため、各クライアントに負荷を分散させることができる。よって、本研究では、負荷の分散が行える方式2を採用し、LBOVM方式と呼ぶ。

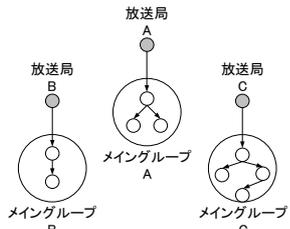


図2 メイングループ構成

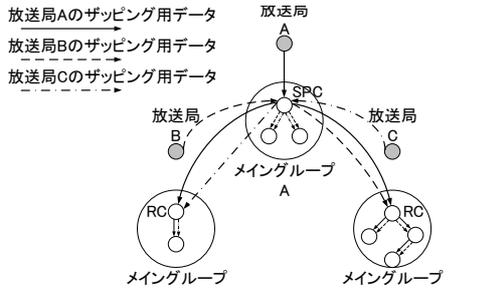


図3 ザッピンググループ構成(方式1)

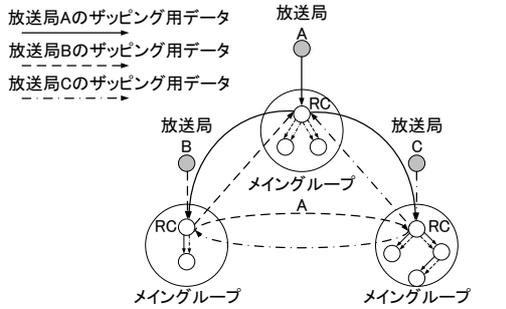


図4 ザッピンググループ構成(方式2)

4. 提案方式の評価

各方式を評価するため、視聴用データとザッピング用データの転送シミュレーションプログラムを方式ごとに構築し、クライアントの回線速度を変化させた時の全クライアントの平均の遅延とスループットの検証を行った。シミュレーションでは、放送局が送る視聴用データとザッピング用データの送信レートをそれぞれ5Mbpsと1Mbps、放送局数を10局、クライアントが受け取ったデータを転送するクライアント数を2、全体のクライアント数を100、データ送信時間を1分間としてシミュレーションを行った。また、ノードの管理は座標で行い、座標でノードの位置とアドレスを表現し、座標間隔を各ノード間の遅延(0.5 μ sec)とした。さらに、今回のシミュレーション

では、パケットの優先制御を用い、より重要である視聴用データを優先的に転送できるよう設計した。

図5と図6が、シミュレーションの結果である。それぞれの結果を比較すると、視聴用データのスループットは優先制御を用いたため、収束する回線速度が各方式でほぼ同時になっていることが分かる。しかし、ザッピング用データのスループットの収束する回線速度は方式1に比べ、方式2の方が低いことが分かる。これは、方式1のSPCに負荷が集中してしまい、データ転送により高い回線速度を必要としたためであると考えられる。また、遅延に関しても、方式2の方が低遅延でデータ転送できていることが分かる。

以上の結果から、方式2の方が負荷が分散されており、方式1より低い回線速度で低遅延のザッピング視聴が可能であることを実証できたと考えられる。

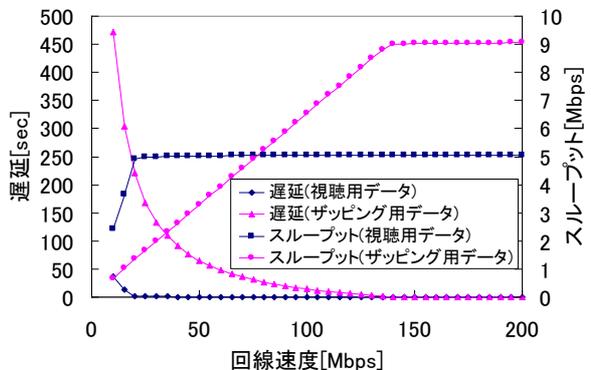


図5 方式1のシミュレーション結果

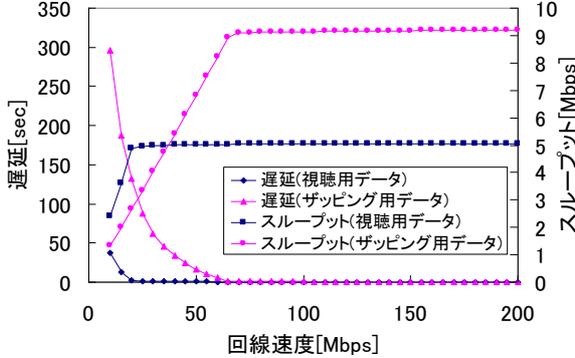


図6 方式2のシミュレーション結果

5. まとめと今後の課題

本稿では、ザッピング用のチャンネルデータを用意することで、オーバーレイ型のテレビ放送におけるザッピング時の遅延を最小限に抑えるLBOMV方式を提案し、提案方式の評価のためシミュレーションを行い、LBOMV方式の有効性を示した。

今後は、各クライアントが周囲の状況を自律的に判断し、データ転送のトポロジを変化させることで、効率的なデータ転送が行えるよう改良していく予定である。

参考文献

[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, "Yoid: Extending the Internet Multicast Architecture", Unpublished paper (2000), available at <http://www.aciri.org/yoid/>.