

追加マルチキャストを用いた IP 放送の高速チャンネルザッピング手法

佐々木 力[†] 田上 敦士[†] 長谷川輝之[†] 阿野 茂浩[†]

[†] (株) KDDI 研究所 〒356-8502 ふじみ野市大原 2-1-15

E-mail: †ch-sasaki@kddilabs.jp

あらまし 近年、IP マルチキャストを用いた放送サービス (IP 放送サービス) が広く提供されている。本サービスでは、電波によるアナログ放送とは異なりチャンネル切替 (チャンネルザッピング) に数秒かかるという問題がある。この1つの原因としては、ジッタ吸収や各種デコーディングのためのデータバッファリングが挙げられる。これまで、このバッファリング時間を短縮するために、ザッピングの瞬間に専用サーバから付加的なユニキャスト転送を行う手法が提案されている。しかしながら、ユニキャスト通信をベースとしているため、番組終了やCMを契機に多数の視聴者からザッピング要求があった場合、ネットワークやサーバに瞬間的な負荷がかかってしまう。そこで本稿では、負荷緩和のためにバッファリングの加速化を追加マルチキャストにより実現する高速チャンネルザッピング手法を提案する。試作装置を用いた評価では、商用サービスのザッピング遅延を最大で1秒程度短縮できることを確認した。

Rapid Channel Zapping for IPTV Broadcasting Service with Additional Multicast

Chikara SASAKI[†], Atsushi TAGAMI[†], Teruyuki HASEGAWA[†], and Shigehiro ANO[†]

[†] KDDI R&D Laboratories Inc., 2-1-15 Ohara, Fujimino-shi, 356-8502 Japan

E-mail: †ch-sasaki@kddilabs.jp

Abstract In this paper, we propose a novel scheme for accomplishing rapid channel zapping on IPTV broadcasting services. Today, channel zapping delay is a crucial issue for digital TV broadcasting, which entails audio/video data buffering before reproduction. Such delay could be mitigated in an IP environment if a receiver accelerated this buffering using additional burst transmissions from dedicated servers when a channel zap occurs. However, most of conventional solutions are based on the unicast burst, which may cause an impulsive server/network load because channel zaps apt to happen simultaneously when programs are finished or suspended by commercial messages. To suppress this load, we newly propose a multicast-based solution considering the timing variation of channel zap on each receiver. Through verifications on a prototype system, we have confirmed a maximum 1-second reduction in the zapping delay using commercial multicast streams.

1. ま え が き

近年、インターネットが飛躍的に発展し、様々なサービスがネットワーク経由で提供されている。宅内向けの

アクセス回線は光/VDSL回線に移行しつつあり、高速なデータ通信環境を背景に、大容量の映像・音声コンテンツが配信されるようになってきた。最近では、GyaO [1] 等の VOD (Video On Demand) サービスの利用が増加

しつつある。一方で、多数の視聴者が同じ映像コンテンツを同時に視聴する IP 放送サービスも提供されており、大手通信事業者では 2003 年頃より自主型放送サービスを開始している [2]~[4]。さらに、2011 年のアナログ放送停波をターゲットに地上デジタル放送の IP 再送信実現が見込まれている [5], [6]。このような放送型コンテンツを対象とした場合、一度の配信で複数の視聴者に同時配信可能な IP マルチキャスト [7] が有効である。

IP 放送は電波によるアナログ放送とは異なり、視聴者のチャンネル切替（チャンネルザッピング）要求に対して即座に再生を開始できないという問題がある。これは、(1) ネットワークジッタの吸収、(2) FEC [8], [9] 復号、(3) 映像・音声復号（MPEG2 [10], H.264 [11]）を行うために、再生開始前に受信データのバッファリングを要することが一因である。そのため、従来手法では、チャンネルザッピング時に専用サーバから直接ユニキャストでバースト転送を行い、バッファリング時間を短縮している [12], [13]。一方、IP 放送では多数の視聴者が同一の映像を視聴しており、番組終了や CM を契機にチャンネルザッピングが集中する傾向が予想される。したがって、ザッピング要求に対して個別に回答しなければならない従来手法では、サーバやネットワークへの瞬間的な負荷が問題となる。

そこで本稿では、要求数に比例した負荷を緩和するために、バースト転送をマルチキャストにより行う高速チャンネルザッピング手法を提案する。提案手法では、別途配信した遅延付きの追加マルチキャストと通常のマルチキャストを同時受信することで、バッファリング時間を 1/2 に短縮することができる。さらに、追加マルチキャストの送信速度を数倍に高速化することで、更なるバッファリング時間の短縮を行う。この際、受信開始タイミングによらず短縮効果が得られるように、高速化に適した送信パケット生成規則を考案した。また、試作機による評価を行い、3 倍の送信速度で再生開始時間を最大で 1 秒程度短縮できることを確認した。

本稿の構成は以下の通りである。2 節では、既存技術として IP 放送の概要とユニキャストを用いた高速チャンネルザッピング手法について説明する。3 節では、遅延付き追加マルチキャストを用いた高速チャンネルザッピング手法を提案する。また、さらなるザッピング時間の短縮を目指し、数倍速の追加マルチキャストに適したパケット送信規則を示す。4 節において、試作機を用いたザッピング時間の評価を行う。5 節で結果を考察し、6 節でまとめを行う。

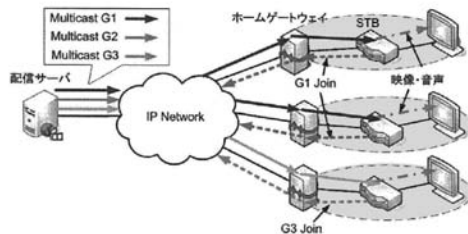


図 1 IP 放送の概要

Fig. 1 IPTV broadcasting overview.

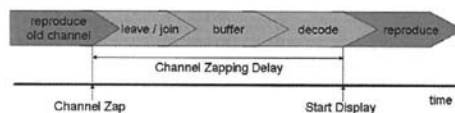


図 2 チャンネルザッピング遅延の内訳

Fig. 2 Channel zapping delay composition.

2. 既存技術

2.1 IP 放送の概要

本稿では、地上波放送や衛星放送、ケーブル放送に対して、IP ネットワーク上で提供される映像・音声の放送サービスを IP 放送と呼ぶ。IP 放送では、番組表に従って多数の視聴者が同一の映像を視聴するため、IP マルチキャストによる配信が適している。図 1 に IP マルチキャストを用いた IP 放送の実現例を示す。配信サーバは、MPEG2 や H.264 等により圧縮された放送コンテンツ（映像・音声（データ放送））をチャンネルごとに異なるマルチキャストアドレスで送信する。なお、ネットワークでのパケットロス回復のために、配信サーバから送信される前に圧縮データに対して FEC を適用するのが一般的である。受信装置である STB（Set Top Box）は、視聴者からのチャンネルザッピング要求があると、視聴チャンネルに対応したマルチキャストグループ G に参加（join）し、また前チャンネルに対応したマルチキャストグループからは離脱（leave）する。STB は受信したデータに FEC の復号化処理や映像圧縮の復号化処理を行い、映像をディスプレイへ送る。このとき、STB ではネットワークジッタ吸収や FEC・圧縮の復号処理を行うために、一定量のデータバッファリングが必要となり、再生開始に秒単位の時間がかかる。図 2 にチャンネルザッピング遅延の内訳を示す。例えば、leave/join にかかる時間は数十 ms、バッファリング時間は 1~2 秒、復号遅延は数百 ms~数秒程度である。本稿では、チャンネルザッピング遅延を削減するために、バッファリング時間を短縮する手法を検討する。

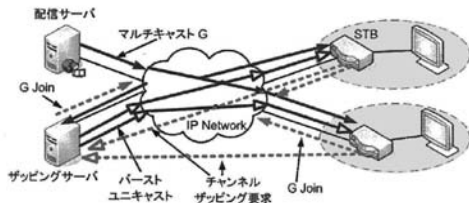


図3 従来の高速チャンネルザッピングの構成 (ユニキャストベース)

Fig.3 Conventional rapid channel zapping framework (unicast based).

2.2 従来のチャンネルザッピング手法

従来、チャンネルザッピング遅延を削減する方法として、ザッピング時に別途高速なユニキャスト通信を行ってバッファリング時間や最初のパケット到達を短縮する手法が提案されている [12], [13]。図3にユニキャストベースの高速チャンネルザッピングの構成を示す。STBは視聴者からチャンネルザッピング要求があると、ザッピングサーバへザッピング要求を送信するとともに、対応するマルチキャスト G へ *join* する。ザッピングサーバは、該STBに対してザッピングが完了するまでユニキャストによる高速データ転送を行う。STBは、ザッピングサーバとマルチキャストの両方からデータを受信することにより、バッファリングの加速化が可能となる。また、[13]では、各視聴者の切替タイミングに応じて、映像圧縮の復号においてベースになる I フレームから転送を行うことで不要なデータ転送や転送時間の削減を実現している。

3. 提案手法

3.1 追加マルチキャストを用いた高速チャンネルザッピング手法

従来手法では、ユニキャスト通信を用いているため、数百万規模の視聴者が同時にチャンネル切替を実施すると、サーバや中継ノードに負荷がかかる。特にテレビ放送のようなコンテンツでは番組開始・終了やCM等を契機に番組切替が集中することが予想される。今後は、10Mbps以上の高品質なコンテンツ配信も想定されることから、負荷集中による影響はより深刻になる可能性がある。

上述の問題を解決するために、本稿ではユニキャストの代わりに遅延付き追加マルチキャストを用いた高速チャンネルザッピング手法を提案する。以下に、本稿で想定したSTBの仕様を示す。

- STBは初期バッファとして再生開始前に b 個のパケットを蓄積する。
- STBは受信順序逆転のあったパケットを正しく並び替えて処理することができる。

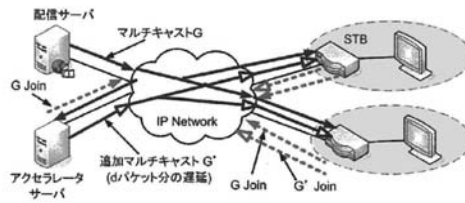


図4 提案方式の概要 (マルチキャストベース)

Fig.4 Proposed framework (multicast based).

図4に提案手法に基づくシステム構成例を示す。具体的手順は以下の通りである。

提案手法：

- (1) アクセラレータサーバは、*IGMP join* メッセージを隣接ルータに送信する (*join*) ことで、マルチキャスト G を受信する。
- (2) アクセラレータサーバは、受信したマルチキャスト G のアドレスを G' に変換し、 d パケット受信時間分遅らせて送信する。
- (3) STBは視聴者からのザッピング要求に対して、 G と G' の両方に *join* する。
- (4) STBは初期バッファ分の b パケットを蓄積するまで、 G と G' の受信を続け、パケット順序の整理を行う。
- (5) STBはバッファリングしたパケットを復号し、再生を開始する。さらに、マルチキャスト G' から離脱 (*leave*) する。

□

本手法では、追加転送にマルチキャストを使用することにより、複数の視聴者が同時にザッピングを行っても、アクセラレータサーバはマルチキャスト G' を複数回送信する必要はない。したがって、サーバやネットワークでは総チャンネル数に応じた負荷が発生するのみであり、ユニキャストの追加転送に比べて負荷が小さくなる。アクセラレータサーバで付加する遅延のパラメータ d は、STBに必要なバッファリング量を b として、

$$d = \lceil b/2 \rceil \quad (1)$$

に設定する。ただし、 $\lceil x \rceil$ は x の小数点以下切り上げを表す。

図5に $b=6, d=3$ としたバッファリングの例を示す。通常のザッピングでは、マルチキャスト G のみを受信するため、STBが G の番号4のパケットから受信し始めたとすると、番号9のパケットを受信したところで初期バッファリングが完了し、番号4から復号・再生が行われる。それに対して、本手法では G' により番号1~3が受信されるため G の番号6を受信した後に番号1からの復号・再生が可能である。すなわち、バッファリング時

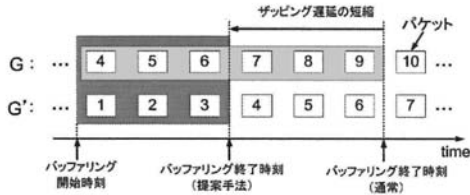


図5 バッファリングの例 ($b = 6, d = 3$)
Fig. 5 Example of buffering ($b = 6, d = 3$).

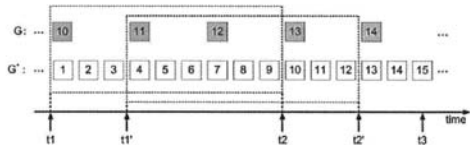


図6 r 倍のバッファリングの例 ($r = 3, b = 12$, 不適切な順序)
Fig. 6 Example of r multiplied rate ($r = 3, b = 12$, silly ordering).

間を $1/2$ 程度に短縮できる。なお、本手法を用いた場合と通常のザッピングを行った場合で再生のずれは発生しない。

3.2 r 倍速への拡張

上述の手法ではバッファリング時間を $1/2$ にしか短縮できない。そこで、アクセラレータサーバから送信するマルチキャスト G' の送信速度を数倍にし、さらなるザッピング時間の短縮を行う。以下では、 G に対する G' の送信速度比を r (r : 自然数) とする。

図6に $r = 3, b = 12$ としたときのSTBにおけるバッファリングの例を示す。STBは時刻 t_1 から受信を開始し、時刻 t_2 までの間に G と G' から12個の packets を受信する。このとき、受信した12個の packets 番号は1~12で連続しているため、バッファリングは完了する。しかしながら、 t_1' から t_2' まで同じ時間受信した場合、重複する packets (番号11,12) が存在しているため、有効な packets は10個しかない。さらに、 G' の送信 packets 番号は単純に1ずつ増加しているため、時刻 t_3 において G の packets 番号を越えてしまう。 G' は G から生成されるマルチキャストのため、この G の送信規則は不適切である。これらを考慮し、以下の方針に従った G' の送信規則を提案する。

方針:

- (1) 各処理は、 G の1個 packets と G' の r 個の packets を1つの単位として扱う (図7参照)。
- (2) G の1個の packets に対して、対応する G' の r 個の packets 番号を決定する。
- (3) どこから受信を開始しても並び替えによりユニークかつ連続した packets がバッファリングできる。

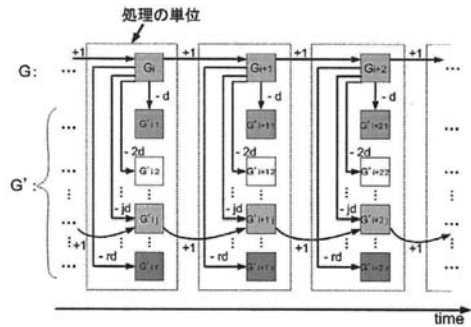


図7 送信 packets 順序の関係
Fig. 7 Relationship of packet orders.

- (4) G' の packets 番号は、 G の packets 番号を越えない。

□

方針(1)では、受信のタイミングを単純化するために、STB内の処理として、 G' の r 個の packets が G の1個の packets と同時に受信されると仮定している。さらに方針(2)において、 G' の r 個の packets 番号を G の1個の packets 番号に基づいて定義することで、送信すべき G' の packets 番号を繰り返し計算することができる。図7に、packets 番号の順序関係を示す。詳細な G' の packets 送信規則は以下の通りである。

提案規則:

マルチキャスト G の packets 番号を G_i (i は自然数) とし、 G_i と同時に受信される G' の r 個の packets 番号を $G'_{i-1}, G'_i, \dots, G'_{i+r}$ とする。このとき、 $j = 1, 2, \dots, r$ に対して、 G'_{i-j} を

$$G'_{i-j} = G_i - j * d \quad (2)$$

により決定する。ただし、 $d = \lfloor b / (r + 1) \rfloor$ とする。 $[x]$ は x の小数点以下切り上げである。

□

図7に示すように、 G' の他の packets $G'_{i+1}, G'_{i+2}, \dots, G'_{i+r}$ についても、 G の packets 番号を G_i から G_{i+1} に変更することで、式2を用いて packets 番号を決定することができる。また、マルチキャスト G の packets 番号において、 $G_i + 1 = G_{i+1}$ の関係が成り立っているため、任意の i に対して、

$$G'_{i-j} + 1 = G'_{i+1-j} \quad (3)$$

である。

図8に $r = 3, b = 12, d = 3$ の送信規則の例を示す。例えば、図8において、 t_1 もしくは t_1' からバッファリングを始めた場合、それぞれ t_2, t_2' でバッファリングが

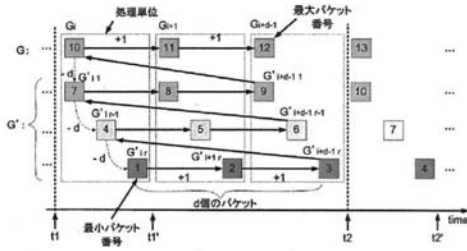


図8 提案手法によるパケット送信順序例 ($r = 3, b = 12, d = 3$)
 Fig. 8 Example of packet order in proposed rule ($r = 3, b = 12, d = 3$).

完了し、バッファリング時間は同じである。これはパケット送信規則に周期性があるためである。なお、STBは $1+r$ 個のパケット (G の1個のパケットと G' の r 個のパケット)を同時に受信することを(STBの処理単位で見れば)想定しているが、実際アクセラレータサーバから送信する G' のパケット順序は、 G'_{ij} ($j = 1, 2, \dots, r$)の r 個内であればどの順番でも良い(図8では降順である)。

以下では、図8を例に、本提案規則によってバッファリングしたパケットの番号がユニークかつ連続であることを説明する。STBは、 G から d 個、 G' から dr 個のパケットを受信しているとし、パケット番号をそれぞれ G_k と G'_{kj} ($k = i, i+1, \dots, r$)とする。これら $d(1+r)$ パケットの中で最小のパケット番号は、 G'_{ir} である。式3を再帰的に適用することにより、 G'_{ir} から G'_{i+d-1r} までのパケット番号が連続であることが分かる。さらに、式2, 3から、 $G'_{ir} + d - 1 = G'_{i+d-1r}$ と $G'_{i-1r} = G'_{ir} + d$ が成立するため、

$$G'_{i+d-1r} + 1 = G'_{i-1r}$$

が成り立つ。したがって、STBは G'_{ir} から G'_{i+d-1r} を経て G'_{i-1r} まで連続なパケットをバッファリングしている。同様に、 G'_{i-1r} から $G'_{i+d-1r-1}$ までの d 個のパケット番号も連続であり、 $G'_{i+d-1r-1} + 1 = G'_{i-2r}$ も成り立つ。そのため、これらの規則を再帰的に適用することで、 G'_{ir} から G'_{i+d-1r} までの dr 個のパケット番号が連続である。一方、 $G'_{i+1+d-1} = G'_{i+d-1r+1}$ と $G_i = G'_{i+1+d}$ が成立するため、 $G'_{i+d-1r+1} + 1 = G_i$ も成立する。また、明らかに、マルチキャスト G の G_i から G_{i+d-1} までの番号は連続である。以上より、 $d(1+r)$ 個の受信パケットの番号はユニークかつ連続であることが分かる。すなわち、STBは G のみから $b (= d(1+r))$ 個受信する通常のザッピングと同様に、本手法でも同じパケットを受信することができる。

図8の例では、 $\lfloor b/(1+r) \rfloor = b/(1+r)$ が成り立って

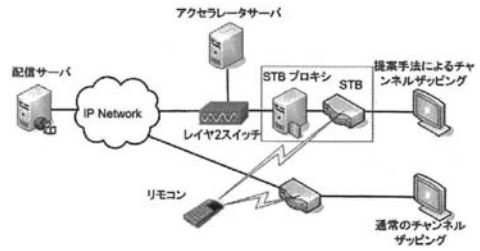


図9 検証ネットワーク構成
 Fig. 9 Test bed.

いるが、 $\lfloor b/(1+r) \rfloor \neq b/(1+r)$ の場合も考えられる。このとき、パケット送信順序はSTBの初期バッファサイズが $b' = \lfloor b/(1+r) \rfloor$ の場合と同じになる。STBでは $b' (> b)$ 個の連続するパケットを G の d 個のパケットに相当する時間で受信することができる。すなわち、STBは $b' - b$ 個の余分なパケットをバッファリングするだけであり、処理上特に問題ではない。

最後に、提案手法によるバッファリングの短縮時間について議論する。STBは、 G から1個パケットを受信する間に、 $1+r$ 個のパケットを G と G' から受信することができる。提案手法のバッファリング時間は、 G から $d (= \lfloor b/(1+r) \rfloor)$ 個のパケットを受信する時間であるため、本来必要なバッファリング時間に比べて、 $d/b (= \lfloor b/(1+r) \rfloor / b \approx 1/(1+r))$ に短縮することができる。

4. 性能評価

本節では提案した高速チャンネルザッピング手法を実装した試作ソフトウェアの評価を行う。本評価試験では、試作ソフトウェアを商用サービス[3]に適用し、そのときのチャンネルザッピング時間を比較する。検証ネットワーク構成を図9に示す。アクセラレータサーバは、レイヤ2スイッチ^(注1)に接続した。使用したSTBは、ザッピング要求に対して2つのマルチキャスト(G と G')にjoinする機能がなく、また受信パケットの順序並び替え機能がない。そのため、これら機能を代行するSTBプロキシを導入した。STBプロキシはザッピング時にSTBから送信される G へのjoinを検出すると、 G と G' へjoinする。また、受信した G と G' のパケットの並び替えを行い、初期バッファリングに必要なパケット数 b が揃うと、マルチキャストアドレス G を用いてSTBにパースト転送すると同時に G' からleaveする。なお、上記フレーム以外に対して、STBプロキシはレイヤ2ブリッジとして動作する。その他の条件を以下に示す。

(注1): 余計なマルチキャストのフラグディングを抑制するために、レイヤ2スイッチにIGMP Snooping [13] 機能を実装している。

表 1 チャンネルザッピング時間 (秒)
Table 1 Channel zapping delay(second).

	Average	Maximum	Minimum
Proposed scheme	1.32	2.1	0.8
Normal	1.83	2.2	1.3

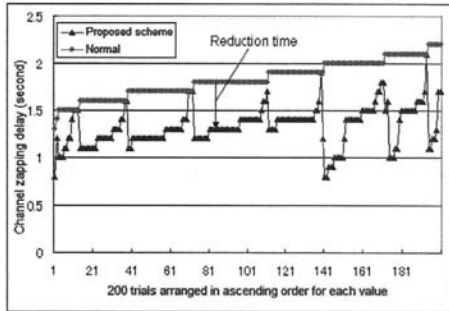


図 10 ザッピングの短縮時間

Fig. 10 Reduction time of channel zapping.

- マルチキャスト G の送信速度：約 6Mbps ($p = 500$ パケット/秒)
- コンテンツは MPGE2 により圧縮
- $r = 3, d = 100, b = 400$
- STB プロキシから STB への送信速度：約 42Mbps (マルチキャスト G に対して 7 倍速： $q = 7$)

STB プロキシを用いた本評価では STB プロキシから STB へのバースト転送が別途必要になるため、その分ザッピング遅延が増加する。

評価方法としては高速チャンネルザッピング手法を適用した STB と通常の STB を並列して置き、リモコンを用いて同時にザッピングを行い、実際に映像がディスプレイに映されるまでの時間を比較した。ザッピング時間は、1 秒当たり 30 フレームで撮影可能なビデオで測定した。200 回試行した結果を表 1 と図 10 に示す。図 10 は、各試行における 2 種の STB のザッピング時間の差をグラフ化したもので^(注2)、提案手法によるザッピング短縮効果を表している。通常の再生では、1.3~2.2 秒かかるところを本手法の導入により、0.8~2.1 秒に短縮することができた。また、ザッピングのタイミングによっては、最大で 1.1 秒程度の効果が見られた。一方で、4 回の試行においては短縮効果がなかった (0 秒短縮)。平均短縮時間は、0.51 秒である。本評価結果のばらつきは、1 フレームの受信タイミングならびに FEC の処理ブロックの受信タイミングによるものと思われる。なお、ザッピング

(注2)：横軸は、通常の STB 使用時、提案手法使用時の順に、ザッピング時間が短い方から試行を並び替えている。

完了後は 2 つの STB の間で再生のずれはほとんどなく、ほぼ同期していた。

5. 考 察

5.1 試作装置によるザッピング短縮時間の解析

4 節において、提案手法が商用サービスのザッピング時間を平均 0.51 秒短縮できることを確認した。本項では、評価試験における平均ザッピング短縮時間について考察する。

提案手法では、単に短縮バッファリング時間分だけザッピングが高速に行われる。そのため、理論的な平均ザッピング短縮時間は、

$$\begin{aligned} & (\text{平均ザッピング短縮時間}) \\ &= (\text{平均バッファリング短縮時間}) \\ &= \{r/(1+r) - 1/q\}b/p \end{aligned} \quad (4)$$

により与えられる。式 4 の各要素は、以下の通りである。

- $r/(1+r)$: $1+r$ 倍速による削減比率 (= 3/4)
- b/p : 通常ザッピングにおけるバッファリング時間 (= 400/500 秒)
- $1/q$: STB プロキシから STB へのバースト転送における時間増加比率 (= 1/7)

式 4 に上述の値を代入することで、理論値 0.49 秒を得ることできる。本評価における平均ザッピング短縮時間 0.51 秒は、理論値 0.49 秒と非常に近い値である。

5.2 帯域見積もり

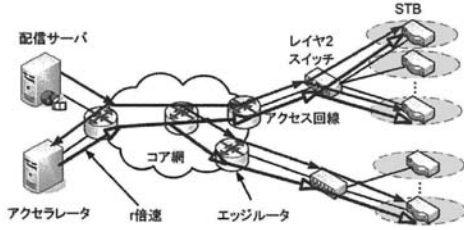
本項では、提案手法における帯域使用量について、従来手法との比較を行う。まず、 R, L, M, u を

- R : 1 チャンネルの使用帯域
- L : 総チャンネル数
- M : 評価地点より下流にある STB の数
- u : 従来手法のバースト転送時の総送信速度

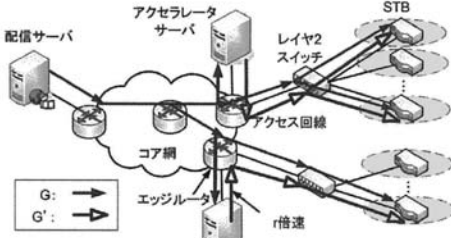
と定義する。1 回チャンネルザッピングを行った場合、提案手法では 2 つのマルチキャスト (G と G') を受信するため、 $(1+r)R$ のトラフィックが発生する。それに対し、従来手法では uR のトラフィックが発生する。 M 個の STB が同じチャンネルへ同時にザッピングを行った場合、提案手法では $(1+r)R$ 以上のトラフィックは発生しない。それに対して、従来手法では M に比例してトラフィックが増加する。したがって、非常に大きな M ($\gg L$) に対して、提案手法と従来手法で必要になる帯域は、それぞれ $(1+r)LR$ と uMR になる。提案手法は STB の数に比例せず一定の帯域で運用可能であり、スケラビリティや帯域見積もりが容易な点で優れている。

5.3 アクセラレータサーバの設置位置

本項では、アクセラレータサーバの設置位置について考



(a) Attachment to the same router as media server.



(b) Attachment to edge routers.

図 11 アクセラレータサーバの設置位置

Fig. 11 Variations of accelerator configuration.

察する。図 11 に、2つの設置形態の例を示す。いくつかの STB は、レイヤ 2 スイッチを介して、同じアクセス回線を共有している。当該スイッチでは *IGMP Snooping* が動作しており、各 STB は要求するマルチキャストのみ受信する。図 11 では、極端な例として、以下 2 つを挙げている。

- (a) 単一のアクセラレータサーバを配信サーバと同じルータに収容する。
- (b) 複数のアクセラレータサーバをプロバイダエッジのルータに収容する

設置形態 (a) では、1 台のアクセラレータサーバで全ての STB をサポートできることが利点である一方で、追加マルチキャスト G' がコア網に流れることが欠点となる。それに対して設定形態 (b) では、追加マルチキャスト G' がコア網を流れないため、リソースの面で優れているが、エッジルータの数だけアクセラレータサーバを設置しなければならない。

サーバとネットワーク負荷の点では、提案手法はどちらの設置形態でも適用は容易である。しかしながら、従来手法は、STB の数が増加すると設置形態 (a) での適用は困難である。

5.4 複数種類のザッピング速度への拡張

本稿では、追加マルチキャストとして 1 種類の倍速 r のみを検討してきた。しかしながら、要求されるザッピング速度やアクセス回線の帯域が全ての視聴者で同じとは限らない。そこで、本項では複数種類のザッピング速度への対応を検討する。

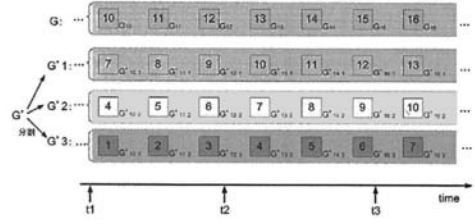


図 12 複数種類倍速の例 ($r = 3, b = 12, r' = 1or3$)

Fig. 12 Example of multiple acceleration rate ($r = 3, b = 12, r' = 1or3$).

一般に、ある自然数 n に対して、要求する追加マルチキャストの速度比 r' が、

$$1 + r = n(1 + r') \quad (5)$$

を満足すれば、元の追加マルチキャスト速度比 r との共存が可能である。ザッピング速度 r' を要求する STB は、 G_i と同じタイミングで、 G' のパケットの中から、 $G'_{i/n}, G'_{i/2n}, \dots, G'_{i/r'n}$ のみを受信すればよい。このとき、バッファリングは G から nd 個のパケットを受信する時間で完了する。 G' を以下 2 つのパケットを含むマルチキャストグループに分解し、 r' を所望する STB は、(1) の追加マルチキャストのみ受信すればよい。

- (1) $\{G'_{i/n}, G'_{i/2n}, \dots, G'_{i/r'n} | i = 1, 2, \dots\}$
- (2) $\{G' \text{ の残りのパケット } \}$

一方、複数のザッピング速度に対応するためには、 G' をより細かく分割する必要がある。例えば、式 5 を満足する全ての r' に対し共存を実現するためには、 G' を r 個のマルチキャスト $G'1, \dots, G'r$ に分割すればよい。このとき、ある分割マルチキャストグループ $G'j$ ($j = 1, 2, \dots, r$) の G_i ($i = 1, 2, \dots$) に対応するパケット番号は、 $G'_{i/j}$ ただ 1 つとする。STB は追加マルチキャストとして、 $G'_{i/n}, G'_{i/2n}, \dots, G'_{i/r'n}$ に join すればよい。図 12 に、図 8 と同様のパラメータ ($r = 3, b = 12, r' = 1$) での追加マルチキャストの分割例を示す。

図 12 において、ある STB は、 $r' = 1$ の速度比を所望し、一方、他の STB は、 $r' = 3$ を所望する場合を考える。前者の STB は、 $G'2$ のマルチキャストのみを受信すればよい。このとき、受信するパケット番号は、

$$G'_{i/2} = G_i - 2d \quad (6)$$

を満足しており、ザッピング時間は $t3 - t1$ で $1/2$ に短縮される。式 6 は、式 2 において、 d と r を nd ($= 2d$) ならびに r' に置き換えたものと同じである。すなわち、 $r = 1$ の場合と同じにパケットを受信している。他方、他の STB は $G'1, G'2, G'3$ の全ての分割された追加マルチキャストを受信することで、ザッピング時間が正しく $1/4$

(ザッピング時間: $t_2 - t_1$) に短縮される。以上のように、アクセス回線等が不均質な環境においては、追加マルチキャストを分割することによって複数のザッピング速度が実現可能である。

6. ま と め

本稿ではサーバやネットワーク負荷を考慮し、STBのバッファリングを追加マルチキャストを用いて加速することでIP放送のチャンネルザッピング時間を短縮する手法を提案した。さらに、よりザッピング時間を短縮するために、追加マルチキャストの転送速度を r 倍に拡張した。このとき、視聴者のザッピングタイミングによらず一定時間でバッファリングが完了するように、適切な追加マルチキャストのパケット順序を示した。

試作機による検証試験では、商用サービスにおいてザッピング時間を最大1.1秒程度削減できることを確認した。平均短縮時間は0.51秒で理論値と近い値であった。さらに、複数のチャンネルザッピング速度への拡張を考察した。

謝辞 日頃御指導頂く KDDI 研究所秋葉所長に感謝する。

文 献

- [1] GyaO, <http://www.gyao.jp/>.
- [2] 4th MEDIA, <http://4media.tv/>.
- [3] MOVIE SPLASH, <http://www.hikari-one.com/tv/index.html>.
- [4] <http://www.bbtv.com/>.
- [5] "地上デジタル放送の利活用の在り方と普及に向けて行政の果たすべき役割～2011年全面移行ミッションの確実な実現に向けて～," 情報通信審議会 平成16年諮問第8号 第2次中間答申, July 2005.
- [6] "著作権法の一部を改正する法律の制定について," http://www.bunka.go.jp/ityosaku/chosakukenhou/_kaisei.html.
- [7] Deering, S. E. "Multicast routing in internetworks and extended LANs," SIGCOMM Comput. Commun. Rev., Vol. 25, No. 1, pp. 88-101, 1995.
- [8] "Forward error correction (FEC) building block," IETF RFC3542, Dec. 2002.
- [9] "The use of forward error correction in reliable multicast," IETF RFC3543, Dec. 2002.
- [10] ISO/IEC 13818-2 (Mpeg2-Video), Information Technology. Coding of Moving Pictures and Associated Audio for Digital Storage Media at up about 1.5 Mbit/s: Video, 1993.
- [11] T. Wiegand, Joint Final Committee Draft (JFCD) of Joint Video Specification (ITU-T Rec. H.264 — ISO/IEC 14496-10 AVC), JVT-D157, August 2002.
- [12] <http://www.isma.tv/newsletter/2006-winter.html>.
- [13] "Challenges in the design and deployment of IP video delivery," Cisco Systems, 2006.
- [14] "Internet Group Management Protocol, Version 2," IETF RFC2236, Nov. 1997.
- [15] "Catalyst 2960 Switch Software Configuration Guide, Rel. 12.2(40)SE,"