

再生中断時間短縮のための 無線放送型ビデオオンデマンドシステム

松本 哲^{1,a)} 義久 智樹¹ 下條 真司¹

受付日 2021年7月15日, 採録日 2021年11月26日

概要: 近年のインターネットの普及により, ビデオオンデマンド (VoD) システムが大きな注目を集めている。従来の VoD システムでは, 映像データの再生端末の数が増えると, 再生中断時間が長くなる問題が発生する。このため, 放送型配信を活用した VoD システム (放送型 VoD システム) を活用した研究がいくつか行われている。放送型 VoD システムでは, 放送サーバは, 映像ファイルをいくつかの部分に分割し, 既定の放送スケジュールに従って分割されたデータを放送する。従来の放送方法では, 実環境で発生する初期バッファリングの遅延や映像ファイルの分割によるデータサイズの増加等を考慮していなかった。これらを考慮することにより, 再生中断時間をさらに短縮できる。そこで, 本研究では, 無線放送を用いた放送型 VoD システム (無線放送型 VoD システム) における分割データのスケジューリング手法を提案する。無線放送 (VHF 帯) を利用して放送型の配信を実現し, 実環境の無線放送型 VoD システムで提案手法の性能測定実験を行う。実験の結果, 映像データを 6 つの部分に分割し, 並行して放送することで, 比較手法と比べて再生開始までの待ち時間を最大 84%短縮できることを確認した。

キーワード: ビデオ配信, ストリーミング配信, ワイヤレス放送, 連続メディアデータ

A Wireless-Broadcast-Type Video-on-Demand System for Interruption Time Reduction

SATORU MATSUMOTO^{1,a)} TOMOKI YOSHIHISA¹ SHINJI SHIMOJO¹

Received: July 15, 2021, Accepted: November 26, 2021

Abstract: Due to the proliferation of the Internet in recent years, video-on-demand (VoD) systems has got great attention. In conventional VoD systems, a larger number of the clients receiving video data causes a longer playback interruption. Accordingly, several research focus on the VoD systems that adopt broadcast-type VoD. In broadcast-type VoD systems, video files are divided into several parts and distribution servers broadcast them in the sequences of predetermined broadcast schedules to reduce interruption times. However, conventional broadcasting methods do not consider the issues that occur in actual broadcast-type VoD systems, such as initial buffering delay and the increases of data sizes caused by video file divisions. Interruption time can be further reduced by considering them. Hence, in this research, we propose a data segment scheduling method for broadcast-type VoD systems using wireless broadcasting (wireless-broadcast-type VoD systems). We measure the performance of our proposed method on an actual wireless-broadcast-type VoD system utilizing radio broadcasting (VHF-band) to realize a broadcast-type distribution. The results of the experiments confirmed that, under the condition of an uninterrupted playback, the waiting time until the start of a playback can be reduced by up to 84% under our proposed method by dividing the requested video data into 6 parts and broadcasting the parts in parallel.

Keywords: video distribution, streaming delivery, wireless broadcasting, continuous media data

¹ 大阪大学サイバーメディアセンター
Cybermedia Center, Osaka University, Ibaraki, Osaka 567-0047, Japan

^{a)} smatsumoto@cmc.osaka-u.ac.jp

1. はじめに

近年の無線通信の高速化にともない, ビデオオンデマンド (VoD) システムに対する注目が高まっている。VoD シ

システムでは、利用者は、ホームページ等で公開されている映像リストの中から視聴したい映像を選択し、映像の初めから再生して視聴を開始する。利用者が映像ファイルをダウンロードしてから再生するビデオダウンロードシステムと比べて、利用者は映像ファイルのダウンロード完了を待つ必要がないこと等から、Amazon Prime Video や Netflix 等の多数の VoD サービスが開始されている。

VoD システムでは、映像を再生する再生端末は、配信サーバから映像データを受信しながら再生する。ある配信サーバから映像データを受信する再生端末の数が多くなると配信サーバの処理負荷や通信負荷が大きくなると、映像データの受信が再生に間に合わず途切れが発生したり再生開始までの待ち時間が長くなったりする。このため、放送型配信を活用した VoD システムが研究されている。放送型配信は、多数の端末にまとめてデータを配信できる配信形態である。これらの VoD システムでは、配信サーバは映像データを放送できるため、配信サーバの負荷がクライアント数に依存しなくなり、再生が中断する可能性が低くなる。本研究では、このような VoD システムを放送型 VoD システムと呼ぶ。放送型 VoD システムでは、映像ファイルをいくつかのセグメントに分割し、放送サーバが既定の放送スケジュールに従ってセグメントを放送して再生中断時間を削減する [1], [2], [3], [4], [5], [6]。しかし、従来の放送方式では、実環境の放送型 VoD システムで発生する以下の問題が考慮されていなかった。

- 初期バッファリング遅延
- 映像ファイルの分割によるデータサイズの増加

これらを考慮することにより、放送サーバは、クライアントが各セグメントの再生を終了する直前に、次のセグメントを受信するようにセグメントを効率的に配信できる。これにより、再生中断時間をさらに短縮できる。さらに、放送型 VoD システムに関するこれまでの研究では、シミュレーションプログラムを使用して性能を測定していた。実環境における性能測定により、放送型 VoD システムの実環境における有用性を確認できる。

そこで、本研究では、放送型 VoD システムのセグメントのスケジューリング手法を提案する。また、実環境で提案手法の性能を測定する。これまでにイントラネット上で放送型 VoD システムの実システム評価を行った研究はあるが、高価なハードウェアが必要であったり電波法上の制限から電波の発射が非常に困難であり、無線放送を用いた放送型 VoD システムである無線放送型 VoD システムは実現されていなかった。本研究では、無線放送 (VHF, Very High Frequency 帯) を利用して放送型配信を実現する。実験の結果、再生が途切れない状態で、要求された映像データを 6 つの部分に分割して放送することにより、提案手法では再生開始までの待ち時間を最大 84 % 短縮できることを確認した。本研究の貢献は、実環境を考慮したセグメン

トスケジューリング手法の提案と、無線放送型 VoD システムの実証実験、従来のシミュレーション評価と実環境評価の比較にある。

以下、2 章で関連研究を紹介し、3 章で無線放送型 VoD システムと提案方式について説明する。4 章で実システムの構成を説明し、5 章で評価実験結果を示す。6 章で議論を行い、最後に 7 章で本論文をまとめる。

2. 関連研究

VoD システムにおける再生待ち時間や途切れ時間を短縮する手法として、映像ファイルを複製配置する手法 [1] やピアツーピア技術を用いた手法 [2], [3], [4] が提案されている。これらの手法では、インターネット等の通信網を用いて各再生端末を 1 対 1 で通信する場合を考慮しており、再生端末の数が多くなると配信サーバの負荷が大きくなる問題がある。

文献 [7], [8], [9] では、マルチキャストを用いた 1 対多の通信を行って配信サーバの負荷を削減する手法を提案している。これらの手法では、マルチキャストを確実に実行できるように最低の帯域や再生端末へのデータ到達性が保証されている環境を想定している。実環境では、通信サービスの都合上マルチキャスト帯域を保証できなかったり、ルータでマルチキャストの転送を禁止している場合がある。

通信と、無線電波を用いた放送型配信を併用するシステムがいくつか提案されている [10], [11], [12]。これらのシステムでは、放送で映像を配信し、通信で関連する情報を配信している。しかし、オンデマンド型の映像配信を行えない。

我々の研究グループでは、放送チャンネルと通信チャンネルを用いて映像を配信する環境において、再生中断時間を効果的に削減する帯域割り当て手法を提案した [13]。しかし、1 章で述べた実環境の無線放送型 VoD システムで発生する問題を考慮していなかった。

3. 無線放送型 VoD

本章では、初めに無線放送型 VoD システムについて説明し、その後、無線放送型 VoD システムにおける映像データの放送手法に関して、3 つの手法を説明する。これらの手法は、本研究の実システムで実装する手法である。

3.1 無線放送型 VoD システム

無線放送型 VoD システムの想定環境を図 1 に示す。放送サーバは映像ファイルを保持しており、放送設備と接続されている。映像ファイルはいくつかの部分 (セグメント) に分割でき、放送サーバは、放送設備を介してセグメントを放送できる。放送範囲内にある再生端末は、放送設備が放送したセグメントを受信できる。再生端末は、受信したセグメントをそれぞれ再生できる。たとえば、映像ファイ

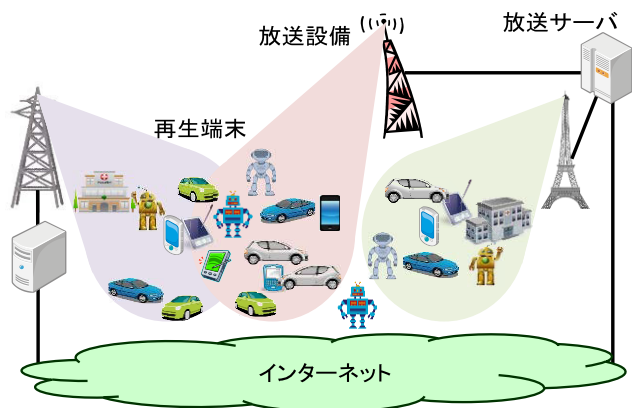


図 1 無線放送型 VoD システムの想定環境

Fig. 1 An assumed system environment for broadcast-type VoD systems.

ルを N 個のセグメント S_i ($i = 1, \dots, N$) に等分割する場合、再生端末は S_1 を受信完了すると映像の再生を開始できる。60[s] の映像を 6 個のセグメントに等分割する場合、 S_1 には映像の始めの 10[s] のデータが含まれることになる。

無線放送型 VoD システムでは、放送サーバは、各セグメントをチャンクと呼ぶさらに細かいデータ構造に分割して放送する。チャンクはセグメントより小さいデータサイズで構成される。チャンクを放送する際、放送サーバは、セグメント番号 (S_i の i) とチャンク番号 (S_i のデータの先頭側から各チャンクに順番に割り振った番号) を付加して放送する。これにより、再生端末はセグメントの放送中に受信を開始しても途中からチャンクを受信できる。

このような無線放送型 VoD システムの構築に必要な無線放送を実現できる放送型配信システムの例として地上波デジタル放送や、衛星放送があげられる。無線放送以外の放送型配信システムとして、ケーブルテレビ放送や帯域保証された IP プロトコルのブロードキャストアドレスやマルチキャストアドレスがあげられる。

映像ファイルを分割して順番に再生する映像符号化方式として、MPEG がある。MPEG では、映像ファイルは GoP (Group of Pictures) と呼ぶいくつかのフレームの集合で構成されており、再生端末は GoP ごとに映像を再生できる。また、HLS (HTTP Live Streaming) においても、映像ファイルは TS ファイルと呼ぶいくつかの映像データに分割され、再生端末は各 TS ファイルを独立して再生できる。GoP や TS ファイルは想定環境におけるセグメントに相当する。

3.2 従来の映像データ放送手法

本節では、無線放送型 VoD システムにおいて、再生端末が放送されているデータの受信を開始してから映像の再生を開始するまでの待ち時間 (再生待ち時間) を短縮する

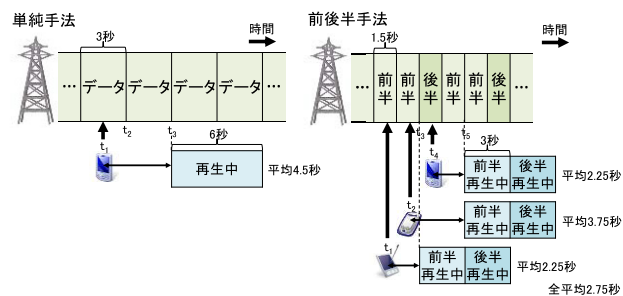


図 2 単純手法および前後半手法のイメージ

Fig. 2 Images of the simple method and the binary method.

従来の手法を説明する。初めに単純な放送手法を説明した後、単純手法と比べて待ち時間を短縮する手法を説明する。

(1) 単純手法

単純手法では、映像ファイルを繰り返して放送する。単純手法のイメージを図 2 に示す。放送設備は放送帯域 B を用いて同じ映像ファイルを繰り返して放送している。放送帯域は、放送サーバがデータの放送に利用できる可用帯域である。映像ファイルのデータサイズ D は、放送サーバが放送する映像ファイルのデータサイズである。セグメント数 N は映像ファイルの分割数と等しく、映像ファイルを構成するセグメントの個数である。たとえば、4.75 [Mbytes] の映像ファイルを 6 分割して 3.8 [Mbps] の帯域で放送する場合、 $B = 3.8$ [Mbps], $D = 4.75$ [Mbytes], $N = 6$ となる。以下に、単純手法の平均再生待ち時間を、 S_1 の放送中に受信を開始した再生端末と、それ以外の再生端末の平均再生待ち時間に分けて説明し、算出する (図 3)。

S_1 の放送中に受信を開始した再生端末 (図 3 の左側の端末) は、受信開始時より S_1 の内の後ろ側のチャンクを受信できる。この再生端末が S_1 を復元するためには、受信開始時および受信開始時より前側にある S_1 内のチャンクを受信する必要がある。単純手法では、各セグメントが順番に繰り返して放送されているため、この前側のチャンクを受信完了するまで D/B かかることになる。この D/B が、 S_1 の放送中に受信を開始した再生端末の、 S_1 の受信完了 (映像の再生が開始される) までにかかる平均時間、すなわち平均再生待ち時間になる。単純手法では、すべてのセグメントの放送に D/B かかり、 S_1 の放送には $D/(NB)$ かかるため、映像ファイルを 1 回放送する間のこのような再生端末の割合は $1/N$ になる。

次に、 S_1 以外のセグメントの放送中に受信を開始した再生端末 (図 3 の右側の端末) は、 S_1 の放送が開始されるまで平均して $(N - 1)D/(2NB)$ 待つことになる。これは、最長で $(N - 1)D/(NB)$ 、最短で待ち時間なしとなることから $\{(N - 1)D/(NB) + 0\}/2$ より算出できる。この再生端末は、 S_1 の放送が開始されると放送された S_1 の受信を開始する。 S_1 の受信を開始してから完了するまでの時間は $D/(NB)$ になる。このため、 S_1 以外のセグメントの

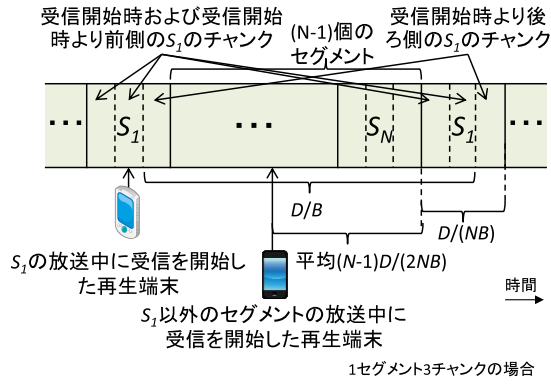


図 3 単純手法の平均再生待ち時間を説明するための図

Fig. 3 An image for explaining the waiting time under the simple method.

放送中に受信を開始した再生端末の平均再生待ち時間は、 $(N-1)D/2NB + D/NB = (N+1)D/(2NB)$ となる。 S_1 以外のセグメントの放送期間は、 $(N-1)D/(NB)$ であるため、映像ファイルを1回放送する間のこのような再生端末の割合は $(N-1)/N$ になる。

以上より、単純手法の平均再生待ち時間は、 $1/N \times D/B + (N-1)/N \times (N+1)D/(2NB)$ で与えられ、

$$\left\{ \frac{(N-1)(N+1)}{2N} + 1 \right\} \times \frac{D}{NB} \quad (1)$$

となる。

(2) 前後半手法

前後半手法では、セグメントを映像ファイルの前半に含まれるセグメント（前半セグメント群）と後半に含まれるセグメント（後半セグメント群）に分類して前半セグメント群を頻繁に放送する。前半セグメント群を頻繁に放送することで、再生端末が最初のセグメントを受信できる機会が増え、平均再生待ち時間を短縮できる。前後半手法のイメージを図2に示す。再生端末が前半セグメント群の再生中に後半セグメント群を受信完了できる条件を満たすように前半セグメント群の放送回数を決定することで、再生端末は前半セグメント群の再生終了後途切れることなく後半セグメント群を再生できる。

映像ファイルのビットレートを R とすると、判別式 A に応じて前半セグメント群を次に示す回数だけ放送した後後半セグメント群を1回放送すると、上記条件を満たしたうえで最短の再生待ち時間を与えることが分かっている [5].

$$\begin{aligned} a &= B/R \\ A &= [a] (a - [a]) - [a] \\ \text{if } A < 0: & [a] \text{ times} \\ \text{if } A > 0: & [a] \text{ times} \end{aligned} \quad (2)$$

図2の例は $A < 0$ の場合であり、前半セグメント群を $[a] = 2$ 回放送してから後半セグメント群を1回放送して

いる。再生が途切れなくなるまで待つから再生する場合の平均再生待ち時間は次式で与えられる。

$$\begin{aligned} \text{if } A < 0: & \frac{\alpha + 3}{2\alpha + 2} \times \frac{D}{B} \\ \text{if } A > 0: & \frac{3\alpha + 6 - 2[\alpha]}{2\alpha + 4} \times \frac{D}{B} \end{aligned} \quad (3)$$

3.3 提案手法

提案する Initial Buffering based Parallel Broadcasting (IBPB) 法では、初期バッファリングのデータサイズを考慮して映像ファイルをいくつかのセグメントに分割し、それらを並行して放送する。(IBPB)方式と呼ぶ。下記で詳細を説明する。

(1) IBPB法の再生待ち時間を短縮するアイデア

IBPB法では、映像ファイルを N 個のセグメントに分割する。無線放送型 VoD システムでは、クライアントは、初期バッファリングのために映像データの初めの方を受信、保存した後に再生を開始するため、IBPB法では、最初のセグメントのデータサイズが初期バッファリングのデータサイズと同じになるように、映像ファイルを分割する。

また、IBPB法では、各セグメントを周期的に放送して再生待ち時間を短縮するために、放送サーバはセグメントを並列に放送する。このため、放送帯域幅も N 個の放送チャンネルに分割され、各セグメントは各放送チャンネルで周期的に放送される。

(2) IBPB法で映像ファイルを分割する方法

IBPB法では、映像ファイルをセグメント S_i ($i = 1, \dots, N$) に分割し、放送帯域を N 等分した N 個の放送チャンネルを用いて放送する。放送チャンネル i で S_i を繰り返して放送する。IBPB手法のイメージを図4に示す。再生端末が S_i の再生を終了するまでに S_{i+1} を受信完了できる条件を満たすように各セグメントのデータサイズを決定することで、再生端末は途切れることなく最初から最後まで再生できる [6].

S_{i+1} のデータサイズを D_{i+1} で表す。各放送チャンネルの放送帯域は B/N となるため、 S_{i+1} の放送には ND_{i+1}/B かかることになる。 S_i の再生終了まで $(D_1 + \dots + D_i)/R$ かかることから、 D_{i+1} は次式で与えられる。

$$D_{i+1} = \frac{B}{NR} \sum_{j=1}^i D_j \quad \left(\text{if } \sum_{j=1}^{i+1} D_j < D \right) \quad (4)$$

これより、 D_1 を決定すると D_2 が決定され、 D_2 を決定すると D_3 が決定されるといったように、 D_1 を決めると順番に各セグメントのデータサイズが決定される。IBPB法では、次式を満たす D_1 を算出する。

$$\sum_{j=1}^N D_j = D \quad (5)$$

すなわち、すべてのセグメントのデータサイズの合計が映

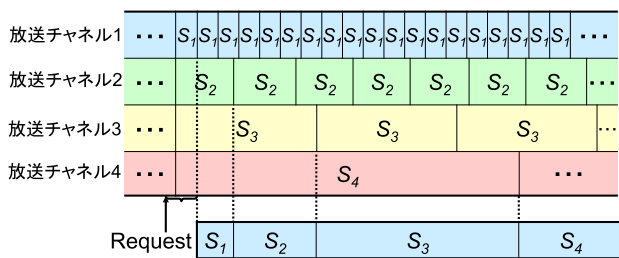


図 4 IBPB 法のイメージ
Fig. 4 An image of the IBPB method.

像ファイルのデータサイズを等しくなる。ただし、式 (5) を満たす D_1 が初期バッファリングのデータサイズより小さくなる場合には、バッファリングに必要なデータが 2 番目以降のセグメントにも含まれて受信に時間がかかるため、 D_1 を初期バッファリングのデータサイズと同じデータサイズとする。これにより、分割数に上限ができる。

映像ファイルを分割すると、分割した部分が分割前の映像ファイルのどの部分のデータかを示す識別子が付加されるためにデータサイズが増加する。IBPB 法では、映像ファイルを分割する際、実際に分割した後のデータサイズを用いて式 (5) を満たすように分割することで、この問題を解決する。これにより、データサイズの増加分を考慮して分割できるため、IBPB 法では問題点を解決したうえでセグメントを放送できる。たとえば、ffmpeg (様々な映像関連処理を行えるソフト [14]) を用いて映像ファイルを分割する場合には、各セグメントの再生時間が必要になる。ただし、3.1 節で説明した GoP 構造の都合で必ずしも与えた再生時間になるとは限らない、本論文における実証実験では、再生時間を与えて分割する操作を繰り返し、式 (5) を満たすように D_1 を決定した (ただし、初期バッファリングのデータサイズ以上)。

平均再生待ち時間は、 S_1 を受信完了するまでの平均時間と等しく、

$$\frac{ND_1}{B} \tag{6}$$

となる。

4. 無線放送型 VoD システムの設計と実装

本章では、設計および実装した無線放送型 VoD システムについて説明する。

4.1 システム設計

図 5 に設計した無線放送型 VoD システムの構成図を示す。上部が送信側、下部が受信側を示す。

送信側では、放送サーバは映像データをチャンクと呼ぶ単位に区切って MPEG エンコーダに送信する。チャンクのデータサイズと送信間隔を変更することで、MPEG エンコーダへのデータ送信速度を調整できる。MPEG エンコーダへの送信には UDP パケット (マルチキャストアド

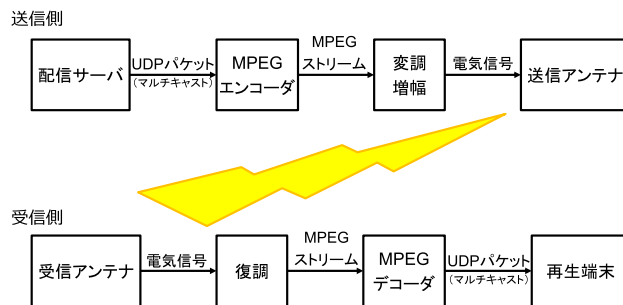


図 5 設計した無線放送型 VoD システム構成図

Fig. 5 An architecture of our designed wireless-broadcast-type VoD system.

レス) を用いる。MPEG エンコーダは、受信した UDP パケットを電波で放送するために、放送波で IP パケットを送信する IPDC (IP Data Cast) を用いる。MPEG エンコーダは、IPDC を行うための MPEG ストリームを作成し、変調器に送信する。変調および増幅された電気信号は、送信アンテナを介して電波として射出される。

受信側では、射出された電波を受信アンテナで受信し、復調して MPEG ストリームを得る。MPEG デコーダで UDP パケットを抽出し、再生端末に送信する。再生端末は、UDP パケットからチャンクを復元し、各セグメントを構成するチャンクをすべて受信すると、そのセグメントを保存する。再生端末は、各セグメントを再生する時刻になると再生する。図 5 では、受信側は 1 つのみ掲載したが、複数であっても動作する。

4.2 システムの実装

本節では、実装した無線放送型 VoD システムを説明する。

(1) 配信ソフトウェア

配信ソフトウェアは、映像ファイルを放送するための、放送サーバで動作するソフトウェアである。Windows 10 で実装した配信ソフトウェアのスクリーンショットを図 6 (上) に示す。

スクリーンショットの左側は配信に関する設定を行う部分である。UDP パケットを送信するためのサーバアドレスやポート番号、マルチキャストアドレスを指定できる。また、3.2 節で説明した放送手法を選択し、各手法のパラメータを設定できる。映像ファイルの場所やセグメントの数も指定できる。放送帯域を調整するためのチャンクの送信間隔やチャンクのデータサイズ、UDP パケットのデータサイズも指定できる。中央上部にあるのは動作確認のためのログ出力部分である。スクリーンショットの右側は評価実験を行うために、各パラメータを自動的に変更する範囲を指定する部分である。

(2) 受信ソフトウェア

受信ソフトウェアは、映像データを受信して再生するための、再生端末で動作するソフトウェアである。Windows

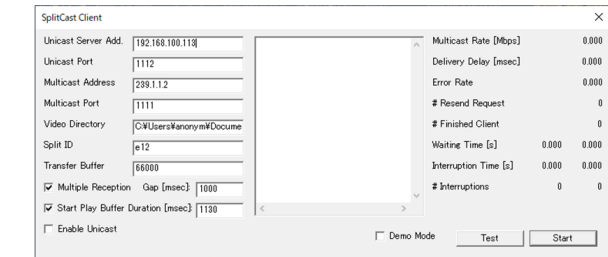
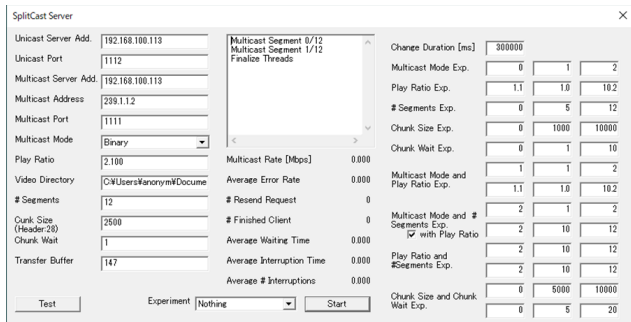


図 6 実装した配信ソフト (上) および受信ソフト (下) のスクリーンショット

Fig. 6 Screenshots of our developed distribution software (top) and reception software (bottom).

10 で実装した受信ソフトウェアのスクリーンショットを図 6 (下) に示す。

スクリーンショットの左側は受信に関する設定を行う部分である。UDP パケットを受信するためのマルチキャストアドレスやポート番号、映像ファイルの保存先を指定できる。評価実験のためのモードの切り替えも行える。中央にあるのは動作確認のためのログ出力部分である。スクリーンショットの右側は放送帯域や遅延、エラー率といった受信に関する情報を表示する部分である。

映像ファイル構成として HLS を用い、TS ファイルがセグメントに相当する。受信した映像ファイルの再生には、ffmpeg を用いた。ffmpeg は、多数の映像エンコードに対応して映像再生できることから、ffmpeg と共に研究開発の分野で多用されている。ffmpeg では、HLS の再生に TS ファイルのバッファリングが必要なため、受信ソフトではバッファリング時間を指定できるようにした。

(3) ハードウェア

実装した無線放送型 VoD システムのハードウェアの写真を図 7、これらの詳細を表 1 に示す。これらのハードウェアを 3.2 節の設計に基づいて接続した。

5. 評価実験

本章では、評価実験について記述する。まず、各手法の再生待ち時間と途切れ時間を示す。その後、以下の点に焦点を当ててシミュレーション結果と実際の結果を比較する。

- 実現可能な放送帯域
- 実処理にともなう配信遅延
- 映像データのバッファリング



図 7 ハードウェアの写真

Fig. 7 Hardware photos.

表 1 ハードウェア一覧

Table 1 Hardware list.

放送サーバおよび再生端末 (2 台)	VAIO 社製 VAIO OS: Windows 10 Home CPU: Intel Core i7 メモリー: 16GB SSD: 256GB NIC: Realtek PCIe GbE Family Controller
MPEG エンコーダ	ASTRO 社製 CX-5528
変調器	栄電社製 MSH3000A 周波数: JP 12ch (219MHz)
増幅器	栄電社製 4034A 10mW→1W
送信アンテナ	基地局用ダイポールアンテナ
受信アンテナ	マスプロ電工社製 162E3-P
復調器 兼 MPEG デコーダ	SK ネット社製 IPDC 受信機

5.1 評価実験環境

評価実験では、4.2 節で説明した実システムを用いた。評価のための映像ファイルを作成するために、[15] で公開されている 4K ステレオ映像の Buck BUNNY を用い、放送帯域が約 3.8 [Mbps] であったことから、60 [s] の 650 [Kbps] (480 × 270, 20 [fps]) の映像に ffmpeg でトランスコードして利用した。これらの値の理由については 6.3 節で詳しく説明する。また、5.3 節で再生時間およびビットレートを変更した場合の評価を行っている。チャンクの送信間隔は 1 [ms]、チャンクサイズは 7.2 [Kbytes] とした (詳細は 5.5 節)。前後半手法や IBPB 法におけるセグメントへの分割については、あらかじめ算出した各セグメントの再生時間を指定して ffmpeg を用いて分割した。初期バッファリングのデータサイズは 1 [s] 相当とした (詳細は 5.4 節 (3) に記述)。

V-High 帯の電波発射については、V-High 特定実験試験局免許を保有する IPDC フォーラムの協力を得た。実験

は、2020年8月5日午後到大阪大学サイバーメディアセンター豊中教育研究棟7階会議室にて行った。

5.2 再生待ち時間と途切れ時間

本節では、実装した放送手法を用いた場合の再生待ち時間と途切れ時間を掲載する。前後半手法については、前半セグメント群の放送回数を変化させて計測したため、まず初めに単純手法とIBPB法の結果を示し、次に前後半手法の結果を示す。

計測には、時刻0[s]で放送サーバの放送開始と同時に再生端末が受信を開始し、再生端末が映像の再生を開始すると、その1[s]後に他の再生端末が受信を開始するものとし、10分間計測した。

(1) 単純手法とIBPB手法

単純手法およびIBPB法では、映像ファイルの分割数であるセグメントの数が再生待ち時間に影響を及ぼす。そこで、セグメントの数を変化させて平均再生待ち時間を計測した。なお、単純手法ではセグメントを単純に順番に繰り返して放送している。

結果を図8に示す。Actualが実計測結果、Simulationがシミュレーション結果である。シミュレーション結果は式(1)および式(5)から得た。Simpleは単純手法、IBPBはIBPB法を示す。横軸がセグメント数、縦軸が平均再生待ち時間である。IBPB法では、 S_1 のデータサイズを初期バッファリングのデータサイズである1[s]で与えたために、最大で6分割までであるため、グラフが6分割までとなっている(式(4)のif条件を満たす最大の*i*が5)。

セグメント数が多くなるほど平均再生待ち時間が短くなること分かる。これは、セグメント数が多くなるほど最初のセグメントのデータサイズが小さくなって受信にかかる時間が短くなるためである。再生端末は最初のセグメントを受信完了すると映像の再生を開始するため、セグメント数が多いほど平均再生待ち時間が短くなる。しかし、セグメント数が多くなるほどその短縮量は小さくなる傾向にあること分かる。これは、セグメントのデータサイズがセグメント数に反比例して小さくなるためである。本実験環境では、放送帯域が映像の再生レートより大きいため、単純手法では映像の再生中に途切れが発生することはない。また、IBPB法においても、式(4)で途切れが発生しないように各セグメントのデータサイズを与えており、途切れが発生することはない。

たとえば、映像ファイルを6分割して放送する場合(短縮率最大)、単純手法の平均再生待ち時間は6.45[s]、IBPB法の平均再生待ち時間は1.01[s]となり、最大で84%短縮できること分かる。

(2) 前後半手法

前後半手法における前半セグメント群の放送回数を変化させて平均再生待ち時間および途切れ時間を計測した。

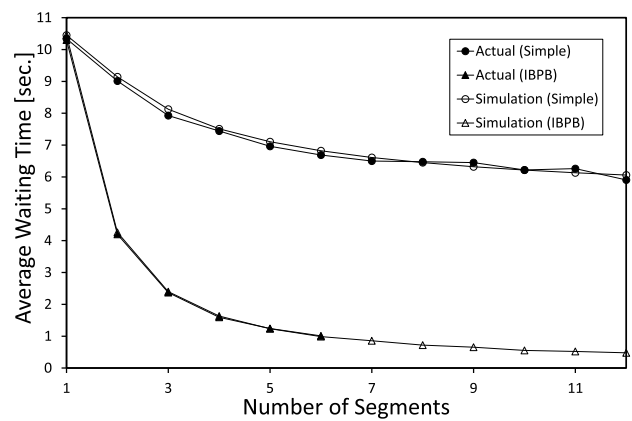


図8 単純手法およびIBPB法における平均再生待ち時間
Fig. 8 Average waiting time under the simple method and the IBPB method.

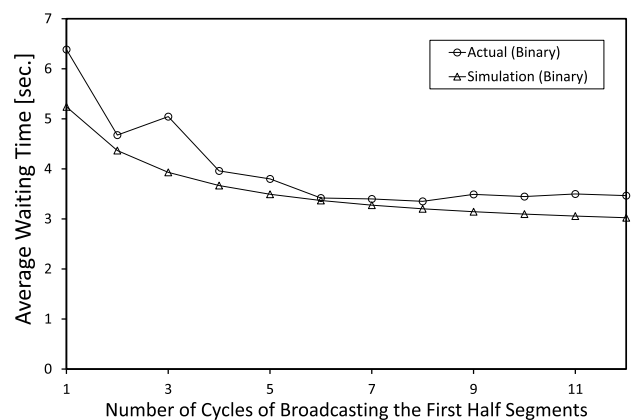


図9 前後半手法における平均再生待ち時間
Fig. 9 Average waiting time under the binary method.

平均再生待ち時間の結果を図9に示す。横軸が前半セグメント群の放送回数、縦軸が平均再生待ち時間である。このグラフより、前半セグメント群を多く放送するほど平均再生待ち時間が短くなる傾向にあること分かる。これは、前半セグメント群が頻繁に放送され、再生端末が S_1 を受信できる機会が増加するためである。

平均途切れ時間の結果を図10に示す。このグラフより、前半セグメント群の放送回数が6より大きくなると途切れが発生し、その後比例して途切れ時間が長くなっていること分かる。これは、前半セグメント群の放送回数が多くなるほど後半セグメント群の放送間隔が長くなって、再生端末が前半セグメント群の再生中に後半セグメント群を受信できる確率が小さくなるためである。

再生途切れが発生しない最大の前半セグメント群の放送回数は6であり、この場合の平均再生待ち時間は3.41[s]である。単純手法より短い、IBPB法より長いこと分かる。

5.3 再生時間とビットレート

平均再生待ち時間は映像の再生時間およびビットレート

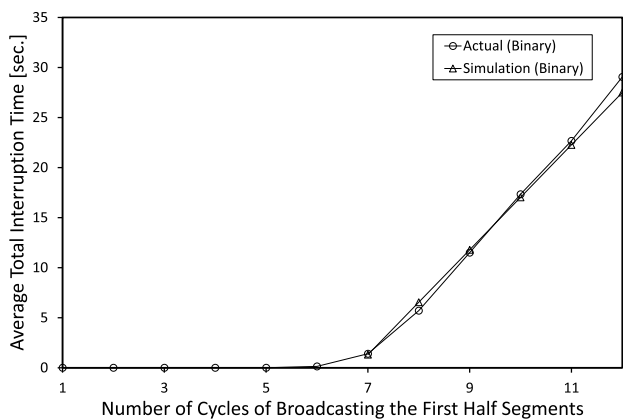


図 10 前後半手法における平均途切れ時間

Fig. 10 Average interruption time under the binary method.

に依存するため、その影響を調査した。なお、本節における評価結果は、実環境実験の後日行ったため、シミュレーションによる評価結果である。ただし、図 8 に示すとおり、分割数が 6 以下の場合には単純手法および IBPB 法において、実性能とシミュレーション性能に大きな違いがないことを確認している。

図 11 に映像の再生時間を変えた場合の平均再生待ち時間を示す。放送帯域は 3.8 [Mbps]、ビットレートは 650 [Kbps] とした。単純手法および IBPB 法で分割数を変更した場合の平均再生待ち時間は図 8 に示しているため、代表例として分割数が 2 ($N = 2$) および 6 ($N = 6$) の場合をシミュレートした。前後半手法では、途切れが発生しないように前半セグメント群の放送回数を式 (2) で与えた。横軸が映像の再生時間、縦軸が映像の再生が開始されるまでの平均再生待ち時間である。この結果より、すべての場合において平均再生待ち時間が再生時間に比例することが分かる。単純手法および前後半手法では、式 (1) および式 (3) で平均再生待ち時間が与えられるため、再生時間に比例することが分かる。IBPB 法では、再生時間が長くなるほど式 (5) を満たす D_1 が大きくなるため、平均再生待ち時間が長くなる。この結果より、手法間の平均再生待ち時間の順位は再生時間に依存していないことが分かる。

次に、図 12 にビットレートを変えた場合の平均再生待ち時間を示す。放送帯域は 3.8 [Mbps]、再生時間は 60 [s] とした。先ほどと同じく、代表例として分割数が 2 ($N = 2$) および 6 ($N = 6$) の場合をシミュレートし、前後半手法では前半セグメント群の放送回数を式 (2) で与えた。横軸がビットレート、縦軸が平均再生待ち時間である。ビットレートが大きくなるほど、映像ファイルのデータサイズが大きくなって平均再生待ち時間が長くなっている。2 分割の IBPB 法の平均再生待ち時間が、6 分割の単純手法や前後半手法の平均再生待ち時間より長くなるビットレートがあるが、これは分割数が少ないためであり、6 分割の IBPB 法の平均再生待ち時間は、これらよりも短くなっている。

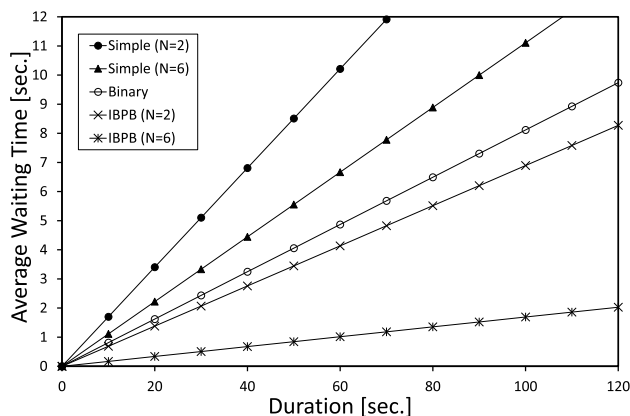


図 11 映像の再生時間と平均再生待ち時間

Fig. 11 Video duration and average waiting time.

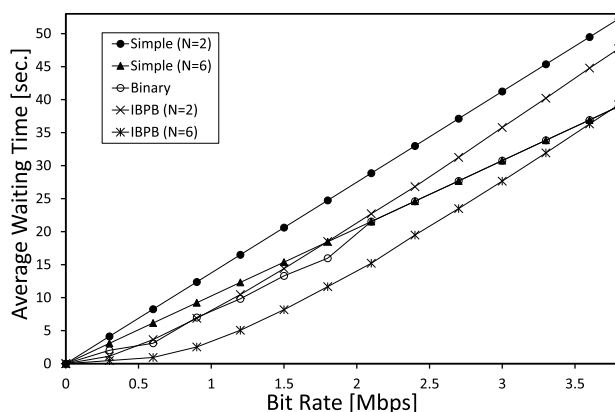


図 12 ビットレートと平均再生待ち時間

Fig. 12 Bit rate and average waiting time.

前後半手法では、前半セグメント群の放送回数が変わると平均再生待ち時間が大きく変化するため、段階的に変化しているように描かれている。前後半手法では、ビットレートが大きくなると単純手法と同じ順番でセグメントを放送することになるため、平均再生待ち時間が単純手法と同じになっている。この結果より、各手法で最短の平均再生待ち時間の順位はビットレートに依存していないことが分かる。

5.4 シミュレーションとの差異

本節では、実システム評価で明らかになった、シミュレーション結果との差異を説明する。

(1) 達成可能な放送帯域幅

シミュレーションでは様々な放送帯域を設定できるが、実システムでは、ハードウェアやソフトウェアの処理性能等の都合から利用できる放送帯域には上限がある。そこで、実現可能な放送帯域を調査した。チャンクのデータサイズと送信間隔を変更して放送帯域を変化させた。

調査の結果、放送サーバが UDP パケットを送出する速度が 3.8 [Mbps] を超えると、復調器から UDP パケットが送出されない現象が発生した。このときのチャンクサイズ

は 7.2 [Kbytes], 送信間隔は 1 [ms] である。これは、復調器の復調やパケット送出といった処理が受信に間に合わず、復調器内で動作するソフトウェアが停止したためであった。このことから、評価実験環境における最大放送帯域は 3.8 [Mbps] であることが分かった。

(2) 実処理にともなう配信遅延

シミュレーションでは変調や復調といった処理にともなう遅延を考慮していなかったが、実システムでは、これらの処理にともなう配信遅延が発生する。そこで、実処理にともなう配信遅延を計測した。チャンクサイズは 7.2 [Kbytes], 送信間隔は 1 [ms] とし、6 分割の IBPB 法を用い、映像データを 10 分間放送する間の配信遅延を計測した。

計測した結果、最大で 155 [ms], 最小で 81 [ms], 平均すると 135 [ms] の配信遅延が発生していることを確認した。実処理のともなう配信遅延により、シミュレーションと比べて再生待ち時間が平均 135 [ms] 程度長くなることになる。実測とシミュレーションの値の差異は、放送帯域の揺らぎ等によっても変化するため、必ずしもこの配信遅延になるとは限らない。このため、図 8 では大きく違わないように見える。なお、電波は光速 (約 30 万 [km/s]) で伝搬するため、送信アンテナと受信アンテナの距離が長くても配信遅延には大きく影響しない。

(3) 映像データのバッファリング

シミュレーションでは映像データのバッファリングを考慮していなかったが、実際のほとんどの再生ソフトでは、ある程度映像データをバッファリングしてから再生する。バッファリングにかかる時間が再生待ち時間に影響する。そこで、バッファリングに必要な時間を調査した。

調査の結果、実装した無線放送型 VoD システムでは、約 1 [s] のバッファリング時間が必要なことが分かった。これは、再生ソフトとして用いた ffmpeg で TS ファイルを再生する際に必要なバッファリング時間と考えられる。

5.5 チャンクの影響

実装した無線放送型 VoD システムでは、チャンクと呼ぶ単位で変調および復調が行われる。チャンクに関するパラメータはチャンクサイズ (チャンクのパケットサイズ) とチャンクの送信間隔がある。チャンクの送信間隔は、あるチャンクの送信が完了した後、次に送信するチャンクの送信を開始するまでの時間である。セグメントはチャンクで構成されるため、セグメントの送信間隔 (あるセグメントの最後のチャンクの送信が完了してから次に送信するセグメントの最初のチャンクの送信を開始するまでの時間) にも相当する。チャンクの送信間隔を短くするほど、放送サーバがチャンクを送信しない時間が短くなるため、送出速度 (単位時間あたりに放送するデータ量) が大きくなる。セグメントは複数のチャンクで構成されるため、セグメントの

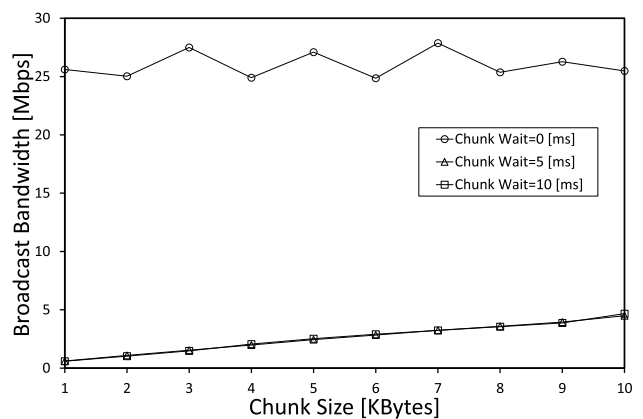


図 13 チャンクと放送帯域

Fig. 13 Chunk and broadcast bandwidth.

送信時間が短くなることになる。また、チャンクサイズを大きくするほど、間隔を空けずにまとめて送信するデータ量が多くなるため、送出速度が大きくなる。このように、チャンクサイズおよびチャンクの送信間隔に依存して送出速度が変化し、これに応じて再生待ち時間も変化する。

本節では、実装したシステムで実現できる放送帯域を調査するために、これらのパラメータを変化させた。復調器を介すると復調器の処理性能で放送帯域が制限されるため、本実験では変調復調を行わずに放送帯域と再生待ち時間を計測した。なお、提案手法とチャンクの影響には直接的な関連性はなく、提案手法の実環境実験にともなって得られたチャンクサイズおよびチャンクの送信間隔と放送帯域に関する知見である。

実現できた放送帯域を図 13 に示す。横軸がチャンクサイズ、縦軸が放送帯域である。凡例中の Chunk Wait はチャンクの送信間隔を意味する。送信間隔が 0 [ms] の場合、チャンクサイズが変わっても放送帯域が大きく変わらないことが分かる。これは、チャンクの送信間隔がなく連続的にチャンクが送信されるためにチャンクサイズの影響が出にくいためである。また、送信間隔が 5 [ms] と 10 [ms] で放送帯域に大きな違いがないことが分かる。チャンクの送信間隔を設けると、最小でも 10 [ms] 以上送信間隔が発生し、5 [ms] に設定しても実際には 10 [ms] 以上の送信間隔になっているためと考えられる。

次に平均再生待ち時間を図 14 に示す。送信間隔が 0 [ms] で放送帯域が大きいほどセグメントを短時間で放送できるため、平均再生待ち時間が短くなっている。送信間隔が 5 [ms] と 10 [ms] で大きく変化がないのは放送帯域で変化がなかった理由と同じである。

これらの結果より、チャンクサイズを大きくし、かつ、チャンクの送信間隔を短くするほど、データの送出速度が大きくなって平均再生待ち時間を短縮できる。しかし、実機での評価では抄出速度が放送帯域に制限されるため、チャンクサイズには上限、チャンクの送信間隔には下限が

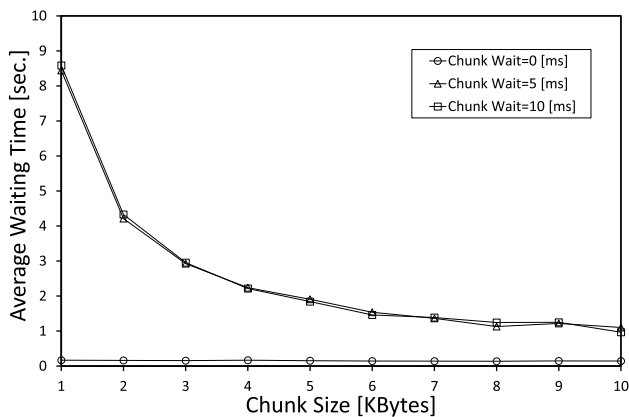


図 14 チャンクと平均再生待ち時間

Fig. 14 Chunk and average waiting time.

ある。これらのため、本論文における実環境評価では、チャンクサイズを 7.2 [Kbytes]、チャンクの送信間隔を 1 [ms] とした。

6. 議論

6.1 分割数について

シミュレーションでは、映像ファイルを分割するほど平均再生待ち時間を短縮できたが、実システムでは、分割できる最小のデータサイズがあり、そのデータサイズより小さいセグメントには分割できないことが分かった。分割数が多いほど平均再生待ち時間を短縮できる傾向にあるため、 S_1 のデータサイズが、この最小のデータサイズになるように分割することで平均再生待ち時間を最も短縮できると考えられる。 S_1 のデータサイズが初期バッファリングのデータサイズより小さくなると、初期バッファリングに時間がかかって平均再生待ち時間が長くなるため、IBPB 法では S_1 のデータサイズが初期バッファリングのデータサイズより小さくならないように映像ファイルを分割する。これにより、分割数に上限ができ、今回の実験環境では 6 分割までとなった。

6.2 処理負荷について

評価実験で用いた IPDC では、MPEG のエンコードやデコード、変調復調にともなう処理により遅延が発生することが分かった。遅延時間は 0.1 [s] 程度であり、この程度の長さで平均再生待ち時間に影響する。また、大きい放送帯域を確保するほど処理負荷が大きくなるため、利用できる放送帯域に上限があることが分かった。このため、実システムではハードウェアやソフトウェアの性能を考慮したうえで放送帯域を確保する必要がある。

6.3 再生時間およびビットレートについて

提案する IBPB 法では、各セグメントの受信にかかる時間より再生にかかる時間が長いほど平均再生待ち時間を

短縮できる。放送帯域と同じビットレートの映像ファイルであっても、分割することで平均再生待ち時間を短縮できるが、図 12 の 6 分割の場合のように単純手法と同じ平均再生待ち時間になる。IBPB 法が単純手法より平均再生待ち時間を短縮できる場合として、評価実験では、帯域幅より小さいビットレートの映像ファイルを用いた。放送型ビデオオンデマンドの研究分野では一般的な環境である [5], [6], [7], [8], [9]。

評価実験では、映像を分割せずに放送した場合でも少し待てば再生が開始される状況の一例として、平均再生待ち時間が 10 [s] になる状況を想定した。すなわち、実現できた放送帯域 3.8 [Mbps] で 10 [s] で映像ファイルが放送できる状況となる。これより、映像ファイルのデータサイズを $3.8 [\text{Mbps}] \times 10 [\text{s}] / 8 = 4.75 [\text{Mbytes}]$ とした。また、緊急地震速報等の短いニュース映像を放送することを想定し、映像の再生時間を 60 [s] とした。これらより、映像のビットレートを $4.75 [\text{Mbps}] \times 8 / 60 [\text{s}] = 633 [\text{Kbps}]$ とした。この設定で映像をトランスコードしたところ、650 [Kbps] の 60 [s] の映像となった。このため、評価実験では、基本的には 60 [s] の 650 [Kbps] の映像ファイルを用いた。

7. おわりに

本研究では、無線放送型 VoD システムにおけるセグメントスケジューリング手法を提案した。実装した V-High 帯を用いた無線放送型 VoD システムにより、再生待ち時間や途切れ時間といった性能計測を実システムで行った。性能計測の結果、これまでのシミュレーション結果とは異なり、実現可能な放送帯域や実処理にともなう配信遅延が発生し、また、映像データのバッファリングが必要になることが明らかになった。評価実験の結果、再生が途切れないという条件のもと、映像データを 6 分割して並列に放送することで、再生が開始されるまでの待ち時間を最大で 84% 短縮できることが明らかになった。

今後、再生端末がインターネットからも映像データを受信する放送通信融合環境で実験することや、選択型コンテンツや VR 放送といったインタラクティブ性のあるコンテンツの放送型配信を考えている。

謝辞 本実験に関し、多大なるご協力をいただいた IPDC フォーラムの諸氏に衷心より感謝申し上げます。本研究の一部は、G-7 奨学財団研究開発助成事業および科学研究費補助金 (18K11316 および 20K11829) の研究助成による。謹んで感謝の意を表する。

参考文献

- [1] Fratini, R., Savi, M., Verticale, G. and Tornatore, M.: Using Replicated Video Servers for VoD Traffic Offloading in Integrated Metro/Access Networks, *Proc. IEEE International Conference on Communications (ICC)*, pp.3438–3443 (2014).

- [2] Zhang, G., Liu, W., Hei, X. and Cheng, W.: Unreeling Xunlei Kankan: Understanding HybridCDN-P2P Video-on-Demand Streaming, *IEEE Trans. Multimedia*, Vol.17, No.2, pp.229-242 (2015).
- [3] Sheshjavani, A.G., Akbari, B. and Ghaeini, H.R.: An Adaptive Buffer-Map Exchange Mechanism for Pull-based Peer-to-Peer Video-on-Demand Streaming Systems, *Springer International Journal of Multimedia and Applications*, Vol.76, No.5, pp.7535-7561 (2016).
- [4] Araniti, G., Scopelliti, P., Muntean, G.-M. and Lera, A.: A Hybrid Unicast-Multicast Network Selection for Video Deliveries in Dense Heterogeneous Network Environments, *IEEE Trans. Broadcasting, Early Access*, 11 pages (2018).
- [5] Yoshihisa, T., Tsukamoto, M. and Nishio, S.: A Scheduling Scheme for Continuous Media Data Broadcasting with A Single Channel, *IEEE Trans. Broadcasting*, Vol.52, No.1, pp.1-10 (2006).
- [6] Yoshihisa, T.: Data Piece Elimination Technique for Interruption Time Reduction on Hybrid Broadcasting Environments, *Proc. IEEE International Conference on Pacific Rim Conference Communications, Computers and Signal Processing (PACRIM)*, 6 pages (2017).
- [7] Guom, J., Gong, X., Liang, J., Wang, W. and Que, X.: An Optimized Hybrid Unicast/Multicast Adaptive Video Streaming Scheme Over MBMS-Enabled Wireless Networks, *IEEE Trans. Broadcasting, Early Access*, 12 pages (2018).
- [8] Tian, C., Sun, J., Wu, W. and Luo, Y.: Optimal Bandwidth Allocation for Hybrid Video-onDemand Streaming with a Distributed Max Flow Algorithm, *ACM Journal of Computer Networks*, Vol.91, No.C, pp.483-494 (2015).
- [9] Boronat, F., Montagud, M., Marfil, D. and Luzon, C.: Hybrid Broadcast/Broadband TV Services and Media Synchronization: Demands, Preferences and Expectations of Spanish Consumers, *IEEE Trans. Broadcasting*, Vol.64, No.1, pp.52-69 (2018).
- [10] Boronat, F., Marfil, D., Montagud, M. and Pastor, J.: HbbTV-Compliant Platform for Hybrid Media Delivery and Synchronization on Single-and Multi-Device Scenarios, *IEEE Trans. Broadcasting*, Vol.64, No.3, 6 pages (2017).
- [11] Christodoulou, L., Abdul-Hameed, O. and Kondo, A.-M.: Toward an LTE Hybrid Unicast Broadcast Content Delivery Framework, *IEEE Trans. Broadcasting*, Vol.63, No.4, pp.656-672 (2017).
- [12] Hang, Y., Gao, C., Guo, Y., Bian, K., Jin, X., Yang, Z., Song, L., Cheng, J., Tuo, H. and Li, X.M.: Proactive Video Push for Optimizing Bandwidth Consumption in Hybrid CDN-P2P VoD Systems, *Proc. IEEE INFOCOM*, 9 pages (2018).
- [13] Matsumoto, S., Ohira, K. and Yoshihisa, T.: A Mathematical Analysis of 2-Tiered Hybrid Broadcasting Environments, *Proc. International Workshop on Streaming Media Delivery and Management Systems (SMDMS)*, pp.454-460 (2019).
- [14] FFmpeg, available from (<https://ffmpeg.org>).
- [15] Big Buck Bunny, available from (<https://peach.blender.org/>).



松本 哲 (正会員)

1990年京都コンピュータ学院教員。2002年信州大学大学院工学系研究科システム工学専攻博士前期課程修了。2004年京都情報大学院大学助教。2007年京都大学産官学連携センター寄附研究部門助教。2010年神戸大学経済経営研究所助教。2015年大阪大学サイバーメディアセンター特任助教(常勤)。2016年同サイバーメディアセンター特任研究員(非常勤)。2016年大阪大学情報推進本部助教, サイバーメディアセンター兼務。現在に至る。電子情報通信学会, IEEE各会員。



義久 智樹 (正会員)

2002年大阪大学工学部電子情報エネルギー工学科卒業。2003年同大学院情報科学研究科マルチメディア工学専攻博士前期課程を修了し, 2005年同専攻博士後期課程修了。博士(情報科学)。2005年京都大学学術情報メディアセンター助教。2008年大阪大学サイバーメディアセンター講師を経て, 2009年より同准教授となり, 現在に至る。この間, カリフォルニア大学アーバイン校客員研究員。ストリームコンピューティングおよびインターネット放送に興味を持つ。IEEE会員。



下條 真司 (正会員)

1986年大阪大学基礎工学部大学院博士後期課程修了。1987年同大学助手。1989年同大型計算機センター講師。1991年同助教授。1998年同教授。2000年同大学サイバーメディアセンター副センター長, 2005年同センター長, 2007年同副センター長, 2008年より3年間情報通信研究機構大手町ネットワーク研究統括センター長・上席研究員。2011年大阪大学サイバーメディアセンター教授。2015年より同センター長。現在に至る。マルチメディアシステムアーキテクチャ, P2P・グリッド, 次世代インターネット等の研究開発に従事。2005年大阪科学賞。2006年総務大臣表彰。2011年本会論文賞受賞。博士(工学)。IEEE CS会員, 電子情報通信学会フェロー。