Vol.2017-DPS-171 No.19
Vol.2017-MBL-83 No.19
Vol.2017-ITS-69 No.19
2017/6/1

# クライアントの視聴情報に基づく分割放送型配信システムの実現

金本 俊平[1]　後藤 佑介[1]

**概要**：Internet Protocol (IP) ネットワークを用いて動画データを配信するサービスでは，サーバは，動画データの再生回数，評価値，およびコメント数といったユーザが動画データの視聴後にフィードバックする情報（以下，視聴情報）を取得することが重要である．筆者らの研究グループでは，動画データの放送型配信で発生する待ち時間を短縮する技術である分割放送型配信システムを設計，実装してきた．しかし，現在の分割放送型配信システムでは，サーバは動画データの配信中に配信スケジュールを更新できない．本研究では，クライアントの視聴情報に基づいて配信中に配信スケジュールを更新する分割放送型配信システムを実現し，本システムにおける動画データの視聴で発生するクライアントの待ち時間を評価する．

## 1. はじめに

近年，YouTube [1] や NHK オンデマンド [2] のように，Internet Protocol (IP) ネットワークを用いて動画データを配信するサービスが広まっている．IP ネットワーク上における動画データの配信方式として，オンデマンド型配信と放送型配信の二つがある．オンデマンド型配信では，サーバはクライアントの受信要求をもとに，クライアントごとに帯域幅を確保して動画データを配信する．このとき，クライアントからの受信要求数の増加にともない，サーバが使用する帯域幅は比例して増加する．一方，放送型配信では，サーバは一定の帯域幅ですべてのクライアントに動画データを配信するため，クライアント数が増加しても使用する帯域幅は増加しない．しかし，クライアントが受信を要求してから動画を再生するまでの間に待ち時間が発生する．

放送型配信で発生する待ち時間を短縮するため，一つの動画データを複数のデータ (以下，セグメント) に分割し，複数の通信路 (以下，チャネル) で配信する分割放送型配信が提案されてきた．また，分割放送型配信において，データ受信時に発生する待ち時間を短縮するスケジューリング手法が数多く提案されている．筆者らの研究グループでは，スケジューリング手法を導入可能なシステムとして，分割放送型配信システム ***TeleCaS*** [3] を設計，実装してきた．

現在の分割放送型配信システムでは，動画データの再生回数，評価値，および動画に対するコメント数といったユーザが動画データの視聴後にフィードバックする情報（以下，視聴情報）の取得が重要である．視聴情報は，動画データの配信開始から時間経過によって変動するため，放送型配信において視聴情報を反映できるシステムが必要となる．しかし，現在の ***TeleCaS*** では，クライアントの視聴情報をもとに，配信中に動画データの配信スケジュールを更新できない．

本研究では，クライアントの視聴情報に基づく分割放送型配信システムを実現する．実現するシステムでは，動画データの配信中に，クライアントからフィードバックされた視聴情報を反映した新たな配信スケジュールを読み込むことで，配信スケジュールを更新できる．

## 2. 動画データの配信方式

### 2.1 概要

動画データの配信方式として，オンデマンド型配信と放送型配信の二つがある．オンデマンド型配信では，サーバはクライアントの受信要求に応じて帯域幅を確保して動画を送信する．一方，放送型配信では，サーバは一定の帯域幅で複数のクライアントに動画データを繰り返し配信する．

### 2.2 オンデマンド型配信

オンデマンド型配信は，YouTube [1] や NHK オンデマンド [2] といった Video on Demand による動画配信サービスで用いられる配信方式である．オンデマンド型配信では，クライアントが動画データをサーバに要求すると，サーバは帯域幅を確保し，チャネルを用いてクライアントに動

[1] 岡山大学大学院自然科学研究科
Graduate School of Natural Science and Technology, Okayama University

ⓒ 2017 Information Processing Society of Japan

Vol.2017-DPS-171 No.19
Vol.2017-MBL-83 No.19
Vol.2017-ITS-69 No.19
2017/6/1

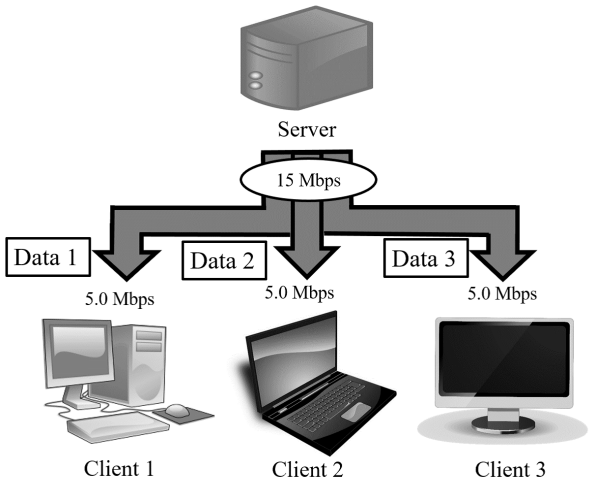



**図 1** オンデマンド型配信による動画データの配信例

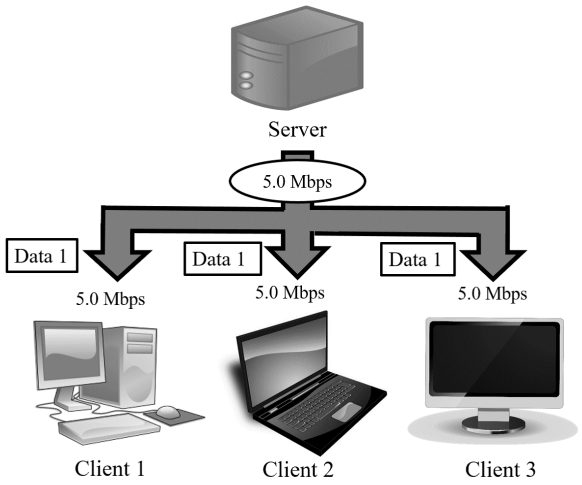



**図 2** 放送型配信による動画データの配信例

画データを送信する．サーバはクライアントの要求に応じて動画データを送信するため，クライアントは要求した動画データを即座に再生開始できる．一方で，サーバは，クライアントが動画データを要求するたびに帯域幅を確保する必要があるため，動画データを要求するクライアント数の増加に比例してサーバが使用する帯域幅は増加し，サーバの処理負荷は大きくなる．

オンデマンド型配信による動画データの配信例を**図 1** に示す．この例では，3 台のクライアントがサーバにそれぞれ別の動画データを要求している．サーバがクライアント 1 台あたり 5.0 Mbps の帯域幅を確保する場合，サーバが使用する帯域幅は 15 Mbps となり，クライアント数の増加に比例してサーバが使用する帯域幅は増加する．

### 2.3 放送型配信

放送型配信は，多くのクライアントが同じ動画データを受信する場合に用いられる配信方式である．即時性の高いスポーツ番組やコンサート映像のストリーミング配信，および特定の番組を一つのチャネルで繰り返し配信する Communication Satellite (CS) デジタル放送 [4] における利用を想定している．

放送型配信による動画データの配信例を**図 2** に示す．サーバは一定の帯域幅で同じ動画データを繰り返し配信し，クライアントは所望の動画データを受信して再生する．サーバは，マルチキャストやブロードキャストといったサーバとクライアントが 1 対多の場合の通信方式を用いて，クライアントに放送型で動画データを配信することで，オンデマンド型配信と比べてサーバの処理負荷や使用する帯域幅の増加を抑制できる．一方で，クライアントは動画データの受信を要求してから再生を開始するまでの間に待ち時間が発生する．

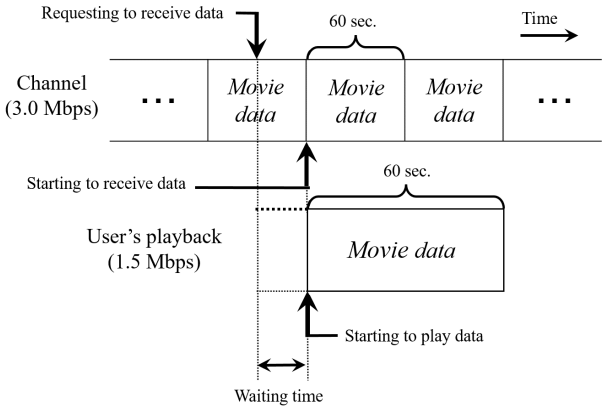



**図 3** 放送型配信における待ち時間発生の様子

放送型配信において待ち時間が発生する様子を**図 3** に示す．図 3 において，チャネルの帯域幅を 3.0 Mbps，動画データの再生レートを 1.5 Mbps，再生時間を 1 分とする．サーバは複数のクライアントに同じ動画データを繰り返し配信する．放送型配信における待ち時間が最長になる場合は，動画データの配信直後に受信要求を出した場合であり，データ受信の 1 周期分となる．一方で，最短になる場合は，動画データの配信直前に受信要求を出した場合であり，0 秒となる．以上より，待ち時間の平均は 0.5 周期分となる．図 3 では，待ち時間の最長は $(60 \times 1.5)/3.0 = 30$ 秒，最短は 0 秒となり，平均待ち時間は $(30 + 0)/2 = 15$ 秒となる．

### 2.4 配信方式の比較

オンデマンド型配信と放送型配信の長所および短所を表**表 1** に示す．オンデマンド型配信では，クライアントは動画データの受信と同時に動画データを再生できる．一方で，クライアント数が増加すると，サーバが使用する帯域

ⓒ 2017 Information Processing Society of Japan

Vol.2017-DPS-171 No.19
Vol.2017-MBL-83 No.19
Vol.2017-ITS-69 No.19
2017/6/1

表 1　オンデマンド型配信と放送型配信の比較

| | 長所 | 短所 |
|---|---|---|
| オンデマンド型配信 | 動画データを要求すると即座に再生可能 | サーバが使用する帯域幅が増加 |
| 放送型配信 | サーバが使用する帯域幅を抑制可能 | 動画データの要求から再生開始までの間に待ち時間が発生 |

幅は増加する．また，放送型配信では，クライアント数が増加してもサーバが使用する帯域幅は一定であるが，クライアントは動画データの要求から再生を開始するまでの間に待ち時間が発生する．

## 3. 分割放送型配信

### 3.1 分割放送型配信における待ち時間の短縮

分割放送型配信は，放送型配信で発生する待ち時間を短縮するため，動画データを複数のセグメントに分割し，最初のセグメントを頻繁に配信することで待ち時間を短縮する配信方式である．

例えば，図 3 で用いたデータを二つのセグメント $S_1, S_2$ に分割し，3.0 Mbps の 2 チャネルで $S_1, S_2$ をそれぞれ配信する場合，クライアントは受信した動画データ $S_1, S_2$ を動画を途切れなく再生できる．このとき，クライアントの待ち時間は，$S_1$ の先頭を受信するまでの時間であり，最長で $(60 \times 0. \times 1.5)/3.0 = 15$ 秒，最短で 0 秒となり，平均待ち時間は $(15 + 0)/2 = 7.5$ 秒となる．

### 3.2 スケジューリング手法

#### 3.2.1 概要

分割放送型配信では，最初のセグメントのデータサイズを小さくすることで，再生開始までの待ち時間を短縮できる．しかし，データサイズを小さくしすぎると，動画データの再生中にセグメント間で途切れが発生する．途切れが発生する場合，途切れ時間分だけ再生開始を遅らせて動画が途切れないようにするため，全体の待ち時間は長大化する．そこで，動画データの再生中に途切れが発生しないようにした上で待ち時間を短縮するため，動画データの配信条件に応じてセグメントの分割比率を決定するスケジューリング手法が提案されている．

#### 3.2.2 Fast Data Broadcasting and Receiving Scheme

既存のスケジューリング手法の一つである Fast Data Broadcasting and Receing Scheme （以下，FB 法）[5] について述べる．FB 法は帯域幅を $k$ 個のチャネルに分け，動画データを $2^k - 1$ 個のセグメントに等分割した上で，$i$ 番目のチャネル $C_i$ に連続した $2^{i-1}$ 個のセグメントを配置する手法である．

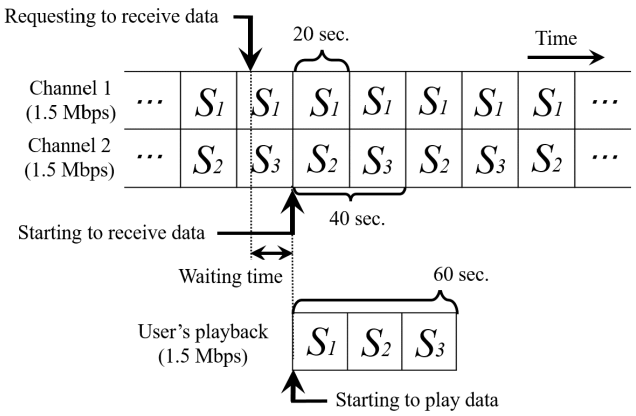



図 4　分割放送型配信におけるスケジューリング例（FB 法）

分割放送型配信における FB 法の配信スケジュールを図 4 に示す．サーバが使用できる帯域幅，動画データの再生レート，および再生時間は図 3 と同じとする．図 4 では，サーバの帯域幅を 1.5 Mbps の二つのチャネル $C_1, C_2$ に分割し，一続きの動画データを三つのセグメント $S_1, S_2, S_3$ に分割し，$C_1$ で $S_1$，$C_2$ で $S_2, S_3$ を繰り返し配信する．FB 法では，データを $2^k - 1$ 個のセグメントに等分割するため，$S_1, S_2, S_3$ の分割比率は等しく，再生時間はそれぞれ 20 秒となる．このとき，待ち時間は動画データの受信要求から $S_1$ の先頭を受信開始するまでの時間となり，最長で $(20 \times 1.5)/1.5 = 20$ 秒，最短で 0 秒となり，平均待ち時間は $(20 + 0)/2 = 10$ 秒となる．データを分割しない場合，およびデータを二つに等分割する場合の平均待ち時間はどちらも 15 秒であり，スケジューリング手法を導入することで，待ち時間を約 33.3 %短縮できる．

### 3.3 関連研究

動画データの分割配信において，データの分割比率の偏りにより途切れ時間が発生しないように，配信スケジュールを決定するスケジューリング手法は数多く提案されている．これらのスケジューリング手法では，チャネルの帯域幅や動画の再生レートといったさまざまな条件を考慮して適切な配信スケジュールを決定する．例えば，Harmonic Broadcasting（HB）法 [6] は，配信に必要とするチャネルの帯域幅の増加を抑制する手法である．Bandwidth Equivalent-Asynchronous Harmonic Broadcasting（BE-AHB）法 [7] は MPEG2 の Group of Pictures（GOP）や MP3 のフレームといった，再生単位と呼ばれるデータの再生に最低限必要なひとかたまりのデータの受信に着目している．クライアントが再生単位のデータの受信が完了するまで再生できない状況で，再生単位を考慮して配信スケジュールを作成することで，待ち時間を短縮する．さらに，複数の再生単位をまとめて配信することで，配信に必要なチャネル数を

ⓒ 2017 Information Processing Society of Japan

抑制できる．

また，放送中に配信スケジュールを更新するスケジューリング手法の例として Seam-less Channel Transitition（SCT）法 [8] が挙げられる．SCT 法は，放送開始時および配信スケジュール更新時に FB 法を用いて待ち時間を短縮する．Hierarchical Real-time Broadcasting（HRB）法 [9] は，放送中に使用できる帯域幅が追加された時点で，追加された帯域幅を最大限利用してチャネルを追加する．また，更新が要求された時点で追加分のチャネルの配信をすぐに開始することで，配信するセグメントのデータサイズを小さくする．さらに，配信スケジュールの変更で発生する処理負荷を抑制することで，待ち時間の長大化を抑える．

### 3.4 分割放送型配信システム

#### 3.4.1 *TeleCaS*

これまで提案されてきたスケジューリング手法の多くは，計算機を用いたシミュレーション環境を想定している．このため，システムの負荷やパケット欠損による動画再生への影響，および実際のネットワーク環境においてスケジューリング手法を用いた場合に発生する影響を考慮していない．この問題に対処するため，分割放送型配信システム ***Tele***communication and Broad***Ca***sting ***S***ystem (***TeleCaS***) が提案されている．***TeleCaS*** は，スケジューリング手法を導入可能であり，実際のネットワーク環境でスケジューリング手法を評価できる．しかし，現在の ***TeleCaS*** では，動画データの視聴回数，および視聴後にクライアントがフィードバックする評価やコメントといった視聴情報を反映できない．

#### 3.4.2 関連研究

選択型コンテンツの放送型配信システムとして，***Corne*** [10] が挙げられる．選択型コンテンツでは，コンテンツの視聴順序が複数に分岐しており，ユーザの嗜好に合わせてコンテンツを選択し，視聴することを想定している．選択型コンテンツの例として，クライアントの選択に応じて正解のコンテンツと不正解のコンテンツに分岐する択一式のクイズ番組が挙げられる．***Corne*** は，付加情報が配信スケジュールに影響を与える問題に対処したデータの配信契機を同期する方式，および逐次再生に対応する方式を実現することで，実際のネットワーク環境を考慮して選択型コンテンツを配信できる．また，***Corne*** は選択型コンテンツの放送型配信において多くのスケジューリング手法 [11], [12] を適用できる．

クライアント端末間でデータの送受信を行う端末間通信を用いたデータ配信システムとして，Content Delivery System for P2P Streaming（***DesPerS***）[13] が挙げられる．***DesPerS*** では，独自の通信プロトコルに基づいて，各端末は端末間で動画データを送受信する．また，データの受信を要求する端末（要求端末）がデータを要求端末に送信する端末（供給端末）を選択した上で，各供給端末が使用できる帯域幅を考慮したスケジューリング手法を用いて，端末間通信の性能評価を行うことができる．

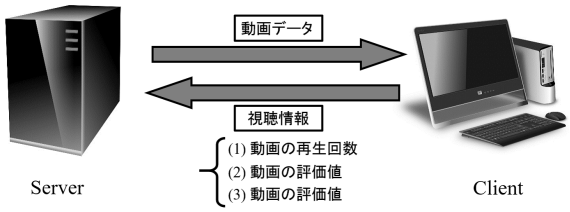



図 5　サーバ・クライアント間で送受信するデータ

## 4. クライアントの視聴情報

### 4.1 概要

視聴情報は，クライアント端末を用いて動画データを視聴するユーザが視聴中もしくは視聴後に配信元へフィードバックする情報を指す．例えば，ニコニコ動画 [14] において，ユーザが動画の視聴中にコメントを挿入したり，視聴後に動画を評価することで，動画の内容に対する感想や評価を動画作成者や配信元に報告できる．動画作成者は，視聴情報を参考にすることで，次に作成する動画データの品質向上に利用できる．また，配信元は，放送する番組の配信スケジュールを作成する判断基準として，視聴情報を利用できる．

サーバ・クライアント間で送受信するデータの例を図 5 に示す．動画データを配信する上で，クライアントの視聴情報を取得して配信スケジュールに反映することは重要である．本研究では，クライアントの視聴情報として，動画の再生回数，動画の評価値，および動画に対するコメントの 3 種類が挙げられる．以降の項で説明する．

### 4.2 想定する視聴情報

#### 4.2.1 動画の再生回数

動画配信サービスの配信元にとって，動画データの人気を判断する上で，再生回数は重要な情報である．再生回数が多い動画データについて，配信元は動画データの続編を配信したり，配信速度を向上させることができる．一方で，再生回数が少ない動画データについて，配信元はこの動画の配信を中止して別の動画を配信したり，配信速度を低下させることができる．

#### 4.2.2 動画の評価値

評価値は，ユーザが動画データの内容に対して視聴後に評価する値である．ユーザは，動画データの視聴後に評価値を入力することで，配信元にフィードバックされる．例えば，ニコニコ動画 [14] では，動画データの視聴後にアンケート形式で複数段階による評価が可能である．

配信元は，動画の評価値を取得することで，動画に対する視聴者の意見を数値化して把握できる．評価値が高い動

Vol.2017-DPS-171 No.19
Vol.2017-MBL-83 No.19
Vol.2017-ITS-69 No.19
2017/6/1

画データは，より多くのユーザが視聴できるように配信品質を向上させる．一方で，評価値が低い動画データは，配信停止や配信品質の低下といった対応を行う．

#### 4.2.3 動画に対するコメント

コメントは，ユーザが視聴中もしくは視聴後に動画データの内容に対して入力した文章であり，配信元は動画に対するユーザの感想を詳細に取得できる．動画作成者は，ユーザのコメントをもとに，次に作成する動画データの内容を決定できる．また，配信元は，ユーザのコメントをもとに，配信サービスの品質向上に利用できる．

## 5. 設計

### 5.1 想定環境

分割放送型配信システム ***TeleCaS*** において，動画データの配信中に視聴情報を受信し，配信スケジュールを更新するシステムを設計する．想定する環境を以下に示す．

(1) サーバからクライアントに配信する動画データの通信プロトコルは，UDP/IP を用いる．また，クライアントからサーバに送信する視聴情報の通信プロトコルは TCP/IP を用いる．
(2) スケジューリング手法の決定およびデータの分割は，配信開始前に行う．
(3) サーバは，複数のチャネルでセグメントを繰り返し配信する．
(4) サーバ，およびクライアントが動作する計算機は，C 言語のコンパイラを利用できる．

### 5.2 *TeleCaS* の課題

現在の ***TeleCaS*** は，サーバがクライアントに動画データを配信する設計であり，サーバは，動画データの配信中にクライアントから送信されるデータを受信できない．また，サーバは，動画データの配信中に，クライアントからの視聴情報をもとに配信スケジュールを更新できない．

***TeleCaS*** 上で動画データを配信する場合，はじめに，サーバは使用するスケジューリング手法を用いて，動画データを複数のセグメントに分割する．次に，分割放送型配信において必要な情報を含むデータ（以下，配信開始部）をクライアントに配信した後，動画データを繰り返し配信する．

現在の ***TeleCaS*** では，サーバがクライアントに動画データを配信する設計であり，クライアントがデータを受信する機能は実装していない．このため，サーバは，クライアントの視聴情報を受信できない．また，サーバは，動画データの配信中に，クライアントからの視聴情報をもとに配信スケジュールを更新できない．このとき，サーバがクライアントの視聴情報の受信中に動画データの配信を中断する場合，クライアントは動画データの再生中に途切れが発生する．したがって，サーバが動画データを配信しながらクライアントの視聴情報を受信するように設計する必要がある．さらに，動画データに対する評価，およびコメントといったクライアントの視聴情報の送信契機は，視聴後に設定する必要がある．

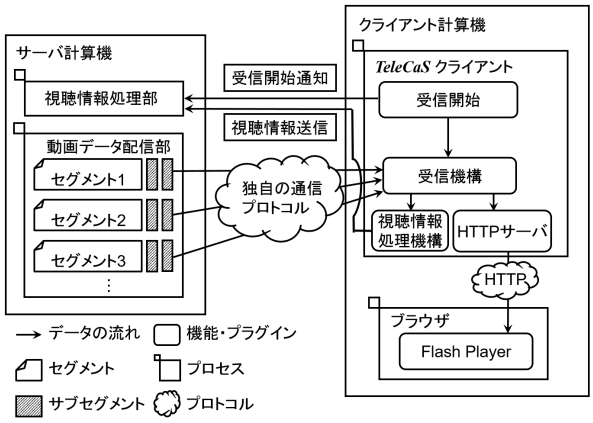



図 6 分割放送型配信システム ***TeleCaS*** の構成

また，現在の ***TeleCaS*** では，配信スケジュールの更新機能は実装しておらず，サーバが受信したクライアントの視聴情報に基づいて配信スケジュールを更新できない．配信スケジュールを更新する場合，動画データの配信を一時的に中断する必要がある．このとき，配信スケジュールの更新直前，もしくは更新中に接続したクライアントは，再生中に途切れが発生する可能性がある．

### 5.3 対処

#### 5.3.1 サーバ・クライアント間における視聴情報の通信方法

サーバがクライアントから視聴情報を受信するため，サーバ・クライアント間で視聴情報の通信機能を実現する．

本研究で実現する分割放送型配信システム ***TeleCaS*** の構成を図 **6** に示す．サーバでは，クライアントが送信する視聴情報の受信処理，および動画データの配信処理を並行して行うため，クライアントから視聴情報を受信するためのスレッド（以下，視聴情報処理部）を作成する．視聴情報処理部では，クライアントが送信した視聴情報を受信し，受信した視聴情報をもとに，配信スケジュールを更新する．一方，クライアントでは，動画データ受信後に，視聴情報処理機構がクライアントの視聴情報を視聴情報処理部に送信する．

#### 5.3.2 視聴情報に基づく配信システムの更新手順

***TeleCaS*** において，クライアントから受信した視聴情報に基づいた配信スケジュールの更新機能を実現する．サーバは，クライアントの視聴情報に基づいて，チャネル数およびチャネルの帯域幅を更新して配信スケジュールを作成し，現在利用している更新前の配信スケジュールと切り替えることで，配信スケジュールの更新を実現する．ク

ⓒ 2017 Information Processing Society of Japan

Vol.2017-DPS-171 No.19
Vol.2017-MBL-83 No.19
Vol.2017-ITS-69 No.19
2017/6/1

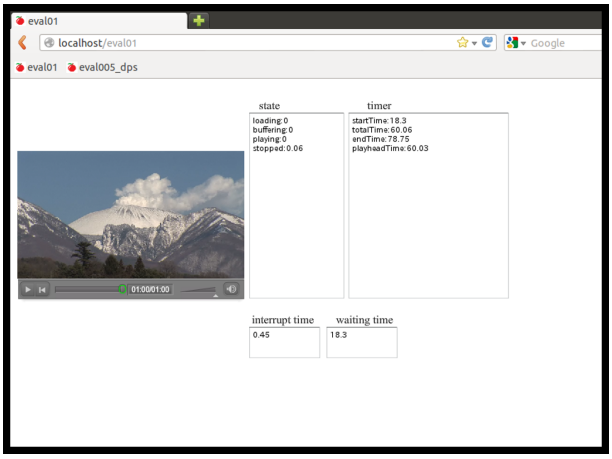



図 7　*TeleCaS* のスクリーンショット

ライアントが更新前の配信スケジュールで動画データを受信している場合，ブラウザを更新し，更新後の配信スケジュールを用いて動画データの受信を継続する．

*TeleCaS* におけるサーバの動作，および配信スケジュールの更新手順を以下に示す．

(1) 配信する動画データを構成するセグメントの情報を配信スケジュールから取得する．
(2) クライアントの視聴情報を受信するためのスレッドを作成する．
(3) 動画データの配信を開始する．
(4) 視聴情報受信部において，クライアントが送信する視聴情報の受信を待機する．
(5) 受信したクライアントの視聴情報に基づいて，配信スケジュールを更新するかを決定する．配信スケジュールを更新する場合，更新後の配信スケジュールを用いて動画データを配信する．一方，更新しない場合，現在利用している配信スケジュールで動画データを引き続き配信する．

### 5.4　実装

5.3 節で述べた実現方式をもとに，クライアントの視聴情報に基づく通信機能，および配信スケジュールの更新処理を *TeleCaS* に実装した．*TeleCaS* では，チャネル数，チャネルの帯域幅，セグメントのデータサイズ，およびセグメントの配信順序を設定することで，動画データを配信できる．また，*TeleCaS* におけるサーバおよびクライアントのプログラムは，多くの計算機で動作することを考慮して，C 言語で開発した．

*TeleCaS* における再生画面のスクリーンショットを**図 7** に示す．図 7 は，サーバが配信した動画データをブラウザで再生している様子を示す．動画データの再生では，Adobe Flash Player を用いた．

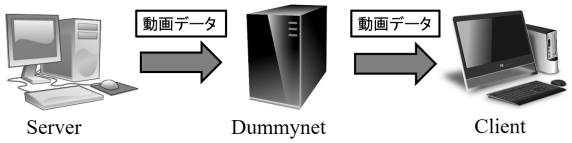



図 8　評価環境の構成

表 2　計算機の性能

| | | |
|---|---|---|
| Server | CPU | Intel®Core™2 Duo E7500 (2.93 GHz) |
| | Memory | 2.0 GBytes |
| | OS | Ubuntu 12.10 |
| | NIC | RTL8101E/RTL8102E |
| Dummynet | CPU | Intel®Core™2 Duo E7500 (2.93 GHz) |
| | Memory | 2.0 GBytes |
| | OS | FreeBSD 8.2-RELEASE |
| | NIC1 | RTL8169SC |
| | NIC2 | RTL8169SC |
| Client | CPU | Intel®Core™2 Duo E7500 (2.93 GHz) |
| | Memory | 2.0 GBytes |
| | OS | Ubuntu 12.10 |
| | NIC | RTL8101E/RTL8102E |

## 6.　評価

### 6.1　評価環境

*TeleCaS* における評価環境を**図 8** に示す．サーバ計算機とクライアント計算機は，Gigabit Ethernet で接続した．サーバ計算機とクライアント計算機の間に帯域制御機能 Dummynet [15] を利用できる計算機を挿入し，ネットワークの帯域幅を制御する．評価に用いた各計算機の性能を**表 2** に示す．

### 6.2　評価項目

配信スケジュールの更新によってクライアントで発生する影響を確認するため，3 種類の評価項目で待ち時間を比較評価する．一つ目は，動画データの受信中に配信スケジュールの更新が発生しない場合（以下，更新無し）である．二つ目は，動画データの受信開始と配信スケジュールの更新が同時となる合（以下，受信開始時更新）である．三つ目は，動画データの受信中に配信スケジュールを更新する場合（以下，受信中更新）である．

クライアントが動画データを再生している間，配信スケジュールを用いた処理の影響で，途切れが発生する可能性がある．本研究では，受信要求から第 1 セグメントの受信開始までのロード時間，および再生中に発生する動画データの途切れ時間の合計を待ち時間とする．また，評価に使用するスケジューリング手法は，FB 法とする．FB 法は，他のスケジューリング手法に比べてセグメントの分割処理が単純であり，評価に用いるスケジューリング手法として適している．

ⓒ 2017 Information Processing Society of Japan

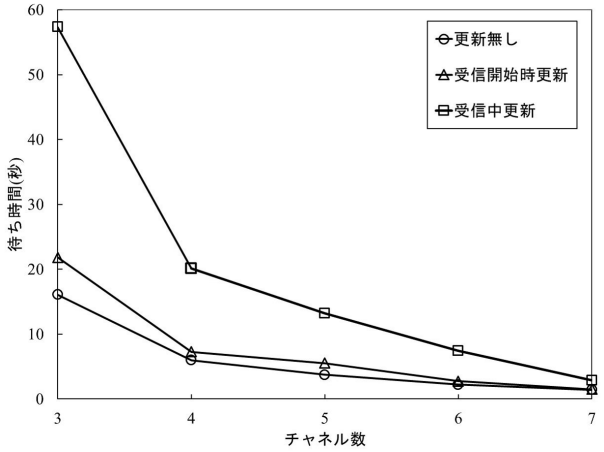



図 9　チャネル数と待ち時間

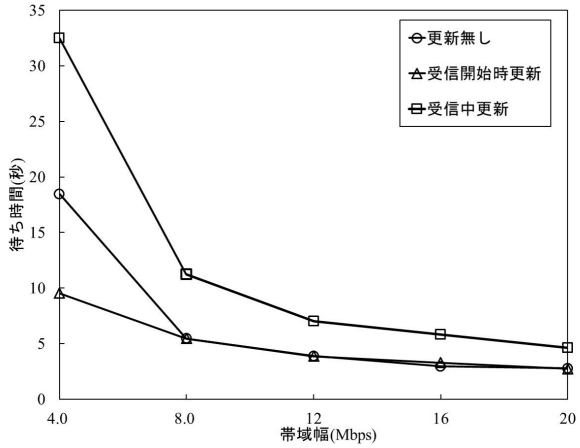



図 10　帯域幅と待ち時間

### 6.3　チャネル数

本節では，使用するチャネル数に応じて，配信スケジュールの更新が待ち時間に与える影響を評価する．チャネル数に応じた待ち時間の変化を図 **9** に示す．横軸はチャネル数である．縦軸は待ち時間であり，クライアント計算機で 5 回測定した平均値とした．チャネルの帯域幅および動画データの再生レートを 1.5 Mbps とする．動画のデータサイズは再生時間が 60 秒となる 11.3 Mbps とする．今回の評価において，配信スケジュールの更新では，サーバはチャネル数が 1 増えた場合の FB 法の配信スケジュールでクライアントにセグメントを引き続き配信する．

図 9 より，受信開始時更新，および受信中更新の 2 種類の場合における待ち時間は，更新無しの場合に比べてそれぞれ長大化する．更新無しの場合，クライアントがサーバに受信要求するときにサーバは動画データを配信しており，配信スケジュールに応じた待ち時間が発生する．一方で，他の 2 種類の場合，クライアントがサーバに受信を要求するとき，サーバは配信スケジュールの更新処理を行っている．このとき，クライアントはサーバから動画データを受信できず，待ち時間は長大化する．特に，受信中更新の場合，更新前の配信スケジュールで動画データを受信中であることが原因で，配信スケジュールの更新後にブラウザ上で再度読み込みを行う必要がある．

また，チャネル数が増加すると，3 種類の評価で発生する待ち時間の差は小さくなる．例えば，チャネル数が 3，および 7 の場合について，更新無し，および受信開始時更新の 2 種類の評価における待ち時間の差は，約 5.73 秒から約 0.09 秒に短縮する．また，更新無し，および受信中更新の 2 種類の評価における待ち時間の差は，約 41.28 秒から約 1.45 秒に短縮される．これは，チャネル数の増加によって配信スケジュールが更新され，動画データの配信周期が短くなるためである．FB 法では，チャネル数が 1 増えると，セグメント数は $(2^i - 1) - (2^{i-1} - 1) = 2^i - 2i - 1$ 個増える．この時，各チャネルの配信周期は $2^{i-1}/2^i$ 倍となり，$S_i$ の配信周期が短くなるため，待ち時間は短くなる．このとき，配信スケジュールの更新による処理時間が発生するが，チャネル数が 7 の場合では，更新処理による影響はほとんどない．

### 6.4　帯域幅

配信スケジュールの更新において，帯域幅に応じた待ち時間の変化を図 **10** に示す．横軸はサーバが使用する帯域幅である．縦軸は待ち時間であり，クライアント計算機で 5 回測定した平均値である．動画のデータサイズは 11.3 Mbps，チャネル数は 4 とする．その他の配信条件は，6.3 節と同じである．

図 10 より，帯域幅が 8.0 Mbps 以上のとき，更新なし，および受信開始時更新の場合における待ち時間は，ほぼ同じとなる．また，受信中更新の場合について，帯域幅が 12 Mbps 以上のとき，待ち時間は他の 2 種類の場合に比べて，大きく長大化しない．以上より，チャネルの帯域幅が十分に確保されている場合，配信スケジュールの更新による待ち時間の長大化に対する影響は小さいことが分かる．

一方，帯域幅が 4.0 Mbps のとき，受信開始時更新の場合における待ち時間は，他の 2 種類に比べて短い．各チャネルの帯域幅が動画データの再生レートより低い場合，再生中の途切れ時間は長大化する．受信開始時更新の場合，サーバが配信スケジュールを更新してから動画データの配信を開始するまでの間，クライアントはサーバから配信される動画データを待つ必要がある．このとき，クライアントは最初のセグメントから受信できるため，途切れ時間は短くなる．

### 6.5　動画のデータサイズ

配信スケジュールの更新において，動画のデータサイズに応じた待ち時間の変化を図 **11** に示す．横軸は動画のデータサイズであり，11.3 Mbytes，17.0 Mbytes，22.5 Mbytes，28.1 Mbytes，および 33.8 Mbytes の 5 項目はそ

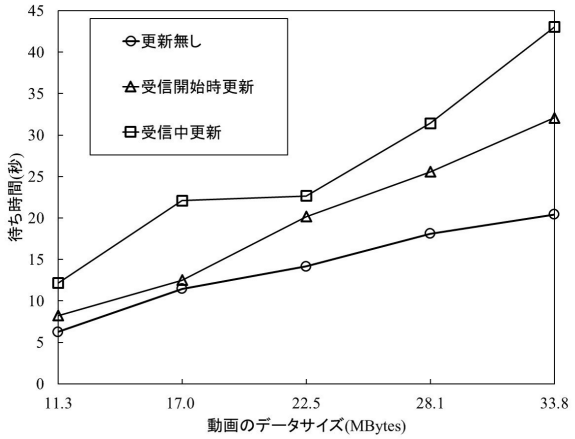



**図 11** 動画のデータサイズと待ち時間

れぞれデータの再生時間が 60 秒，90 秒，120 秒，150 秒，および 180 秒に対応する．縦軸は待ち時間であり，クライアント計算機で 5 回測定した平均値とした．チャネル数は 4 とする．動画データのデータサイズ以外の配信条件は 6.3 節と同じとする．

図 11 より，動画のデータサイズが大きくなると，セグメントのデータサイズは大きくなり，各セグメントの配信時間は長大化する．このとき，各セグメントのデータサイズが大きくなると，配信周期が長くなるため，配信スケジュールの更新にかかる時間は長大化し，各項目における待ち時間の差は大きくなる．3.2.2 節で説明したように，FB 法では，チャネル $C_1$ で $S_1$，$C_2$ で $S_2$ と $S_3$ を繰り返し配信する．クライアントが配信スケジュール更新後にブラウザを再度読み込んで動画データの受信を再開すると，同じチャネルにおいて，次に再生するセグメントより先のセグメントの受信が先に完了する可能性がある．このような場合，次に再生するセグメントの受信が完了するまでの間で途切れが発生し，受信開始時更新の場合と比べて全体の待ち時間は長大化する．

## 7. おわりに

本研究では，分割放送型配信システム ***TeleCaS*** において，クライアントの視聴情報に基づいて配信中に動画データの配信スケジュールを更新するシステムを実現した．実現したシステムでは，サーバは動画データの配信中に，クライアントからフィードバックされた視聴情報を反映した新たな配信スケジュールに更新できる．配信スケジュールの更新における評価では，帯域幅，チャネル数，および動画のデータサイズの 3 項目について，待ち時間をそれぞれ評価した．評価結果より，帯域幅，およびチャネル数が多いほど，配信スケジュールの更新において待ち時間に与える影響は小さくなる．また，動画のデータサイズが小さいほど，待ち時間に与える影響は小さくなる．

今後の予定として，***TeleCaS*** における複数動画への対応，および他のスケジューリング手法の導入がある．

**謝辞**

本研究の一部は，文部科学省科学研究費補助金・基盤研究（B）（15H02702），基盤研究（C）（16K01065），および（公財）中島記念国際交流財団の研究助成によるものである．ここに記して謝意を表す．

**参考文献**

[1] YouTube: YouTube（オンライン），入手先 ⟨https://www.youtube.com/⟩ (参照 2017-02-28).
[2] NHK: NHK オンデマンド，NHK（オンライン），入手先 ⟨https://www.nhk-ondemand.jp⟩ (参照 2017-02-28).
[3] 木村明寛，後藤佑介，谷口秀夫: 動画データを分割配信するシステムの実現と評価，電子情報通信学会論文誌 B，Vol.J96-B，No.10，pp.1217-1225 (2013).
[4] （一社）衛星放送協会：衛星放送協会のしくみ，衛星放送協会（オンライン），入手先 ⟨http://www.eiseihoso.org/guide/howto.html/⟩ (参照 2017-02-28).
[5] Juhn, L. and Tseng, L.: Fast Data Broadcasting and Receiving Scheme for Popular Video Service, IEEE Trans. on Broadcasting, Vol.44, No.1, pp.100-105 (1998).
[6] Juhn, L. and Tseng, L.: Harmonic broadcasting for video-on-demand service, IEEE Trans. Broadcasting, Vol.43, No.3, pp.268-271 (1997).
[7] 義久智樹，塚本昌彦，西尾章治郎: 再生単位を考慮したスケジューリング手法における使用チャネル数について，日本データベース学会 Letters，Vol.4，No.3，pp.5-8 (2005).
[8] Tseng, Y.-C., Yang, M.-H., Hsieh, C.-M., Liao, W.-H. and Sheu, J.-P.: Data Broadcasting and Seamless Channel Transition for Highly Demanded Videos, IEEE Trans. Communications, Vol.49, No.5, pp.863-874 (2001).
[9] Gotoh, Y and Taniguchi, H: A Scheduling Method for Division Based Broadcasting Cnsidering Dynamic Update, 2nd Workshop on Advances in Data Engineering and Mobile Computing (DEMoC-2013), pp.332-337 (2013).
[10] 後藤佑介，山本泰平，谷口秀夫: 選択型コンテンツの放送型配信システムの実現，情報処理学会研究報告，Vol.2014-DPS-161，No.6，pp.1-8 (2014).
[11] 義久智樹，金澤正憲: 選択型コンテンツの放送型配信におけるスケジューリング手法，情報処理学会論文誌，Vol.47，No.12，pp.3296-3307 (2006).
[12] Gotoh, Y., Yoshihisa, T., Kanazawa, M. and Takahashi, Y.: A Broadcasting Scheme for Selective Contents Considering Available Bandwidth, IEEE Trans. Broadcasting, Vol.55, Issue 2, pp.460-467 (2009).
[13] Gotoh, Y., Suzuki, K., Yoshihisa, T. and Kanazawa, M.: Design and Implementation of P2P Streaming Systems for Webcast, Proc. First International Conference on Personal Satellite Services (PSATS 2009), Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Vol.15, pp.103-110 (2009).
[14] ニコニコ動画: ニコニコ動画（オンライン），入手先 ⟨http://www.nicovideo.jp/⟩ (参照 2017-04-13).
[15] Dummynet: The dummynet project（online），入手先 ⟨http://info.iet.unipi.it/∼luigi/dummynet/⟩ (参照 2017-02-28).

ⓒ 2017 Information Processing Society of Japan