

複数動画を同期配信する分割放送型配信システムの評価

井上 祐輔¹ 木村 明寛¹ 後藤 佑介¹

概要: 音声や映像といった一続きの動画を多くのクライアントが視聴する場合、放送型によるデータ配信が有効な手段である。特に、動画データを分割して複数の通信路を用いて繰返し放送することでデータ受信時の待ち時間を短縮する分割放送型配信では、分割したデータを効率的に放送するためのスケジューリング手法が数多く提案されている。我々の研究グループでは、実際のネットワーク環境でこれらのスケジューリング手法を導入可能な分割放送型配信システムを提案し、スケジューリング手法の性能評価を行っている。しかし、本システムでは、サーバが周期ごとに複数の動画をまとめて配信し、クライアントが同時に視聴する環境に対応していなかった。本研究では、複数動画を同期配信する分割放送型配信システムを実現し、スケジューリング手法を用いた待ち時間の短縮効果を評価する。評価では、実際のネットワーク環境でサーバが複数の動画を同期配信する場合、理想的な環境で配信する場合に比べて長大な途切れ時間が発生することを示す。

1. はじめに

インターネットによる映像視聴方法の多様化にともない、複数動画を同時に視聴する環境への注目が高まっている。動画データの配信方式として、Video on Demand (以下、VoD) と放送型配信の二つがある。VoD 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配信する。このため、同じ時間帯に動画を要求するクライアント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複数のクライアントに同じ動画データを繰返し配信するが、クライアントは要求した動画データが配信されるまで待つ必要がある。この待ち時間を短縮するため、一続きの動画データを複数の部分に分割して複数の通信路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おり、我々はスケジューリング手法を導入可能な分割放送型配信システムを実現してきた。

これまでの研究で、複数動画を同時に視聴する場合を考慮したスケジューリング手法は提案されているが、実際のネットワーク環境を考慮したシステム上での評価は行っていない。本研究では、複数動画を同期配信する分割放送型配信システムを実現し、スケジューリング手法を用い

た評価を行う。

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、VoD と放送型配信の二つが挙げられる。以下で、順番に説明する。

2.2 VoD

VoD は、NHK オンデマンド [1] をはじめとするインターネットテレビや YouTube [2] といった多数の動画を配信する動画共有サービスで用いられる配信方式である。クライアントがサーバに動画データの受信を要求すると、サーバは帯域幅を確保し、通信路 (以下、チャンネル) を用いて動画データを配信する。サーバは、クライアントの受信要求に応じて動画データを配信するため、クライアントは要求した動画を即座に受信して再生を開始できる。しかし、クライアントが動画データの受信を要求するたびにサーバは帯域幅を確保する必要があるため、動画を要求するクライアント数が増加すると、サーバが使用する帯域幅は比例して増加し、サーバの処理負荷は大きくなる。

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ンネルで繰返し配信する CS デジタル放送 [3] といったサービスでの利用を想定している。

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

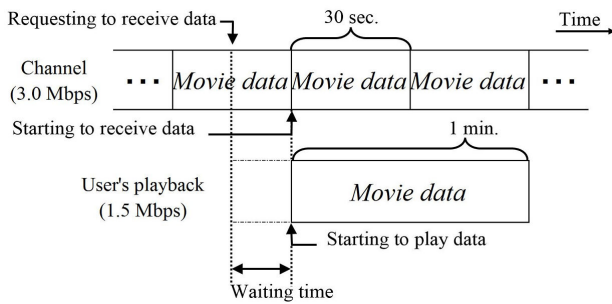


図 1 放送型配信における待ち時間発生の様子

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契機で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通信方式を用いてクライアントに放送型で配信することで、VoD に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が開始されるまでの間に待ち時間が発生する。

放送型配信において待ち時間が発生する様子を図 1 に示す。チャンネルの帯域幅を 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。図 1 の場合、クライアントが動画データの受信を開始する契機は動画データの先頭であり、クライアントは動画データの先頭部分を受信すると再生を開始できる。クライアントが動画データを受信するときの待ち時間は、サーバが動画データの先頭部分を配信した直後にクライアントが受信を要求した場合に最長となり、動画データ一周期分の $(60 \times 1.5) / 3.0 = 30$ 秒かかる。一方、サーバが動画データの先頭部分を配信する直前にクライアントが受信を要求した場合に待ち時間は最短となり、0 秒となる。以上より、図 1 の平均待ち時間は $(30 + 0) / 2 = 15$ 秒となる。

3. 複数動画の分割放送型配信

3.1 分割放送型配信

分割放送型配信は、動画データを複数の部分（以下、セグメント）に分割して、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 1 で用いた動画データを二つのセグメント S_1, S_2 に等分割し、1.5 Mbps の二つのチャンネル C_1, C_2 を用いて C_1 で S_1, C_2 で S_2 を繰り返し配信する場合、クライアントは受信した S_1, S_2 を途切れなく順番に再生できる。このとき、クライアントの待ち時間は S_1 の先頭が配信されるまでの時間であり、最長で $(60 \times 0.5 \times 1.5) / 1.5 = 30$ 秒、最短で 0 秒となり、平均の待ち時間は $(30 + 0) / 2 = 15$ 秒となる。

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より短縮できるが、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。この場合、再生中に途切れが発生しないようにす

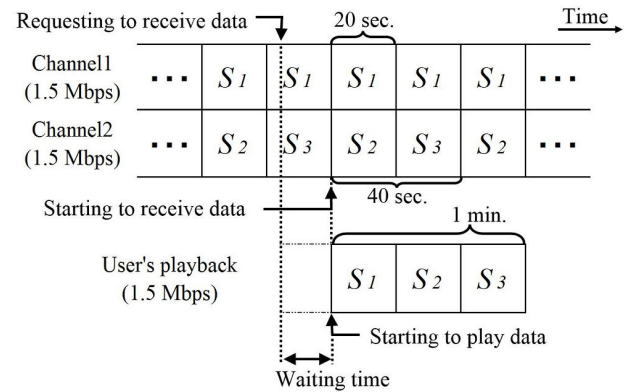


図 2 分割放送型配信における配信スケジュール例 (FB 法)

るため、途切れ時間分だけ再生開始を遅らせる必要があり、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受信時の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 [4] - [11] が提案されてきた。

図 2 に、既存のスケジューリング手法である Fast Data Broadcasting and Receiving Scheme (以下、FB 法) [5] による配信スケジュール例を示す。FB 法は、サーバが使用できる帯域幅を k 個のチャンネルに分け、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 番目のチャンネル 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。チャンネルの帯域幅、動画データの再生レート、および再生時間は図 1 と同じとする。

図 2 の場合、サーバは使用できる帯域幅を 1.5 Mbps の二つのチャンネル C_1, C_2 に分割した上で、一続きの動画データを三つのセグメント S_1, S_2, S_3 に分割して、 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、動画データを $2^k - 1$ 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、セグメントの再生時間は 20 秒となる。このとき、待ち時間は S_1 の先頭が配信され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用いて分割することで、待ち時間を約 33.3% 短縮できる。以上より、分割放送型配信において、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の導入は重要である。

3.2 複数動画の同期配信の必要性

現在、サーバが複数動画をまとめて配信してクライアントが同時に視聴する環境は一般的になっている。例えば、fv ソリューション [12] を利用した通信カラオケのサービス [13] が挙げられる。このサービスを行うカラオケ店舗では、ユーザの要望に応じて部屋の映像を撮影し、テレビ

画面に表示する。さらに、複数のカラオケ部屋でそれぞれ撮影された複数の映像を各部屋のテレビ画面上で同時に表示することで、ユーザはまるで一つのカラオケ部屋で他の部屋のユーザとカラオケを楽しんでいるような経験ができる。

サーバが周期ごとに複数動画を放送型でまとめて配信する場合、一度に配信するデータサイズは大きくなり、再生開始までの待ち時間は長大化する。このため、待ち時間を短縮するスケジューリング手法の利用は、複数動画の同期配信において重要である。

3.3 複数動画の同期配信を考慮したスケジューリング手法

これまでの分割放送型配信では、サーバはスケジューリング手法を用いて配信することで、クライアントの動画再生までの待ち時間を効率的に短縮していた。スケジューリング手法では、サーバが使用できる帯域幅、動画の再生単位、およびクライアントの受信能力といった配信条件をもとに、各セグメントをどのタイムスロットでどのチャンネルから配信するかを決定する。一方で、分割放送型で複数動画を同期配信する場合、一度に配信するすべての動画データについて待ち時間を短縮する必要があり、スケジューリングは複雑になる。以下で、複数動画の同期配信を考慮した既存のスケジューリング手法である Basic multiple-video broadcasting scheme (以下、MV-B 法) [8]、および Multiple-video broadcasting scheme with repairing (以下、MV-R 法) [8] の二つを説明する。

3.4 複数動画の同期配信における想定環境

複数動画を同期配信するためのスケジューリング手法である MV-B 法、および MV-R 法の想定環境を以下に示す。

- (1) 各チャンネルの帯域幅は、再生レートと等しい。
- (2) 各チャンネルは、複数の動画データのセグメントを組み合わせてスケジューリングできる。
- (3) クライアントは、すべてのチャンネルから同時に受信できる。

3.5 MV-B 法

MV-B 法は、複数動画の分割放送型配信において待ち時間を短縮するスケジューリング手法である。動画数が m のとき、 m 番目の動画の第 n セグメントは $S_{(m,n)}$ と表す。また、 j 番目のタイムスロット T_j までに i 番目の動画の第 n セグメント $S_{(i,j)}$ を放送しなければならない条件があり、この条件をスケジューリング条件と呼ぶ。例えば、セグメント $S_{(i,j)}$ は T_j までに最低一つのチャンネルにスケジューリングする必要がある。このとき、スケジューリング手順は以下ようになる。

- (1) サーバは、各動画で j 番目のセグメントである $S_{(1,j)}$, \dots , $S_{(m,j)}$ を $f(j) = \lceil m/j \rceil$ 番目以降のチャンネルに順

Time slot	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}	...
Channel 1	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	
Channel 2	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	
Channel 3	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	
Channel 4	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	
Channel 5	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	
Channel 6	S(1,2)	S(2,2)	S(1,2)	S(2,2)	S(1,2)	S(2,2)	S(1,2)	S(2,2)	S(1,2)	S(2,2)	
Channel 7	S(3,2)	S(4,2)	S(3,2)	S(4,2)	S(3,2)	S(4,2)	S(3,2)	S(4,2)	S(3,2)	S(4,2)	
Channel 8	S(5,2)	idle	S(5,2)	idle	S(5,2)	idle	S(5,2)	idle	S(5,2)	idle	
Channel 9	S(1,3)	S(2,3)	S(3,3)	S(1,3)	S(2,3)	S(3,3)	S(1,3)	S(2,3)	S(3,3)	S(1,3)	
Channel 10	S(4,3)	S(5,3)	idle	S(4,3)	S(5,3)	idle	S(4,3)	S(5,3)	idle	S(4,3)	
Channel 11	S(1,4)	S(2,4)	S(3,4)	S(4,4)	S(1,4)	S(2,4)	S(3,4)	S(4,4)	S(1,4)	S(2,4)	
Channel 12	S(5,4)	idle	idle	idle	S(5,4)	idle	idle	idle	S(5,4)	idle	
Channel 13	S(1,5)	S(2,5)	S(3,5)	S(4,5)	S(5,5)	S(1,5)	S(2,5)	S(3,5)	S(4,5)	S(5,5)	
Channel 14	S(1,6)	S(2,6)	S(3,6)	S(4,6)	S(5,6)	idle	S(1,6)	S(2,6)	S(3,6)	S(4,6)	
Channel 15	S(1,7)	S(2,7)	S(3,7)	S(4,7)	S(5,7)	idle	idle	S(1,7)	S(2,7)	S(3,7)	

図 3 分割放送型配信における複数動画の配信スケジュール例 (MV-B 法)

番にスケジューリングする。

- (2) サーバは、チャンネル $h_{(i,j)}$ でセグメント $S_{(i,j)}$ を放送する。 $h_{(i,j)}$ は、以下の式で求められる。

$$h_{(i,j)} = \sum_{k=1}^{j-1} f(k) + \lceil i/j \rceil, 1 \leq i \leq m, j \geq 1 \quad (1)$$

上記のスケジューリング手順にもとづき、MV-B 法でスケジューリングすると、セグメントがチャンネルにスケジューリングされない時間帯 (以下、アイドル時間) が発生する。動画数を 5、チャンネル数を 15 とした場合の配信スケジュール例を図 3 に示す。図 3 の場合、*idle* と表示されたタイムスロットがアイドル時間であり、チャンネル 8, 10, 12, 14, および 15 でアイドル時間が発生する。

3.6 MV-R 法

MV-R 法は、3.5 節で述べた MV-B 法で発生するアイドル時間に他のセグメントを配信することで、MV-B 法より待ち時間を短縮する手法である。MV-R 法では、MV-B 法と同様にチャンネル 1 から順番にセグメントをスケジューリングするが、あるタイムスロットでアイドル時間が発生した場合、他のセグメントをそのタイムスロットにスケジューリングする。

チャンネルのアイドル時間に他のセグメントを割り当てて補完する方法として、完全補完と強制補完の二つがある。完全補完は、アイドル時間にセグメントを割り当てるとき、割り当てたセグメントがスケジューリング条件を維持できる場合の補完方法である。一方、強制補完は、アイドル時間にセグメントを割り当てるとき、割り当てたセグメントがスケジューリング条件を維持できない場合の補完方法である。すべてのアイドル時間を補完した配信スケジュールを図 4 に示す。MV-R 法では、MV-B 法でセグメントをスケジューリングしながら、アイドル時間を他のセグメントで補完することで、MV-B 法に比べて待ち時間を短縮できる。

Time slot	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}	...
Channel 1	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	S(1,1)	...
Channel 2	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	S(2,1)	...
Channel 3	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	S(3,1)	...
Channel 4	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	S(4,1)	...
Channel 5	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	S(5,1)	...
Channel 6	S(1,2)	S(2,2)	S(1,2)	S(2,2)	S(1,2)	S(2,2)	S(1,2)	S(2,2)	S(1,2)	S(2,2)	...
Channel 7	S(3,2)	S(4,2)	S(3,2)	S(4,2)	S(3,2)	S(4,2)	S(3,2)	S(4,2)	S(3,2)	S(4,2)	...
Channel 8	S(5,2)	S(5,4)	S(5,2)	S(5,8)	S(5,2)	S(5,4)	S(5,2)	S(5,8)	S(5,2)	S(5,4)	...
Channel 9	S(1,3)	S(2,3)	S(3,3)	S(1,3)	S(2,3)	S(3,3)	S(1,3)	S(2,3)	S(3,3)	S(1,3)	...
Channel 10	S(4,3)	S(5,3)	S(5,6)	S(4,3)	S(5,3)	S(4,6)	S(4,3)	S(5,3)	S(5,6)	S(4,3)	...
Channel 11	S(1,4)	S(2,4)	S(3,4)	S(4,4)	S(1,4)	S(2,4)	S(3,4)	S(4,4)	S(1,4)	S(2,4)	...
Channel 12	S(1,5)	S(2,5)	S(3,5)	S(4,5)	S(5,5)	S(1,5)	S(2,5)	S(3,5)	S(4,5)	S(5,5)	...
Channel 13	S(1,6)	S(2,6)	S(3,6)	S(1,10)	S(2,10)	S(3,10)	S(1,6)	S(2,6)	S(3,6)	S(1,10)	...
Channel 14	S(1,7)	S(2,7)	S(3,7)	S(4,7)	S(5,7)	S(4,10)	S(5,10)	S(1,7)	S(2,7)	S(3,7)	...
Channel 15	S(1,8)	S(2,8)	S(3,8)	S(1,9)	S(2,9)	S(3,9)	S(4,9)	S(5,9)	S(1,8)	S(2,8)	...

図 4 分割放送型配信における複数動画の配信スケジュール例 (MV-R 法)

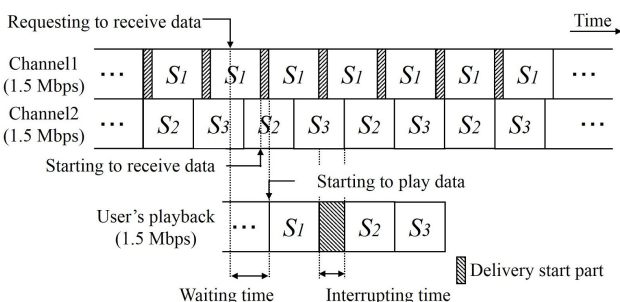


図 5 TeleCaS における非同期配信の例 (FB 法)

4. 分割放送型配信システム TeleCaS

4.1 概要

これまでに提案されてきたスケジューリング手法の多くは、計算機上でのシミュレーション環境を想定している。このため、システムの処理負荷やパケット損失による動画再生への影響を想定しておらず、実際のネットワーク環境でスケジューリング手法を用いる場合に発生する影響を考慮していない。我々の研究グループでは、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可能な分割放送型配信システム *Telecommunication and BroadCasting System (TeleCaS)* [14] を提案してきた。しかし、*TeleCaS* は、複数動画の同期配信や同時再生に対応しておらず、MV-B 法や MV-R 法を適用できない。そこで、*TeleCaS* において、複数動画の同期配信および同時再生を実現する。課題として、セグメントの配信契機を考慮した同期配信方式の実現、および複数動画の同時再生方式の実現という二つの課題がある。以降の節で、順番に説明する。

4.2 セグメントの配信契機を考慮した同期配信方式

分割放送型配信で提案されているスケジューリング手法は、各チャンネルで周期的にセグメントを配信する。各セグメントの配信契機を同期できない場合、再生中に途切れ時間が発生する。例えば、Harmonic Broadcasting [6] において、遅延揺らぎの影響でセグメントの配信を同期できない場合、再生中に途切れ時間が発生する [15]。

TeleCaS において、図 2 に示す FB 法を実際のネットワーク環境で導入する場合の配信スケジュールを図 5 に示す。実際のネットワーク環境でスケジューリング手法を用いて動画データを配信する場合、サーバはセグメントの受信に必要な情報 (以下、配信開始部) をクライアントに送る必要があるが、配信スケジュールには含まれていない。配信開始部を考慮してセグメントをスケジューリングする場合、配信開始部を配信するチャンネルと配信しないチャンネルとの間で、配信開始時刻にずれが発生する。このとき、計算機上でのシミュレーション環境では想定していない再生中の途切れが発生する。

4.3 複数動画の同時再生方式

TeleCaS で複数動画の同時再生に対応するためには、サーバが配信する複数動画の判別方法の提案、および複数動画の逐次再生方式の実現という二つの課題がある。以降で詳しく述べる。

4.3.1 複数動画の判別方法

複数動画の配信において、サーバは一つのチャンネルで複数の動画のセグメントを配信するようにスケジューリングすることで、各動画の配信契機を同期できる。クライアントは、複数動画を同時に視聴するため、複数のチャンネルから同時に受信したセグメントを各動画に復元する必要がある。これまでの *TeleCaS* のデータフォーマットは、配信開始部、およびセグメントを等分割した部分 (以下、サブセグメント) をセグメントに復元する情報 (以下、情報部) の二つで構成されていた。しかし、各セグメントがどの動画のどの部分かを示す情報はデータフォーマットに含まれていなかった。このため、クライアントは動画ごとに異なる再生プレイヤーに動画データを送信できず、複数の動画を同時に再生できない。

4.3.2 複数動画を考慮した逐次再生方式の実現

従来の *TeleCaS* では、複数の再生プレイヤーを同時に読み込むことができず、複数の動画を同時に逐次再生できない問題がある。以下で、*TeleCaS* における逐次再生の手順を説明する。初めに、ブラウザ上の再生プレイヤーを読み込む。次に、情報部の情報をもとに、クライアントはサーバから受信したサブセグメントを配信前のセグメントに復元する。動画の先頭部分のサブセグメントを受信すると、クライアントは分割配信の通信プロトコルを逐次再生の通信プロトコルである Hypertext Transfer Protocol (HTTP) に変換して、サブセグメント単位で動画データをブラウザに送信することで、逐次再生を実現できる。

しかし、従来の *TeleCaS* では、同時に一つの再生プレイヤーしか読み込まず、一つの動画データしか逐次再生できないという問題がある。このため、受信した複数動画を同時に再生できず、複数の動画配信に対応したスケジューリング手法 [8] を *TeleCaS* に適用できない。

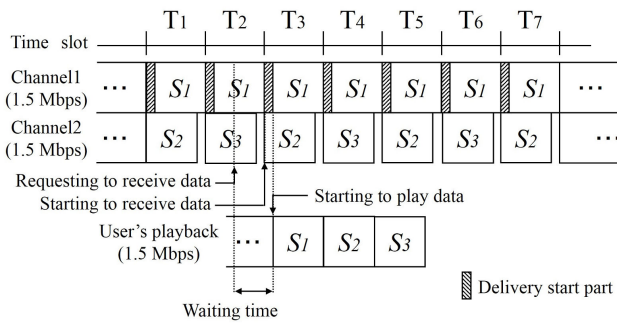


図 6 TeleCaS における同期配信の例 (FB 法)

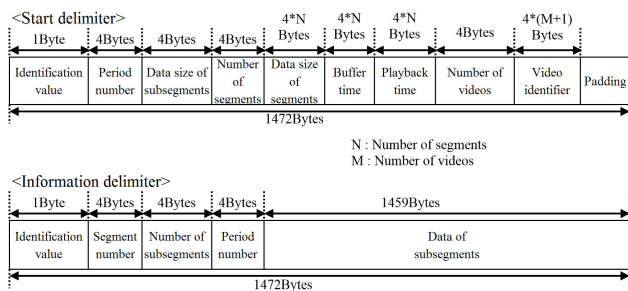


図 7 データフォーマット

5. 実現方式

5.1 セグメントの配信契機を考慮した同期配信方式

4.2 節の課題に対処するため、我々の研究グループでこれまでに提案した配信契機の同期方法 [14] を用いて、複数動画の同期配信を実現する。この方法では、図 6 に示すように、各セグメントの配信契機が同期し、途切れ時間を短縮できる。また、複数の動画配信では、動画データを配信するチャンネル数が増加するため、単一の動画配信と比べて配信開始時刻のずれによる影響は大きくなる。そこで、複数の動画配信において、複数動画を分割した各セグメントを単一動画のセグメントとして処理する。これにより、複数の動画配信におけるセグメントの処理は単一の動画配信と同様の処理となり、複数の動画配信におけるセグメントの配信契機の同期を実現できる。

5.2 複数動画の同時再生方式

5.2.1 複数動画の判別方法

4.3.1 節の課題に対処するため、配信開始部を拡張して、複数動画の配信に対応する。実現方式のデータフォーマットを図 7 に示す。データフォーマットは、従来と同様に、配信開始部と情報部で構成する。従来の配信開始部は、識別子、タイムスロット番号、サブセグメントデータサイズ、セグメント番号、セグメントデータサイズ、バッファ時間および再生時間で構成していた。また、情報部は、識別子、サブセグメント番号、タイムスロット番号およびサブセグメントで構成していた。実現方式では、配信開始部を拡張

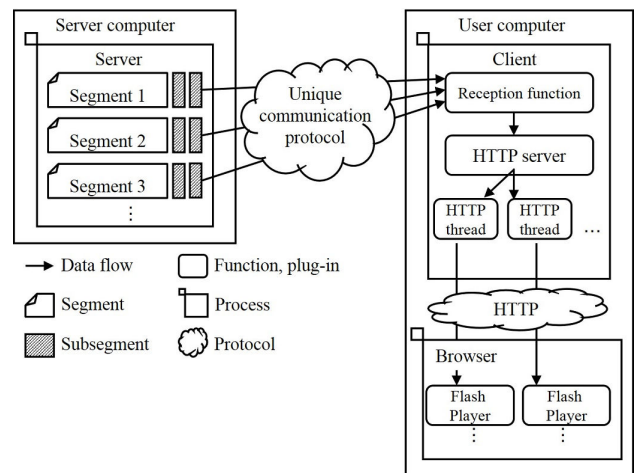


図 8 複数動画の分割放送型配信の構成

して動画数および動画識別子を追加することで、4.3.1 節の課題を解決した。クライアントは、動画識別子を用いることで、各セグメントがどの動画に対応するかを識別できる。

また、TeleCaS 上で MV-B 法のアイドル時間を実現するため、サーバはアイドル時間中に、配信する動画データと関連がない別の動画データをクライアントに配信する。このとき、クライアントが視聴する動画数と実際に配信する動画数は異なるため、クライアントは動画数のデータが必要となる。そこで、追加した動画数のデータを用いることで、クライアントは動画数と同じ数の動画データをブラウザに送信し、スケジューリング手法で設定した数の動画データを再生できる。図 3 における MV-B 法の配信スケジュール例では、サーバは 5 種類の動画データを同時に配信する。しかし、実装した TeleCaS で MV-B 法を実現する場合、配信スケジュールを維持するため、5 種類の動画とは異なる動画をアイドル時間に配信する必要がある。そこで、クライアントは動画数のデータをもとに、クライアントが同時に再生する 5 種類の動画データをブラウザに送信することで、クライアントはスケジューリング手法で設定した数の動画を同時に再生できる。

5.2.2 複数動画を考慮した逐次再生方式の実現

4.3.2 節で述べた課題に対処するため、ブラウザと通信を行う処理部をスレッド化する。TeleCaS における複数動画の分割放送型配信の構成を図 8 に示す。サーバは、分割放送型配信の通信プロトコルを使用して、サブセグメント単位でクライアントに動画データを送信する。クライアントは、ブラウザの読み込みから再生プレイヤーに動画データを転送するまでの部分をスレッド化した上で、並列に処理する。これにより、ブラウザに埋め込まれた複数の再生プレイヤーを同時に読み込むことができ、クライアントは複数の再生プレイヤーと同時に通信できる。また、配信開始部の動画識別子をもとに、各動画で先頭のサブセグメントを受信すると、クライアントは分割配信の通信プロトコルを逐次再生の通信プロトコルである HTTP に変換して、対応

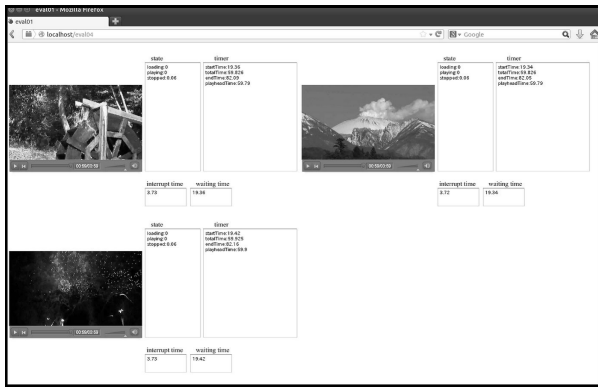


図 9 TeleCaS の再生画面

する再生プレイヤーにサブセグメントを送信することで、複数の動画を同時に逐次再生できる。以上より、分割放送型配信における複数動画の逐次再生を実現できる。

5.3 TeleCaS における実現方式の実装

5.3.1 想定環境

分割放送型配信システム *TeleCaS* において、複数動画の同期配信および同時再生を実装するため、想定する環境を以下に示す。

- (1) サーバは、複数のチャンネルを用いてクライアントにセグメントを繰返し配信する。
- (2) クライアントは、ブラウザ上で動画プレイヤーを用いて動画を視聴する。
- (3) クライアントは、動画の再生を開始すると最後まで途切れずに再生できる。
- (4) クライアントは、すべてのセグメントを蓄積するために十分なバッファをもつ。

5.3.2 実装内容

5.1, 5.2 節で述べた実現方式をもとに、*TeleCaS* で複数動画の同期配信および同時再生に対応するように実装した。*TeleCaS* は、チャンネル数、帯域幅、およびセグメントの配信順序が異なるスケジューリング手法を導入できる。*TeleCaS* における再生画面のスクリーンショットを図 9 に示す。図 9 は、3 種類の動画データを同時に再生している様子を示す。動画再生には、Adobe Flash Player を用いた。

6. 評価

6.1 評価環境

TeleCaS の評価環境を図 10 に示す。実際のネットワーク環境を想定して、*TeleCaS* を導入した計算機を複数台用いてネットワークを構成した。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は、帯域制御機能をもつ Dummynet を利用できる計算機を経由することで、ネットワークの帯域幅を制御する。本研究では、Dummynet によ

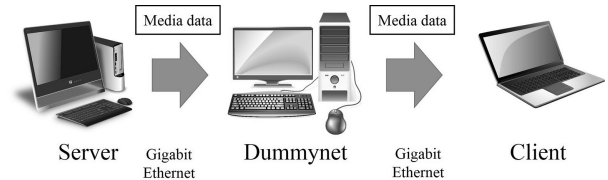


図 10 評価環境の構成

り帯域幅を制御した上で、単位時間あたりに受信するデータ量を帯域幅として算出する。また、マルチキャスト通信を用いることで、放送型による動画配信を実現した。

6.2 TeleCaS を用いた評価

実際のネットワーク環境でスケジューリング手法の有用性を確認するため、FB 法、MV-B 法、および MV-R 法の 3 種類のスケジューリング手法を用いて、*TeleCaS* 上で待ち時間および途切れ時間の評価を行う。計算機シミュレーションで算出される待ち時間の値を理想値とし、*TeleCaS* による配信で算出される待ち時間と途切れ時間の値を測定値とする。各スケジューリング手法における待ち時間の理想値は、3.1 節で説明したように、第 1 セグメントの受信時間の半分とした。理想値の算出が必要となる第 1 セグメントのデータサイズは、スケジューリング手法にもとづき、評価に用いる各動画データを分割した第 1 セグメントのデータサイズの平均値とした。

クライアントが一続きの動画データを再生終了まで途切れなく再生するため、再生中に発生する途切れ時間の分だけ動画の再生開始を遅らせる必要がある。そこで、待ち時間の測定値 (以下、実待ち時間) は、*TeleCaS* で測定する待ち時間と途切れ時間の合計値とした。また、*TeleCaS* で測定する待ち時間と途切れ時間はクライアント計算機 3 台の平均値とし、クライアントが動画データの受信をサーバに要求した時刻から測定を開始する。また、評価では、逐次再生方式を用いることで、第 1 セグメントのうち先頭部分となるサブセグメントを受信すると再生を開始できる。

6.3 待ち時間に関する評価

複数動画の配信において 3 種類のスケジューリング手法を導入し、実際のネットワーク環境における待ち時間の短縮効果を確認する。動画のデータサイズを変化させた場合の実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、縦軸はクライアント計算機で各動画の実待ち時間を同一条件で 10 回求めた平均値とした。また、再生時間が異なる動画データを使用して、動画のデータサイズを変化させた。MV-B 法および MV-R 法の配信スケジュールは、図 3 および図 4 とそれぞれ同じとする。動画のデータサイズ以外の配信条件は、以下の通りである。

- (1) チャンネル帯域幅 : 1.5Mbps
- (2) 動画の再生レート : 1.5Mbps

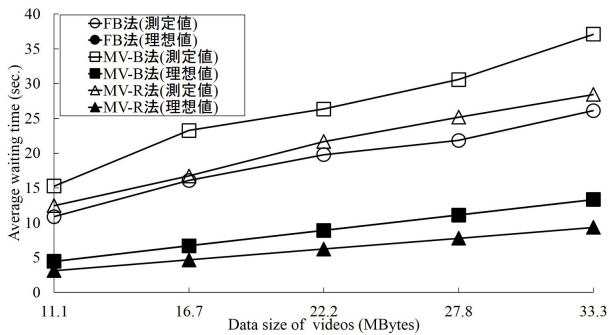


図 11 第 1 セグメントのデータサイズと平均待ち時間

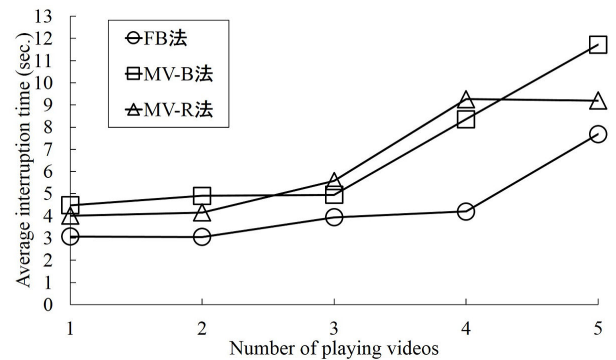


図 12 再生動画数と平均途切れ時間

- (3) ブラウザ内の再生プレイヤー数：5 個
- (4) チャンネル数：15
- (5) 動画数：5 種類

まず、理想値について説明する。図 11 より、FB 法と MV-B 法における平均待ち時間の理想値はどちらも等しい。15 個のチャンネルで 5 種類の動画データを配信する場合、FB 法と MV-R 法の動画分割数は 7 で同じであり、第 1 セグメントの大きさは等しいため、第 1 セグメントを受信する時間の半分である平均待ち時間の理想値は一致する。一方で、MV-R 法の動画分割数は 10 であり、第 1 セグメントの大きさは他の二つの手法より小さくなるため、MV-R 法における平均待ち時間の理想値は他の手法より短くなる。

次に、測定値の評価結果について説明する。図 11 より、すべてのスケジューリング手法について、測定値は理想値に比べて長大化する。例えば、動画データサイズが 22.2 MBytes における FB 法の理想値と測定値について考える。FB 法の理想値は、動画データサイズが 22.2 MBytes の第 1 セグメント (約 3.18 MBytes) を受信する時間の半分であり、約 8.90 秒となる。一方、FB 法の測定値は、待ち時間が約 8.81 秒、途切れ時間が約 10.97 秒であり、実待ち時間は合計で 19.78 秒となる。この時間差は、シミュレーション環境では考慮していない動画再生中の途切れ時間の影響によるものである。

また、動画データサイズが 22.2 MBytes の場合、MV-R 法の測定値は、待ち時間が約 6.21 秒、途切れ時間が約 15.46 秒であり、実待ち時間は合計で約 21.67 秒となるため、FB 法より長い。一方、理想値では、MV-R 法の待ち時間は FB 法より短い。測定値では、動画再生中に発生する途切れ時間の影響により、スケジューリングが FB 法に比べて複雑である MV-R 法の実待ち時間は大きく長大化する。

6.4 同時再生する動画数の変化による途切れ時間の評価

同時再生する動画数の変化による途切れ時間を評価する。6.3 節より、動画再生中に発生する途切れ時間は実待ち時間に大きな影響を与える。そこで、同時再生する動画数の変化による途切れ時間を評価し、再生中に発生する途

切れの要因について考察する。

同時再生する動画数を変化させた場合の途切れ時間の変化を図 12 に示す。横軸は同時再生する動画数、縦軸はクライアント計算機で各動画の途切れ時間を同一条件で 10 回求めた平均値とした。動画データのデータサイズは 11.1 MBytes とし、これ以外の条件は 6.3 節と同じとした。

まず、同時再生する動画数を変化させた場合の途切れ時間の変化について評価する。図 12 より、FB 法では、同時再生する動画数が 1 のときに途切れ時間は約 3.06 秒、同時再生する動画数が 5 のときに途切れ時間は約 7.68 秒となる。同時再生する動画数が増加すると、途切れ時間は長大化する。これは、同時再生する動画数の増加により、ブラウザへ同時に送信する動画のデータ量が増えてシステム全体の処理量が増加することで、途切れ回数が増加したためである。

次に、各スケジューリング手法における途切れ時間の変化について評価する。FB 法では、ほぼ線形的に途切れ時間が長大化した。TeleCaS で FB 法を適用する場合、各動画の配信スケジュールはすべて同じになり、すべてのタイムスロットにおいて受信する各動画データのセグメント数は等しい。このため、動画数の増加に比例して途切れ時間は長大化する。一方で、MV-B 法、および MV-R 法では、一定数以上の動画を同時再生すると、途切れ時間は急激に長大化する。MV-B 法、および MV-R 法は、複数の動画データを同期配信するようにセグメントをスケジューリングするため、配信スケジュールは複雑になる。このとき、各動画データがタイムスロットごとに受信するセグメント数は、タイムスロットによって偏りが生じる。このため、クライアントがデータの受信を開始する契機によって、途切れ時間は大きく長大化すると考えられる。

6.5 受信を開始するタイムスロットの変化による途切れ時間の評価

6.4 節より、動画再生中に発生する途切れ時間は、クライアントが受信を開始するタイムスロットに関係する。そこで、クライアントが受信を開始するタイムスロットを変

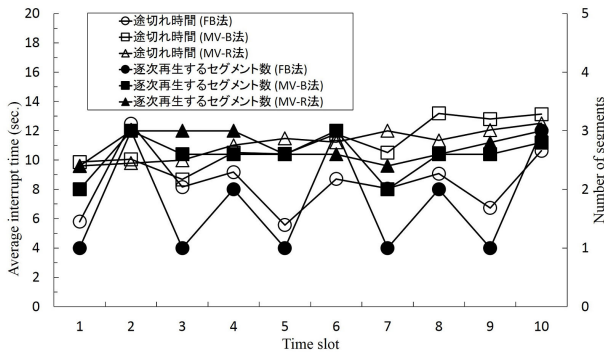


図 13 受信を開始するタイムスロットと途切れ時間

化させた場合における途切れ時間の変化を図 13 に示す。横軸は、クライアントが受信を開始するタイムスロット T ($1 \leq T \leq 10$) とする。縦軸は、あるタイムスロットでクライアントが受信を開始した場合に発生する途切れ時間を同一条件で 10 回求めた平均値、および各動画においてあるタイムスロットで受信を開始する場合に再生中に逐次再生を行うセグメント数の平均値である。

図 13 より、FB 法は、逐次再生を行うセグメント数は他の二つの手法より少なく、動画再生中に発生する途切れ時間は短い。一方、MV-B 法、および MV-R 法は、逐次再生するセグメント数はほぼ同じであるため、動画再生中に発生する途切れ時間はほぼ等しい。以上より、逐次再生を行うセグメント数の変化の大きさに応じて、動画再生中に発生する途切れ時間は大きく変化することがわかる。

7. おわりに

本研究では、分割放送型配信システム *TeleCaS* において複数動画の同期配信を実現するため、セグメントの配信契機を考慮した同期配信方式、および複数動画の同時再生方式を提案した。また、これらの対処を *TeleCaS* で実装して評価を行い、実待ち時間の短縮効果を評価した。まず、3 種類のスケジューリング手法を用いて、実待ち時間について評価した。評価結果より、*TeleCaS* 上で複数動画に対応したスケジューリング手法を導入することで、動画再生を開始するまでの待ち時間は短縮したが、動画再生中に途切れ時間が発生して、実待ち時間は長大化した。次に、同時再生する動画数の変化による途切れ時間について評価した。評価結果より、同時再生する動画数の増加にともない、システム全体の処理量が増加し、途切れ時間は長大化した。最後に、クライアントが受信を開始するタイムスロットの変化による途切れ時間を比較して評価した。評価結果より、再生中に逐次再生を行うセグメント数の増加に応じて、途切れ時間は長大化した。

今後の予定として、動画再生中に発生する途切れ時間を短縮する方法の検討、異種クライアント環境における分割放送型スケジューリング手法 [9] の導入、およびパケット損失を考慮したスケジューリング手法 [10][11] の導入が

ある。

謝辞 本研究の一部は、総務省戦略的情報通信研究開発推進事業 (SCOPE) 「放送通信融合環境による次世代モバイルビデオオンデマンド配信の研究開発」(課題番号: 132107005)、および文部科学省科学研究費補助金 (若手研究 (B)) 「分割放送型ストリーミング配信における待ち時間を短縮するスケジューリング技術の構築」(課題番号: 26730059) の成果である。ここに記して謝意を表す。

参考文献

- [1] NHK : NHK オンデマンド, NHK(オンライン), 入手先<<https://www.nhk-ondemand.jp>>(参照 2014-1-28).
- [2] YouTube : YouTube, Youtube(オンライン), 入手先<<https://www.youtube.com>>(参照 2014-1-28).
- [3] 衛星放送協会 : 衛星放送協会のしくみ | 視聴ガイド | 衛星放送協会, 衛星放送協会 (オンライン), 入手先<<http://www.eiseihoso.org/guide/howto.html>>(参照 2015-1-20).
- [4] 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ンネル数について, 日本データベース学会 Letters, Vol.4, No.3, pp.5-8 (2005).
- [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] L. Juhn and L. Tseng.: Harmonic broadcasting for video-on-demand service, IEEE Trans. on Broadcasting, vol.43, no.3, pp.268-271 (1997).
- [7] 義久智樹, 塚本昌彦, 西尾章治郎 : 分割放送方式におけるチャンネルの帯域幅を考慮した連続メディアデータの分割手法, 情報処理学会論文誌 B, Vol. 通信 J91-B, No.3, pp.300-308 (2008).
- [8] Chen, Y. and Huang, K.: Multiple Videos Broadcasting Scheme for Near Video-on-Demand Services, Proc. IEEE Int. Conf. on Signal Image Technology and Internet Based Systems 2008 (SITIS '08), pp.52-58 (2008).
- [9] Y. Gotoh, T. Yoshihisa, H. Taniguchi, and M. Kanazawa.: A Scheduling Method for Heterogeneous Clients on Media Data Broadcasting, International Journal of Pervasive Computing and Communications (IJPCC), Vol.9, Issue.2, pp.98-114 (2013).
- [10] 安里諒, 青木輝勝, 沼澤潤二 : 受信エラー耐性を有する分割放送型配信手法に関する一検討, 電子情報通信学会技術研究報告 : 信学技報, Vol.109, pp31-36 (2010).
- [11] 後藤佑介, 谷口秀夫 : 連続メディアデータの放送型配信における誤り耐性を考慮したスケジューリング手法, 情報処理学会研究報告 EIP, Vol.2012-EIP-57, No.24, pp1-8 (2012).
- [12] Gnzo : fv ソリューション, Gnzo (オンライン), 入手先<<http://gnzo.biz/service/fv-solution>>(参照 2014-12-28).
- [13] 第一興商 : DAM ★とも [club DAM.com], DAM ★とも (オンライン), 入手先<<http://www.clubdam.com/app/damtomo/top/Index.do>>(参照 2014-12-28).
- [14] 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).
- [15] 青木輝勝, 安里諒, 沼澤潤二 : 分割放送スケジューリング方式における遅延揺らぎ特性の解析とその改善に関する一検討, 電子情報通信学会論文誌 B, Vol.J93-B, No.1, pp112-122 (2010).