

選択型コンテンツの放送型配信システムの実現

後藤 佑介¹ 山本 泰平¹ 谷口 秀夫¹

概要: 近年, 地上波デジタルテレビ放送やワンセグ放送といったデジタル放送の普及にともない, ユーザがコンテンツを選択して視聴する選択型コンテンツの放送型配信に対する注目が高まっている. ユーザは, 選択型コンテンツを再生することで, 自分の嗜好に合わせてコンテンツを視聴できる. 一方, サーバが選択型コンテンツを放送型で配信する場合, 複数のチャンネルでいくつかのコンテンツを同時に配信するため, 途切れのない再生に必要な帯域幅は増加し, 再生中の待ち時間は長大化する. これらの影響を抑えるため, コンテンツの視聴順序を考慮してコンテンツの配信計画を決定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. しかし, これまでに提案されたスケジューリング手法は, 再生開始の契機や逐次再生といった実際のネットワーク環境で必要となる機能を考慮していない. 本研究では, 実際のネットワーク環境を想定して選択型コンテンツを放送型で配信するシステムを提案する. 本システムでは, 制御情報を考慮して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の二つを実現する.

1. はじめに

近年, 地上波デジタルテレビ放送や携帯電話を用いたワンセグ放送といったデジタル放送の普及にともない, ユーザがコンテンツを選択して視聴する選択型コンテンツの放送型配信に対する注目が高まっている [1]. 例えば, 以下のような選択型コンテンツが考えられる.

- 2 択のクイズ番組で, 選択肢が提示され, ユーザが回答を選択する. ユーザの回答が正解であれば正解の映像を表示し, 不正解であれば不正解の映像を表示する.
- ニュース番組で, ユーザはすべてのニュースの見出しを視聴した後に, スポーツニュースの見出しと経済ニュースの二つから視聴するコンテンツを一つ選択する. ユーザがスポーツニュースを選択した場合, ユーザはさらに野球ニュースとサッカーニュースの二つから視聴するコンテンツを一つ選択する.

ユーザは, 選択型コンテンツを再生することで, 自分の嗜好に合わせてコンテンツを視聴できる. 一方, サーバが選択型コンテンツを放送型で配信する場合, 複数のチャンネルでいくつかのコンテンツを同時に配信するため, 配信するコンテンツの数が増加すると, 途切れなく再生するために必要な帯域幅は増加する. このとき, サーバが配信に必要な帯域幅が使用できる帯域幅を上回ると, コンテンツの配信時間は長大化し, ユーザのコンテンツ再生中に待

ち時間が発生する. そこで, コンテンツの視聴順序を考慮して必要な帯域幅の増加や再生時に発生する待ち時間の長大化を抑える配信計画 (以下, 配信スケジュール) を決定するためのスケジューリング手法がいくつか提案されている. しかし, これらのスケジューリング手法は, 実際のネットワークで想定されるコンテンツの制御情報の追加や逐次再生を考慮していない.

本研究では, 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放送型で配信するシステムを提案する. 本システムでは, クライアントがデータを再生するために必要となるコンテンツの制御情報を考慮して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の二つを実現する.

本論文は, 以下のように構成される. 2 章で放送型配信について説明し, 3 章でスケジューリング手法について説明する. 4 章で実現方式について述べ, 5 章で実装を行う. 最後に 6 章で本論文をまとめる.

2. 選択型コンテンツ

2.1 視聴順序グラフ

選択型コンテンツでは, ユーザはいくつかのコンテンツを選択しながら続けて視聴するため, 視聴するコンテンツに順序が生じる. 一般に, ユーザは現在再生しているコンテンツが終了してから次のコンテンツを再生する. このため, 選択型コンテンツの表記方法として, 状態遷移グラフが適している [2]. 本研究では, 選択型コンテンツの視聴順序を表記する状態遷移グラフを視聴順序グラフと呼ぶ.

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

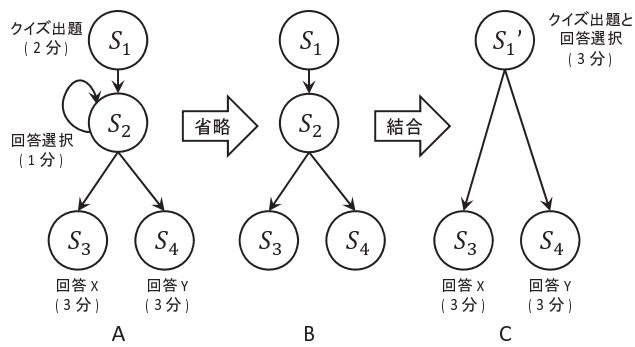


図 1 2 択クイズ番組の視聴順序グラフおよび簡単化
 Fig. 1 Play-sequence graph.

視聴順序グラフでは、各状態が各コンテンツを再生している状態を示し、コンテンツの再生が終了すると次の状態に遷移する。また、一定の方向に時間軸を示し、再生開始時刻から各状態に遷移するまでの時間を示す。ただし、時間軸をさかのぼる状態遷移がある場合、この限りではない。

例えば、2 択クイズ番組の視聴順序グラフは図 1-A で表される。状態の再生時間は、次の状態に遷移するまでの最短時間を示している。状態 S_1 は出題の映像を再生している状態であり、2 分後、再生を終了すると状態 S_2 に遷移する。 S_2 は、選択肢となる回答 X および Y の内容を表示している状態であり、再生時間が 1 分のコンテンツである。ユーザは、 S_2 の再生中に回答を選択する。再生終了後、回答として X を選択すると S_3 、Y を選択すると S_4 といったように、選択したコンテンツに応じた状態に遷移する。ユーザがコンテンツを選択しなかった場合には、再び S_2 に戻ることや、どちらかの回答を自動的に選択することが考えられる。 S_3 は回答として X を選択した場合の状態であり、正解の映像を再生する。一方、 S_4 は回答として Y を選択した場合の状態であり、不正解の映像を再生する。

2.2 視聴順序グラフの簡単化

視聴順序グラフは、放送する番組に応じて様々な形状がある。このため、コンテンツの状態の数が増加すると視聴順序グラフは複雑な構造になり、配信スケジュールの作成は困難になる。そこで、視聴順序グラフに対して「状態遷移の省略」、「状態の結合」、および「状態の分割」といった 3 種の操作を行い、多分木に変形する。

「状態遷移の省略」では、クライアントは、受信したコンテンツをバッファに蓄積することで、任意の時間に再生できる。このため、時間軸をさかのぼる状態遷移は、該当するコンテンツがバッファに蓄積されていれば省略できる。「状態の結合」では、再生終了後に分岐を伴わない状態を次の状態と結合して、1 個の状態として表記できる。「状態の分割」では、ある状態を 2 個の状態に分割することで、分岐のない逐次的な状態遷移として表記できる。状態の結合と異なり、サーバは、再生時間が他の状態より長い

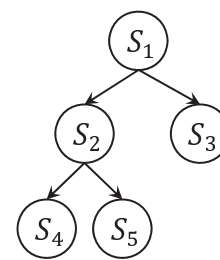


図 2 2 択クイズ番組の視聴順序グラフ例
 Fig. 2 Example of play-sequence graph for quiz program.

状態を分割することで、分岐した他の状態と再生開始時刻の同期をとることができ、配信スケジュールの作成は容易になる。例えば、図 1-A の 2 択クイズ番組の視聴順序グラフは、「状態遷移の省略」および「状態の結合」により、図 1-C に簡単化できる。情報量を減らすことなく視聴順序グラフを簡単な形状に変形することで、配信スケジュールの作成は容易になる。

3. 放送型配信におけるスケジューリング手法

3.1 スケジューリング技術の必要性

選択型コンテンツの放送型配信では、サーバはいくつかのコンテンツを同時に配信する。実際のネットワーク環境で配信する場合、コンテンツの再生レートや配信するコンテンツの視聴順序グラフは配信開始前に決まっている。また、サーバが使用できる帯域幅には上限があり、配信に必要となる帯域幅が使用できる帯域幅を上回ると、コンテンツの受信時間が再生時間より長くなり、再生時に待ち時間が発生する。このため、実際のネットワーク環境で想定される配信条件を考慮して、どのコンテンツをどのタイミングで配信するかを定めた配信スケジュールを決定するスケジューリング手法が必要となる。

3.2 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

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は、これまでいくつか提案されてきた。次節より、既存のスケジューリング手法である単純手法、Contents Cumulated Broadcasting (CCB) [2] 法、および CCB-Considering Bandwidth (CCB-CB) [3] 法の 3 つを順番に説明し、その長所と短所について述べる。

3.2.1 単純手法

単純手法では、視聴順序グラフをもとに、サーバがいくつかのチャンネルを用いて各コンテンツを再生開始時刻と同時に放送開始できるようにスケジューリングすることで、クライアントは途切れなく番組を再生できる。図 2 のような 2 択クイズ番組を単純手法で配信する場合の放送スケジュールを図 3 に示す。MPEG 2 で符号化されたコンテンツを想定し、再生レートを 5.0 Mbps とする。 C_1 およ

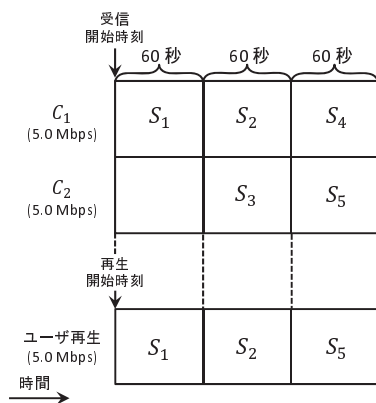


図 3 単純手法の配信スケジュール例

Fig. 3 Broadcast schedule under simple method.

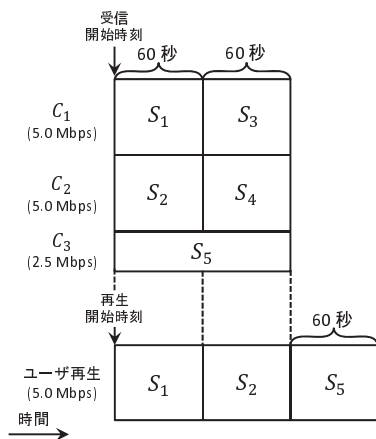


図 4 CCB 法の配信スケジュール例

Fig. 4 Broadcast schedule under CCB method.

び C_2 は放送チャンネルを示す. 単純手法でクライアントが途切れなく再生できるようにスケジューリングする場合, 各チャンネルの帯域幅は再生レートと等しく 5.0 Mbps であり, 各深さにおけるコンテンツの最大数が 2 であるため, 使用する帯域幅は $5.0 \times 2 = 10.0$ Mbps となる. コンテンツ S_1, \dots, S_5 の再生時間はそれぞれ 60 秒である. 時間の経過を右向きに示し, 番組の放送が始まると, C_1 では S_1, S_2 , および S_4 を順番に放送する. C_2 では, 配信開始時刻から S_3 の再生開始時刻までの最短時間である 1 分後までは何も放送せず, この後に S_3 および S_5 を順番に放送する.

3.2.2 CCB 法

CCB 法で配信する場合の配信スケジュールを図 4 に示す. CCB 法では, どの視聴順序を選択しても再生中に途切れが発生しないように配信スケジュールを作成する. また, 単純手法で放送していない時間に放送するようにコンテンツをスケジューリングすることで, 使用する帯域幅と放送時間との積を単純手法に比べて小さくする. 例えば図 4 の場合, C_1 で S_1 および S_3 , C_2 で S_2 および S_4 を再生レートと同じ 5.0 Mbps でスケジューリングする. また, S_5 は放送開始時刻から 120 秒かけて放送するため, C_3 は $5.0 \times 60/120 = 2.5$ Mbps となる. このとき, サーバが放

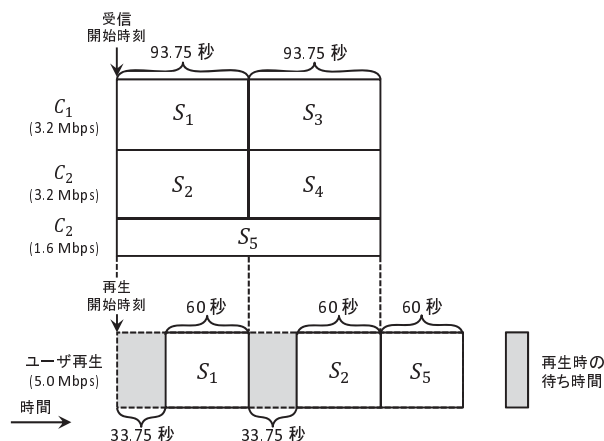


図 5 CCB 法の配信スケジュール例 (使用帯域幅 8.0 Mbps)

Fig. 5 Broadcast schedule under CCB method with interruption.

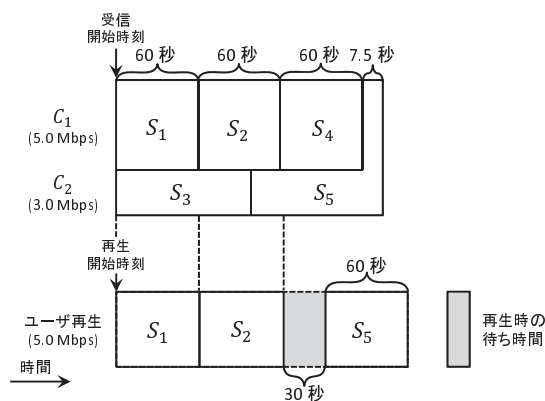


図 6 CCB-CB 法の配信スケジュール例 (使用帯域幅 8.0 Mbps)

Fig. 6 Broadcast schedule under CCB-CB method.

送に必要な帯域幅は $5.0 \times 2 + 2.5 = 12.5$ Mbps となる.

しかし, サーバが放送に必要な帯域幅が使用できる帯域幅を上回る場合, 再生時に待ち時間が発生する. 例えば, 使用できる帯域幅が 8.0 Mbps のとき, CCB 法で配信する場合の配信スケジュールを図 5 に示す. 各チャンネルの帯域幅は単純に $8.0/12.5 = 0.64$ 倍され, C_1, C_2 の帯域幅は $5.0 \times 0.64 = 3.2$ Mbps, および C_3 の帯域幅は $2.5 \times 0.64 = 1.6$ Mbps となる. チャンネルの帯域幅が再生レートを下回ると, コンテンツの受信時間が再生時間より長くなる. このとき, コンテンツの再生を開始しても再生終了までの途切れが発生するため, 途切れが発生しないように再生開始時刻を遅らせる必要があり, 再生時の待ち時間となる. 例えば, 図 3 の例において, ユーザが S_1, S_2 , および S_5 の順に再生する場合, 再生時の待ち時間は合計で $33.75 \times 2 = 67.5$ 秒となる.

3.2.3 CCB-CB 法

CCB-CB 法で配信する場合の配信スケジュールを図 6 に示す. CCB-CB 法では, 再生レートと等しい帯域幅のチャンネルをできるだけ多く確保することで, 必要となる帯域幅が使用できる帯域幅を上回る状況で CCB 法に比べ

表 1 スケジューリング手法の長所と短所
Table 1 Advantage and disadvantage in conventional methods.

	長所	短所
単純手法	(1) 配信スケジュールの作成が容易. (2) 帯域幅に制限がない場合, 待ち時間は 0.	(1) 分岐が多くなるにつれて, 必要なチャンネルの数が増加. (2) チャンネルによって, コンテンツを放送していない時間が発生. (3) 使用できる帯域幅に制限がある場合, 待ち時間が発生.
CCB 法	(1) コンテンツを放送していない時間はない. (2) 帯域幅に制限がない場合, 待ち時間は 0.	(1) 帯域幅が異なるチャンネルが発生. (2) 視聴順序の分岐が多い場合, 配信スケジュールの作成が複雑. (3) 使用できる帯域幅に制限がある場合, 待ち時間が発生.
CCB-CB 法	(1) コンテンツを放送していない時間はない. (2) 使用できる帯域幅に制限がある場合, 待ち時間が一番短い.	(1) チャンネルの帯域幅が配信中に変化. (2) 配信スケジュールの作成が最も複雑.

て待ち時間を短縮する. 例えば, 使用できる帯域幅が 8.0 Mbps の場合, C_1 の帯域幅は 5.0 Mbps, C_2 の帯域幅は 3.0 Mbps となる. 番組の放送が始まると, C_1 では, 図 2 に示す視聴順序グラフの各深さでコンテンツ番号がもつとも若番となる S_1, S_2 , および S_4 のコンテンツを順番に放送する. また, C_2 では S_3 を 100 秒かけて放送した後, S_4 の再生終了時刻までの 80 秒間, S_5 を放送する. この後, C_1 と C_2 の帯域を合わせた計 8.0 Mbps の帯域幅を用いて, S_5 の残りを 7.5 秒間放送する. 図 6 の例において, ユーザが S_1, S_2 , および S_5 の順に再生する場合, 再生時の待ち時間は 30 秒となり, CCB 法に比べて待ち時間を約 55.5% 短縮できる.

3.3 スケジューリング手法の長所と短所

3.2.1~3.2.3 節で述べた単純手法, CCB 法, および CCB-CB 法それぞれについて, スケジューリング手法の長所と短所を表 1 にまとめた.

単純手法では, スケジューリングが容易であるという長所が挙げられる一方で, チャンネルによってはコンテンツを放送していない時間帯が発生する. また, 視聴順序グラフの分岐が多くなるにつれて, 必要なチャンネルの数は増加し, 必要となる帯域幅は増加する.

CCB 法では, すべてのチャンネルでコンテンツを放送しない時間帯は発生せず, 単純手法に比べてチャンネルを有効に利用できる. しかし, 使用できる帯域幅に制限がある場合, チャンネルの帯域幅が再生レートを下回ると待ち時間が発生する.

CCB-CB 法では, 使用できる帯域幅を考慮した配信スケジュールを作成することで, 単純手法や CCB 法に比べてコンテンツ再生時に発生する待ち時間を短縮できる. しかし, 各チャンネルの帯域幅が放送中に変化するため, 配信スケジュールの作成は複雑になる.

3.4 非選択型コンテンツとの違い

非選択型コンテンツとして, 音声や映像といった, 途切れなく再生する一続きのデータである連続メディアデータが

挙げられる. 連続メディアデータの放送型配信では, データをいくつかの部分に分割し, 複数のチャンネルで同時に繰り返し放送して待ち時間を短縮する手法は幾つか提案されている [4], [5], [6], [7], [8]. Harmonic Broadcasting (HB) [9] 法では, データをデータサイズの等しい幾つかの部分に分割し, 帯域幅の異なる幾つかのチャンネルを用いて同時に放送することで, 待ち時間を短縮している. 分割したデータを各チャンネルで繰り返して放送することで, クライアントは初めのデータを受信する機会が増え, 待ち時間を短縮できる. データサイズの等しい幾つかの部分に分割して放送する手法として, 他に Pagoda Broadcasting [10] 法, Fuzzycast [11] 法, Optimized Periodic Broadcast (OPB) [12] 法などがある. Pyramid Broadcasting [13] 法および Asynchronous Harmonic Broadcasting (AHB) [14] 法では, データサイズが異なる幾つかの部分に分割してスケジューリングすることで, 待ち時間を短縮する.

これらのスケジューリング手法は, 視聴順序グラフに従って複数のコンテンツを配信することを想定していない. また, 選択型コンテンツの放送型配信では, 現在再生中のコンテンツと次に再生するコンテンツとの間で発生する待ち時間を許容している点が連続メディアデータの放送型配信と大きく異なる.

4. 実現方式

4.1 課題

これまでに提案されてきたスケジューリング手法は, 計算機上でのシミュレーション環境を想定していた. 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, 課題として, (1) 制御情報の追加による配信スケジュールへの影響, および (2) 再生を開始する契機の二つが挙げられる. 以下で詳細を述べる.

4.1.1 制御情報の追加による配信スケジュールへの影響

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では, 異なるチャンネル間でコンテンツの配信開始時刻を同期する必要がある. しかし, 実際のネットワーク環境で選択型コンテンツを放送型で配信する場合, クライア

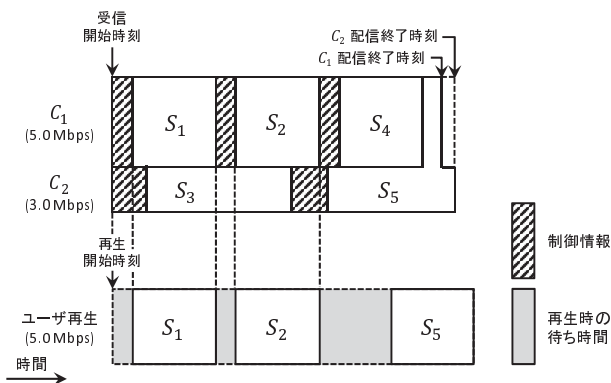


図 7 CCB-CB 法の配信スケジュール (制御情報あり)

Fig. 7 Broadcast schedule under CCB-CB method with control information.

ントが複数のコンテンツを同時に受信して再生処理するために必要となる制御情報を追加する必要がある。既存のスケジューリング手法において、制御情報を追加してコンテンツの配信をスケジューリングすると、コンテンツの配信開始契機を同期できず、途切れ時間が発生する。途切れ時間が発生する様子を説明するため、コンテンツの先頭部分の配信契機が同期するスケジューリング手法である CCB-CB 法を用いて、制御情報を考慮した場合の配信スケジュールを図 7 に示す。制御情報を考慮しない場合の配信スケジュールは図 6 と同じである。図 6, 図 7 はともに、図 2 の視聴順序グラフを用いて、2 つのチャンネル $C_1 = 5.0$ Mbps および $C_2 = 3.0$ Mbps で S_1, \dots, S_5 を配信する。

図 6 の配信スケジュールでは、データの受信を要求したクライアントは、 S_1 と S_2 の受信をすぐに開始し、 S_1 を再生しながら S_2 を先頭からバッファに保存する。また、 S_1 の再生が終了すると、次に再生する S_2 を先頭からバッファに保存しているため、クライアントは途切れなく再生を続けることができる。このとき、クライアントが途切れなく再生を開始するためには、 S_1 と S_2 の配信開始時刻は同じであることが重要である。

一方、図 7 の配信スケジュールでは、 C_1 と C_2 それぞれで配信する制御情報の数や使用できる帯域幅が異なるため、 C_1 と C_2 の配信終了時刻は同じにならない。このとき、再生中に発生する待ち時間は、図 6 の場合に比べて長大化する。以上のように、制御情報を考慮すると、サーバがあらかじめ作成した配信スケジュールに従って選択型コンテンツを配信できない問題が発生する。

4.1.2 データの再生を開始する契機

4.1.1 節で述べた課題は、サーバの配信契機とクライアントの受信契機が放送開始前に作成した配信スケジュールと異なることによる待ち時間の長大化である。さらに、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では、データの再生を開始する契機が受信を開始する契機と同じになることを想定して配信スケジュールを作成す

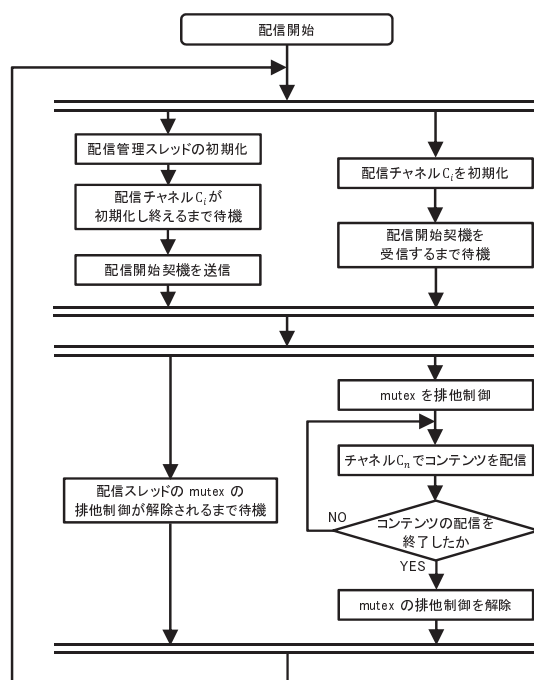


図 8 データの配信契機を同期する方式における配信サーバの処理流れ

Fig. 8 Flow chart of server in synchronizing timing of delivering data.

る。このため、実際のネットワーク環境で選択型コンテンツを放送型で配信する場合、コンテンツの受信が完了してから再生を開始するダウンロード型ではなく、クライアントがデータを受信しながら再生する逐次再生方式を実現する必要がある。

4.2 対処

4.2.1 制御情報を考慮してデータの配信契機を同期する方式

4.1.1 節で述べた制御情報による配信スケジュールへの影響に対処するため、データの配信契機を同期する方式を実現する。この方式では、各チャンネルの配信処理とデータの配信契機を同期する処理をスレッドでそれぞれ並列化した上で、並列化した処理の間で処理の開始と停止を通知する仕組みと並列化した処理を排他制御により同期する仕組みの二つを用いて、データの配信契機を同期する。

実現方式におけるサーバの処理流れを図 8 に示す。配信に用いるチャンネル数を N とし、 i ($1 \leq i \leq N$) 番目のチャンネルを C_i とする。サーバは、データの配信契機を管理する処理を管理スレッド、 C_i の配信処理を C_i の配信スレッドとしてそれぞれ並列化する。管理スレッドは、並列化した処理の間で処理の開始と停止を通知するために利用する変数 (以下、条件変数) を初期化し、配信スレッドごとに条件変数を設定する。次に、条件変数を用いて、すべての配信スレッドに配信開始を通知する。通知後、すべての配信スレッドにおいて、mutex と呼ばれる排他制御を管理す

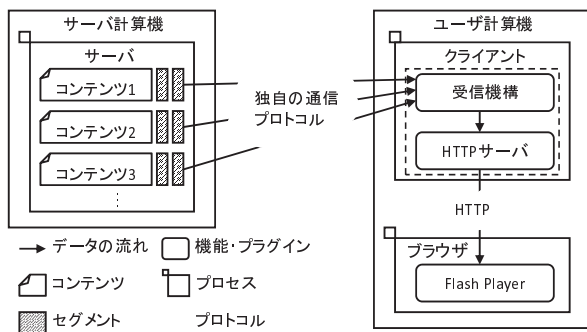


図 9 逐次再生の実現方式

Fig. 9 Configuration of sequential playback in proposed system.

る仕組みにより排他制御が解除され、すべての配信スレッドと処理が同期するまで待機する。一方、 C_i の配信スレッドは、管理スレッドから配信開始が通知されるまで待機する。管理スレッドから配信開始が通知されると、 C_i の配信スレッドは mutex を排他制御し、 C_i でコンテンツを配信する。コンテンツの配信が完了すると、 C_i の配信スレッドは mutex の排他制御を解除し、管理スレッドから条件変数により配信の再開が通知されるまで待機する。以上より、コンテンツの先頭部分の配信契機は同期する。

4.2.2 逐次再生方式

4.1.2 節で述べたデータの再生開始契機の課題に対処するため、動画を再生するプレーヤが逐次再生を行う上での取り決め（以下、逐次再生の通信プロトコル）を利用して、コンテンツをセグメントと呼ばれる部分に分割して配信する方式を実現する。実現方式では、逐次再生方式の通信プロトコルに対応するため、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の一つである Hypertext Transfer Protocol (HTTP) に変換して、分割配信で逐次再生方式の通信プロトコルを実現する。図 9 に、逐次再生の実現方式を示す。初めに、サーバは、コンテンツのデータをいくつかのセグメントに等分割する。また、分割配信の通信プロトコルでこれらのセグメントをクライアントに配信する。一方、クライアントでは、受信機構において分割配信の通信プロトコルを用いてセグメントを受信する。受信機構は、コンテンツ番号とセグメント番号をもとにセグメントを各コンテンツの先頭から順番にソートし、プロトコル変換機構に送信する。プロトコル変換機構は、逐次再生方式の通信プロトコルを用いてセグメントをブラウザに送信する。最後に、クライアントは、ブラウザを用いてコンテンツの逐次再生を行う。以上より、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に変換することで、分割配信における逐次再生を実現できる。

5. 実装

5.1 想定環境

選択型コンテンツの放送型配信システムを設計するにあ

たり、想定する環境を以下に示す。

- 放送されるデータは、選択型コンテンツである。
- 放送帯域は、一定である。
- サーバは、複数のチャンネルから同時にデータを放送できる。
- クライアントは、複数のチャンネルから同時にデータを受信できる。
- クライアントは、コンテンツの蓄積に十分な容量のバッファを持つ。
- クライアントは、逐次再生を行う。
- クライアントは、コンテンツの再生を開始すると、最後まで途切れずに再生できる。

5.2 データフォーマット

選択型コンテンツの放送型配信システムにおいて、データのフォーマットを図 10 に示す。

5.2.1 配信開始部

配信開始部は、データフォーマット識別値、コンテンツの種類識別値、分岐の深さ、分岐の数、コンテンツ数、各コンテンツのデータサイズ、各コンテンツの再生開始待ち時間、およびデータサイズ調整部で構成される。データフォーマット識別値は、配信開始部であることを識別するために先頭に設定する。コンテンツの種類識別値は、選択コンテンツがクイズ番組やドラマ番組といった番組形式を識別するために用いる。分岐の深さについて、クイズ番組の場合は問題数、ドラマの場合は分岐の回数を示す。分岐の数は、分岐のあるコンテンツにおける選択肢の数を示す。コンテンツ数は、視聴順序グラフを構成するコンテンツの合計数 n を示す。各コンテンツのデータサイズは、コンテンツごとのデータサイズの値を示す。各コンテンツの再生開始待ち時間は、コンテンツごとの再生開始待ち時間の値を示す。末尾にあるデータサイズ調整部は、あらかじめ設定したパケットサイズを情報部と同じにするために用いる。配信開始部のサイズは、合計で 1509 bytes となる。

5.2.2 情報部

情報部は、データフォーマット識別値、コンテンツ番号、セグメント番号、およびセグメントデータで構成される。データフォーマット識別値は、情報部であることを識別するために用いる。コンテンツ番号は、配信するデータがどのコンテンツであるかを識別するために用いる。セグメント番号は、分割されたセグメントのうち何番目のデータであるかを識別するために用いる。セグメントデータは、セグメント番号に対応したデータの情報である。情報部のサイズは、合計で 1509 bytes となる。

5.3 処理手順

選択型コンテンツの放送型配信システムでは、配信開始前に、選択型コンテンツの視聴順序グラフを構成する状態

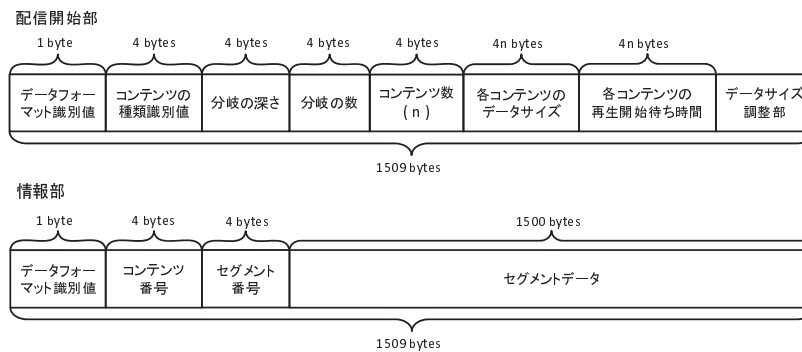


図 10 データフォーマット
Fig. 10 Data format.

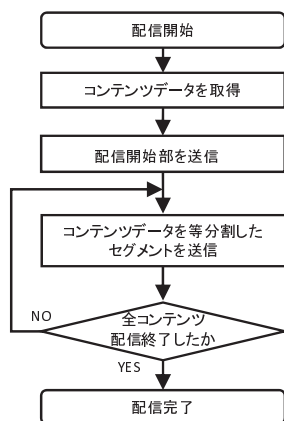


図 11 サーバのフローチャート
Fig. 11 Flowchart of server.

数, 使用する帯域幅, および再生レートといったさまざまなパラメータをもとにスケジューリング手法を決定し, 配信スケジュールを作成する. 次に, 作成した配信スケジュールをもとに, サーバはクライアントにコンテンツを配信し, クライアントはコンテンツを受信して再生する. 以下で, サーバとクライアントそれぞれの処理手順を説明する.

5.3.1 サーバの処理

サーバは以下の手順に従って配信処理を行う. サーバのフローチャートを図 11 に示す.

- (1) 配信するコンテンツのデータ情報を読み込む.
- (2) 配信開始部を配信する.
- (3) コンテンツデータを等分割したセグメントを情報部に格納して配信する.
- (4) すべてのコンテンツを配信し終わるまで (3) を繰り返す.

5.3.2 クライアントの処理

クライアントは以下の手順に従って配信処理を行う. クライアントのフローチャートを図 12 に示す.

- (1) サーバに受信要求を送り, データの受信を開始する.

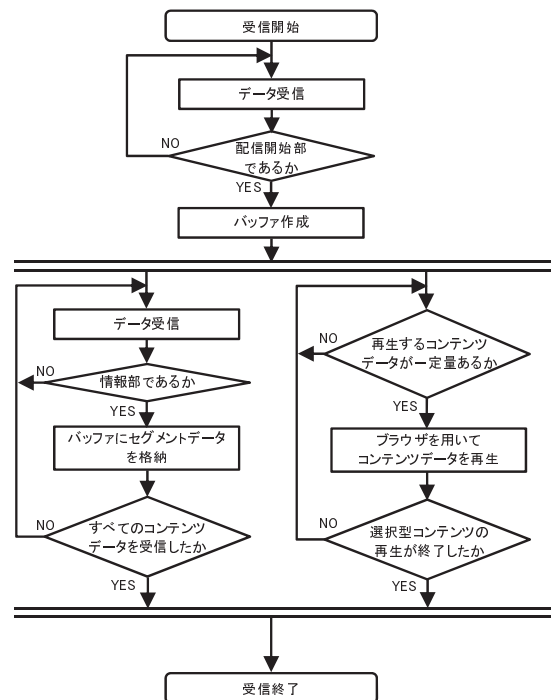


図 12 クライアントのフローチャート
Fig. 12 Flowchart of client.

- (2) サーバから受信したデータが配信開始部の場合, バッファの作成を行う. そうでなければ (1) に戻る.
- (3) サーバから受信したデータが情報部の場合, クライアントは受信したセグメントデータをバッファに格納する.
- (4) 再生するコンテンツのデータがバッファに先頭部分から一定量格納されていれば, ブラウザを用いてコンテンツを再生する. 格納されていなければ, コンテンツのデータがバッファに一定量格納されるまで待機してから再生する.
- (5) 次に再生するコンテンツがあれば, (4) に戻る. 次に再生するコンテンツがなければ, 受信を終了する.

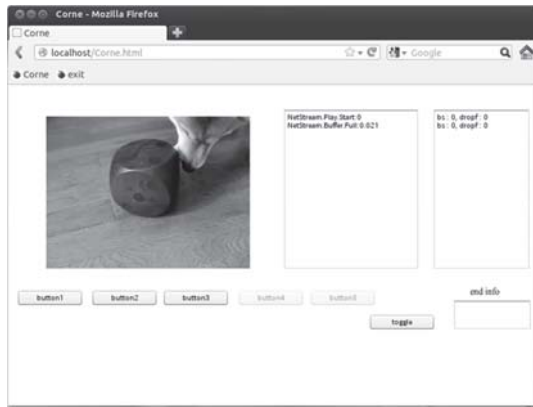


図 13 Corne のスクリーンショット
Fig. 13 Screen shot in Corne.

5.4 実装内容

4章で述べた実現方式をもとに、選択型コンテンツの放送型配信システムとして、**Contents-based Broadcasting system (Corne)** を実装した。Corne は、使用するチャンネル数、チャンネルの帯域幅、コンテンツのデータサイズ、および視聴順序グラフをもとにスケジューリング手法を決定し、配信スケジュールを作成してコンテンツを配信する。

Corne のスクリーンショットを図 13 に示す。Corne におけるサーバとクライアントのプログラムは、計算機環境の汎用性を考慮して C 言語で開発した。再生に使用するプレイヤーは、Web ブラウザ上で動作する Adobe Flash Player とした。また、データの再生制御機能をもつ ActionScript を用いることで、Web ブラウザ上でコンテンツ再生時の待ち時間と途切れ時間を測定できる。

6. おわりに

選択型コンテンツの放送型配信において、再生時の待ち時間を短縮するスケジューリング手法を実際のネットワーク環境で評価するため、放送型配信システム Corne を実装した。Corne は C 言語で設計されており、多くの計算機ネットワーク環境に対応できる。設計では、制御情報による配信スケジュールへの影響、および再生を開始する契機への対応について検討した上で、制御情報を考慮してデータの配信契機を同期する方式、および逐次再生方式の二つを提案し、Corne に実装した。

残された課題として、複数のスケジューリング手法を用いた Corne の性能評価がある。

謝辞

本研究は、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] 総務省: 情報通信白書平成 25 年版, 総務省 (オンライン), 入手先 <<http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h25/index.html>> (参照 2014-08-26).
- [2] 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).
- [3] 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).
- [4] Jinsuk, B. and Jehan, F.P.: A Tree-Based Reliable Multicast Scheme Exploiting the Temporal Locality of Transmission Errors, *Proc. IEEE Int. Performance, Computing, and Communications Conf. (IPCCC2005)*, pp.275-282 (2005).
- [5] Juhn, L.-S. and Tseng, L.M.: Fast Data Broadcasting and Receiving Scheme for Popular Video Service, *IEEE Trans. Broadcasting*, Vol.44, No.1, pp.100-105 (1998).
- [6] J.-F. Paris: An Interactive Broadcasting Protocol for Video-on-Demand, Proc. IEEE Int. Performance, Computing, and Communications Conference (IPCCC '01), pp.347-353 (2001).
- [7] J.-F. Paris, D.D.E. Long, and P.E. Mantey: Zero-delay Broadcasting Protocols for Video-on-demand, Proc. ACM Int. Multimedia Conf. (Multimedia '99), pp.189-197 (1999).
- [8] Zhao, Y., Eager, D.L., and Vernon, M.K.: Scalable On-Demand Streaming of Non-Linear Media, *Proc. IEEE INFOCOM*, Vol.3, pp.1522-1533 (2004).
- [9] Juhn, L.-S. and Tseng, L.M.: Harmonic Broadcasting for Video-on-Demand Service, *IEEE Trans. Broadcasting*, Vol.43, No.3, pp.268-271 (1997).
- [10] Paris, J.-F., Carter, S.W., and Long, D.D.E.: A Hybrid Broadcasting Protocol for Video on Demand, *Proc. Multimedia Computing and Networking Conf. (MMCN'99)*, pp.317-326 (1999).
- [11] Janakiraman, R. and Waldvogel, M.: Fuzzycast: Efficient Video-on-Demand over Multicast, *Proc. IEEE INFOCOM*, pp.920-929 (2002).
- [12] Mahanti, A., Eager, D.L., Vernon, M.K., and Stukel D.S.: Scalable On-demand Media Streaming with Packet Loss Recovery, *IEEE/ACM Trans. Networking*, Vol.11, No.2, pp.195-209 (2003).
- [13] Viswanathan, S. and Imilelinski, T.: Pyramid Broadcasting for Video on Demand Service, *Proc. SPIE Multimedia Computing and Networking Conf. (MMCN'95)*, pp.66-77 (1995).
- [14] Yoshihisa, T., Tsukamoto, M., and Nishio, S.: A Scheduling Scheme for Continuous Media Data Broadcasting with a Single Channel, *IEEE Trans. Broadcasting*, Vol.52, Issue 1, pp.1-10 (2006).