

データの細分割による効率的な放送スケジュールの作成手法

義久 智樹[†] 塚本 昌彦[†] 西尾 章治郎[†]

近年のデジタル放送の普及にともない、音楽や映画を配信する連続メディアデータ放送に対する注目が高まっている。連続メディアデータ放送では、一般に、クライアントはデータの再生が開始されるまで待つ必要があるため、待ち時間を短縮する様々な手法が提案されている。これらの手法の幾つかは、データを幾つか個の部分に等分割して、単一チャンネルを用いて初めの部分を頻繁に放送することで待ち時間を短縮している。しかし、データの分割数を多くして、効率的な放送スケジュールを作成することで既存手法よりも待ち時間を短縮できる。そこで本稿では、データを細分割して分割数を多くすることで待ち時間を短縮するスケジューリング手法を提案する。

A Scheme for Producing Efficient Broadcast Schedule by Data Segmentation

TOMOKI YOSHIHISA,[†] MASAHIKO TSUKAMOTO[†] and SHOJIRO NISHIO[†]

Due to the recent popularization of digital broadcasting systems, broadcasting continuous media data such as music or movies has attracted great attention. On general continuous media data broadcasting, since clients have to wait for starting playing data, various schemes to reduce the waiting time have been studied. Most of these schemes reduce the waiting time by dividing the data into several segments and broadcasting precedent segments frequently with a single channel. However, by dividing the data into finer segments and producing an effective broadcasting schedule, the waiting time is reduced. In this paper, we propose a scheduling scheme to reduce the waiting time by segmenting the data.

1. はじめに

地上波デジタル放送¹⁴⁾ やモバイル放送⁹⁾ といった近年のデジタル放送の普及にともない、音楽や映画といった連続メディアデータの放送型配信に対する注目が高まっている。放送型配信では、一般に、サーバはデータを繰り返して放送する。クライアント数が多い場合に有効だが、クライアントは必要なデータが放送されるまで待たなければならない。一方、連続メディアデータ放送では、クライアントがデータを初めから最後まで途切れずに再生できることが重要になる。このため、クライアントがデータを途切れずに再生できることを考慮したうえで、データの再生を開始するまでの待ち時間を短縮する様々な手法が提案されている^{1),2),4),6)~8),10)~13),15),17)}。これらの手法のほとんどは、複数のチャンネルを用いているため、単一のチャンネルを用いて放送する場合に比べて処理が複雑になる。

既存のシステムにおいても、処理の複雑さや実装面積の大きさといった理由から、複数のチャンネルから同時にデータを受信できるように複数のチューナを装備している受信機は多くない。例えば、デジタル放送を受信するには、セットトップボックスと呼ばれる受信機が必要になるが、複数のチャンネルから同時にデータを受信できるセットトップボックスは多くない。このようなことから、複数チャンネルを用いるよりも、単一チャンネルを用いる方が現実的であるといえる。

これに対し、分割放送方式¹⁸⁾ は、データを幾つかの部分に等分割して、単一チャンネルを用いて初めの部分を頻繁に放送することで待ち時間を短縮している。例えば、分割放送方式の一手法であるセグメント挿入法¹⁸⁾ では、5 Mbps の 30 分の動画データを 24 Mbps の帯域幅³⁾ を用いて放送する場合、平均待ち時間は 97 秒になる。しかし、さらなる待ち時間短縮に対するユーザの要求があると考えられる。

本稿では、データを細分割して分割数を多くすることで、既存の手法よりも待ち時間を短縮する手法を提案する。例えば、先程の例の場合、データを 100 個の部分に分割すると平均待ち時間は 67 秒になる。デー

[†] 大阪大学大学院情報科学研究科マルチメディア工学専攻
Department of Multimedia Engineering, Graduate
School of Information Science and Technology, Osaka
University



図 1 単純繰り返し手法
Fig. 1 A simple repetition scheme.

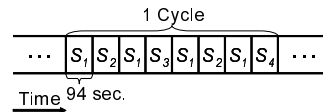


図 3 セグメント挿入法の放送スケジュールの例
Fig. 3 An example of a broadcast schedule under the segment insertion method.

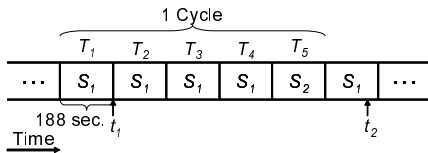


図 2 データを二等分する場合の放送スケジュールの例
Fig. 2 An example of a broadcast schedule for dividing the data into two segments.

データを細分割する場合、オーバーヘッドの増加による待ち時間の増加が考えられるため、待ち時間の短縮とオーバーヘッドの増加がトレードオフとなる。検証を行った結果、実用レベルのヘッダサイズ（1K バイトまで）であれば、オーバーヘッドの増加は待ち時間に大きく影響しないことを確認した。以下、2章で関連研究について説明し、3章で提案手法を説明する。4章で評価を行い、5章で議論を行う。最後に6章で本稿をまとめる。

2. 関連研究

連続メディアデータを分割せずに単一チャネルを用いて繰り返し放送する場合（単純繰り返し手法）を考える。例えば、MPEG2 で符号化された 5 Mbps の 30 分の動画データを、衛星デジタル放送を想定した 24 Mbps の帯域幅を用いて放送する場合、データの放送には 375 秒かかる（図 1）。この場合、ユーザが任意のタイミングでクライアントに受信要求を出すと、データの再生開始まで平均で 188 秒待つことになる。しかし、データを幾つかの部分に等分割して、単一チャネルを用いて初めの部分を頻繁に放送することで平均待ち時間を短縮できる。

義久らは、データを二等分して単一チャネルを用いて放送する場合の平均待ち時間の最小値と最小値を与える放送スケジュールを示している¹⁸⁾。この放送スケジュールは、放送に使用する帯域幅と再生レートから導出される。例えば、先程の例で二等分して放送する場合、平均待ち時間の最小値は図 2 に示す放送スケジュールで与えられる。 N 個の部分に等分割したデータをセグメントと呼び、 S_1, \dots, S_N で示す。ここでは、 S_1 は前半のデータ、 S_2 は後半のデータを示し、 S_1 および S_2 の再生には 15 分かかる。 T_1 内で

受信要求を出したクライアントが、 S_1 を初めて受信する t_1 から再生を開始すると、 S_1 の再生終了時刻は t_2 になる。クライアントは S_1 を再生しながら T_5 で放送される S_2 を受信し、バッファに保存しておくことで、 S_1 の再生終了後続けて S_2 を再生できる。この場合の平均待ち時間は 94 秒になる。同様に T_2, \dots, T_5 内で受信要求を出さず場合の平均待ち時間を算出して平均すると、結局、平均待ち時間は 131 秒になり、単純繰り返し手法に比べて短縮されている。

三等分以上の場合の平均待ち時間の最小値を算出することは、放送スケジュールの組み合わせの数が非常に多く困難なため、セグメント挿入法が提案されている。セグメント挿入法では、クライアントは S_1 の受信と同時に再生を開始しても最後まで途切れずに再生できるという特徴がある。この特徴を保持するために分割数の上限が存在し、任意の分割数を与えられない。平均待ち時間は分割数が上限値の場合に最小になる。例えば、先程の例の場合、分割数の上限は 4 になる。図 3 に $N = 4$ の放送スケジュールを示す。平均待ち時間は 94 秒になり、二等分の最小値に比べて短縮されている。

交替放送法¹⁶⁾は、複数のデータを放送することに焦点をあてたヒューリスティックなスケジューリング手法である。データの分割数によって平均待ち時間が変化するため、分割数の計算式が二種類提案されている。処理重視分割数決定法では、セグメント挿入法と同じく、クライアントは S_1 の受信と同時に再生を開始しても最後まで途切れずに再生できる。ただし、待ち時間重視分割数決定法よりも平均待ち時間が長くなる場合がある。ここでは、一個のデータを放送するとし、先程と同じ例の場合の放送スケジュールを図 4 に示す。‘Mechanism Dominance Method’ が処理重視分割数決定法を示し、‘Waiting Time Dominance Method’ が待ち時間重視分割数決定法を示す。処理重視分割数決定法の平均待ち時間は 125 秒になり、待ち時間重視分割数決定法の平均待ち時間は 97 秒になる。

以上のように、単一のチャネルを用いて放送する場合に平均待ち時間を短縮する幾つかの手法が提案されているが、分割数を多くすることで、さらに平均待ち

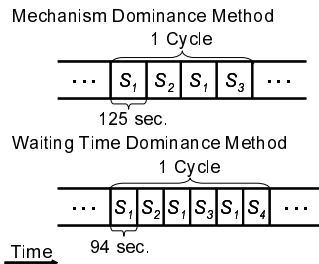


図 4 交替放送法の放送スケジュールの例
Fig. 4 Examples of a broadcast schedule under the alternative broadcasting method.

時間を短縮できる。

3. 提案手法

データを細分割して分割数を多くすることで待ち時間を短縮するスケジューリング手法「飛び越し挿入法」を提案する。待ち時間とは、ユーザがクライアントに連続メディアデータの受信要求を出してから、データの再生を開始されるまでの待ち時間を示す。

3.1 想定環境

ある一定の通信速度以上の帯域幅や送信したデータがクライアントに届くことが保証できるギャランティ型の通信を想定する。IP ネットワークでは、RSVP などの帯域保証プロトコルを用いることで、ギャランティ型の通信網と見なせる。データの送受信には「フレーム」と呼ぶ単位を用いる。フレームの初めの部分にあるヘッダには、そのフレームに含まれるデータの再生開始時刻など、データに関する情報を記述できる。本稿では次のシステム環境を想定する。

- 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ない。
- クライアントがデータの再生を開始すると、最後まで途切れずに再生できる。
- クライアントはバッファを持ち、受信したデータを再生している間も放送されているデータを受信し、バッファに保存できる。バッファには受信する連続メディアデータを保存するために十分な容量がある。
- クライアントは受信要求を出してから放送されているデータを受信し、受信要求を出す前に放送されたデータはバッファに保存しない。
- クライアントはフレームを途中から受信できず、初めからしか受信できない。
- クライアントは受信したデータをすぐに再生できる。
- データの放送にかかる時間は再生時間より短い。

- サーバは 1 つのチャンネルで 1 つの連続メディアデータを放送する。
- サーバはフレームを隙間なく繰り返し放送する。
- サーバはデータ更新を行わない。

複数の連続メディアデータが放送されており、ユーザが受信する連続メディアデータを予測する必要があるため、ユーザが受信要求を出す以前にバッファリングすることは考えない。以上のようなシステムの例として、MPEG2 で符号化された動画データをデジタル放送で配信する場合があげられる。

3.2 スケジューリング手順

- (1) 再生時間が D の連続メディアデータを N 個のセグメント S_1, \dots, S_N に等分割する。各セグメントの再生時間は D/N になる。
- (2) S_1, \dots, S_L を左から順に $\lceil N/L \rceil$ 回並べる。 L は次式で与える。ここで、 B は使用する帯域幅、 R は連続メディアデータの再生レートである。

$$L = \left\lceil \frac{N}{\exp(0.57 \times (\frac{B}{R} - 1))} \right\rceil \quad (1)$$

ただし、 $N \geq 2$ で $L = 1$ となる場合、 $L = 2$ とする。論拠は 5.3 節に示す。

- (3) $i = 2$ とする。
- (4) $j = 1$ とする。
- (5) 左端のセグメントの右側に $S_{(i-1)L+j}$ を挿入する。
- (6) j を一つ増やす。ただし、 $j = L + 1$ となった場合には $j = 1$ とする。
- (7) 挿入したセグメントから (2) で並べたセグメントを右向きへ数えて i 番目のセグメントの右側に $S_{(i-1)L+j}$ を挿入する。最後のセグメント S_N を挿入した時点で放送スケジュールが完成する。
- (8) 右端のセグメントにたどり着くまで (6) (7) を繰り返す。
- (9) i を一つ増やし (4) へ。

サーバは作成された放送スケジュールにしたがって、左側から順に繰り返し放送する。たとえば、MPEG2 で符号化された 5 Mbps の 30 分の動画データ ($D = 30$ 分、 $R = 5\text{Mbps}$) を 100 等分 ($N = 100$) して 24 Mbps の衛星デジタル放送 ($B = 24\text{Mbps}$) で配信する場合のスケジューリング手順を図 5 に示す。灰色の部分は (2) で並べたセグメントを示す。セグメントを挿入する際 (2) で並べたセグメントを飛ばして挿入していくことが手法名の由来である。

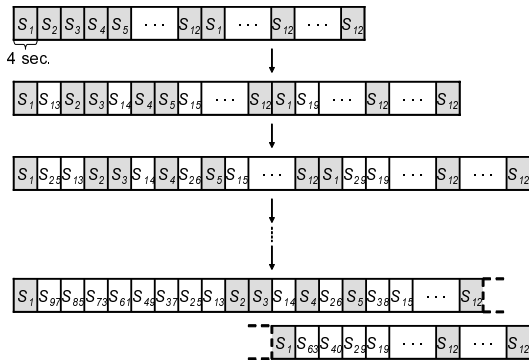


図 5 飛び越し挿入法のスケジューリング手順
Fig. 5 A scheduling process under the hopping insertion method.

3.3 導入方法

連続メディアデータの配信者は、放送するデータを等分割し、作成した放送スケジュールに従ってデータを繰り返し放送する。セグメントを放送する際、そのセグメントが含まれる連続メディアデータの識別子やセグメントの番号がわかるように、これらの情報をヘッダに記述する。

ユーザがクライアントに連続メディアデータの受信要求を出すと、クライアントは、放送されているデータの中から、指定された連続メディアデータの受信を開始する。クライアントはデータの再生中にも放送されているデータを受信し、バッファに保存する。クライアントは S_1 を受信してもすぐに再生を開始せず、最後まで途切れずにデータを再生できるようになるまで待ってから再生を開始する。この待ち時間はヘッダに記述された情報から得ることができる。 S_i ($i = 1, \dots, N - 1$) の再生終了後、放送されているか、もしくはバッファにある S_{i+1} を即座に続けて再生することで、クライアントはデータを最後まで途切れずに再生できる。例えば、5Mbps の 30 分の動画データを 100 等分して 24 Mbps の帯域幅を用いて放送する場合 (図 6) を考える。データの再生レートは帯域幅より小さいため、1 個のセグメントの放送にかかる時間は約 4 秒だが、再生には 18 秒かかる。ユーザが時刻 t_1 で受信要求を出すと、クライアントは t_2 から放送される S_1 を受信できる。クライアントが t_2 から S_1 を再生すると S_1 の再生終了時刻は t_4 となり、 t_4 までに S_2 が放送されていないため、 S_1 の再生終了後続けて S_2 を再生できない。この場合、 t_3 から S_1 を再生すると、 S_1 の再生終了時刻は t_5 となり、 t_5 から放送されている S_2 を受信し再生することで、 S_1 の再生終了後続けて S_2 を再生できる。 t_3 から再生開始

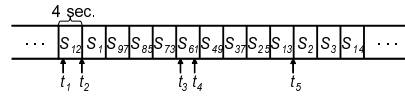


図 6 飛び越し挿入法の導入例
Fig. 6 An example for implementing the hopping insertion method.

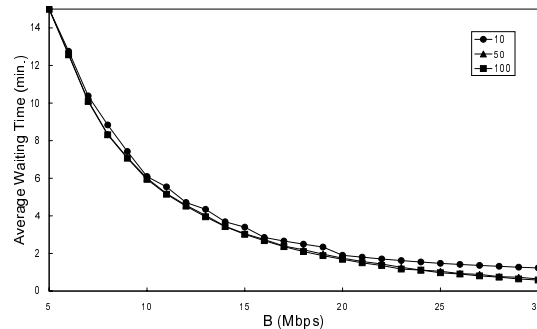


図 7 使用する帯域幅と平均待ち時間
Fig. 7 The available bandwidth and the average waiting time.

した場合には、 S_3, \dots, S_{100} についても同様に、再生開始時刻までに受信開始できる。結局、クライアントは t_3 まで待ってから再生を開始することで、最後まで途切れずに再生できる。

4. 評価

4.1 使用する帯域幅

使用する帯域幅 B をパラメタとして、5Mbps の 30 分の連続メディアデータを放送する場合を想定し、コンピュータでシミュレーションを行い、平均待ち時間を算出した。 $N = 10, 50, 100$ の場合の平均待ち時間を図 7 に示す。横軸は B とし、縦軸は平均待ち時間とした。ヘッダには、データの番号、セグメントの番号、データの受信開始から再生開始までの待ち時間が含まれるとし、各 4 バイトで合計 12 バイトのヘッダサイズとした。このグラフより、 B が大きいほど平均待ち時間が短くなるのがわかる。これは、使用する帯域幅が大きいほどセグメントの放送にかかる時間が短くなるためである。例えば、24 Mbps の衛星デジタル放送を用いて放送する場合、平均待ち時間は $N = 100$ では 67 秒となる。

4.2 分割数

分割数 N をパラメタとして、5Mbps の 30 分のデータを放送する場合の平均待ち時間を算出した。 $B = 10, 20, 24, 30$ Mbps の場合の平均待ち時間を図 8 に示す。横軸は N 、縦軸は平均待ち時間とし、ヘッダサイズは 12 バイトとした。このグラフより、 N に対して

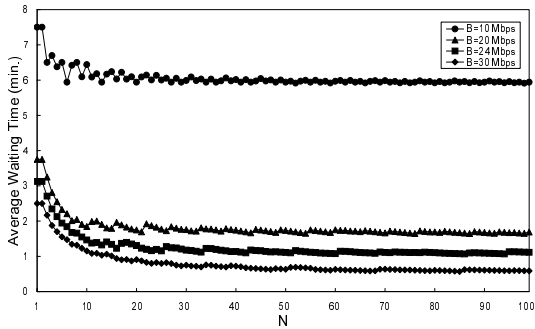


図 8 分割数と平均待ち時間

Fig. 8 The number of segments and the average waiting time.

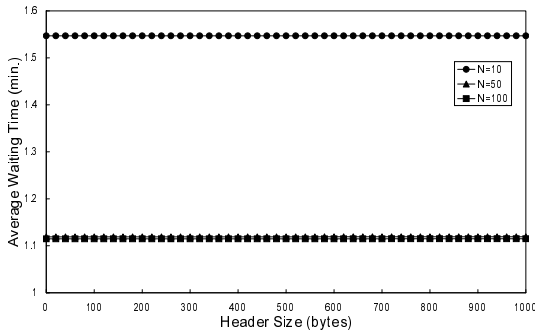


図 9 ヘッダサイズと平均待ち時間

Fig. 9 The header size and the average waiting time.

周期的に平均待ち時間が増加し、 N が大きくなるほど平均待ち時間が収束していることがわかる。これは、 N が大きくなるほどセグメントのデータサイズが小さくなり、放送スケジュールの変化に対する平均待ち時間の変化が小さくなるためである。

4.3 ヘッダサイズの影響

分割数が多いほど平均待ち時間は短くなる傾向にあるが、セグメント毎にヘッダを付加するため、分割数が多いほどヘッダの数が多くなり、オーバーヘッドが増加する。そのため、オーバーヘッドの増加による待ち時間の増加が考えられ、データを細分割する場合には、待ち時間の短縮とオーバーヘッドの増加がトレードオフとなる。そこで、ヘッダサイズをパラメタとして、5Mbps の 30 分のデータを 24Mbps の帯域幅を用いて放送する場合の平均待ち時間を算出し、トレードオフを検証した。 $N = 10, 50, 100$ の場合の平均待ち時間を図 9 に示す。横軸はヘッダサイズとし、縦軸は平均待ち時間とした。

このグラフより、ヘッダサイズが 1K バイト以下であれば、オーバーヘッドは大きくなく、平均待ち時間はほとんど増加しないことが分かる。しかし、ヘッダサ

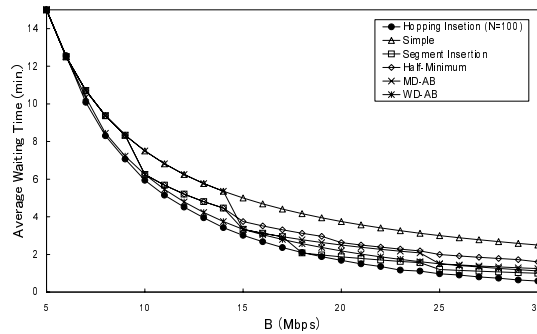


図 10 平均待ち時間の比較

Fig. 10 A comparison of average waiting time.

イズが比較的大きい場合には、分割数が多いほど平均待ち時間が長くなることがある。例えば、ヘッダサイズが 5M バイトの場合、平均待ち時間は $N = 10$ では 98 秒、 $N = 50$ では 117 秒、 $N = 100$ では 241 秒となる。

MPEG1 のシーケンスヘッダは 144 バイト、MPEG2 では 171 バイト⁵⁾といったように、多くの場合、ヘッダサイズは数百バイトであると考えられ、実用レベルのヘッダサイズであれば、平均待ち時間に大きな変化は見られない。

5. 議 論

5.1 平均待ち時間の比較

飛び越し挿入法で 5Mbps の 30 分のデータを放送する場合の平均待ち時間を既存手法と比較した。結果を図 10 に示す。横軸は B 、縦軸は平均待ち時間とし、ヘッダサイズは 12 バイトとした。‘Hopping Insertion ($N=100$)’は $N = 100$ の飛び越し挿入法の平均待ち時間を示し、‘Simple’は単純繰り返し手法、‘Half-Minimum’はデータを二等分して放送する場合の最小値を示す。また、‘Segment Insertion’はセグメント挿入法、‘MD-AB’は処理重視分割数決定法を用いた交替放送法、‘WD-AB’は待ち時間重視分割数決定法を用いた交替放送法の平均待ち時間を示す。セグメント挿入法では、平均待ち時間が最小になるように分割数を決定した。このグラフより、飛び越し挿入法の平均待ち時間がこれらの手法の中で最も短い待ち時間を与えていることがわかる。これは、データを細分割し、分割数を多くすることで、多重化を多く行っているためである。

例えば、24Mbps の帯域幅を用いて放送する場合、既存手法の中で最も短い平均待ち時間を与えるセグメント挿入法の平均待ち時間は 94 秒だが、飛び越し挿入法の平均待ち時間は 67 秒となり、29% 短縮されて

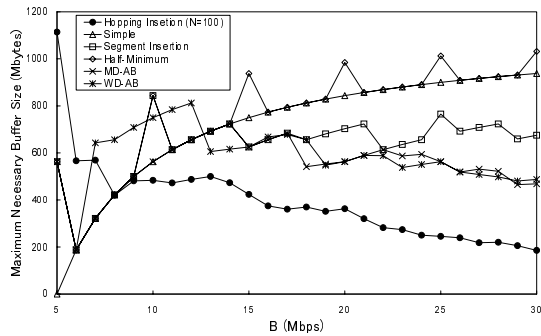


図 11 必要となる最大バッファサイズの比較
Fig. 11 A comparison of maximum necessary buffer size.

いる。

5.2 バッファサイズの比較

5Mbps の 30 分のデータを放送する場合に必要な最大のバッファサイズを比較した。結果を図 11 に示す。横軸は B 、縦軸は平均待ち時間とし、ヘッダサイズは 12 バイトとした。このグラフより、飛び越し挿入法の最大バッファサイズは、使用する帯域幅が比較的大きい場合 ($B \geq 8$ Mbps) に、最も少ないことがわかる。これは、データを細かく分割してバッファに保存できるためである。例えば、 $B = 10$ Mbps の場合、セグメント挿入法では $N = 2$ となる。このため、全データサイズの半分 (563M バイト) を単位としてバッファにデータを保存することになる。しかし、飛び越し挿入法では $N = 100$ であるため、11M バイトずつバッファに保存できる。必要以上に保存するデータのサイズが少なくなるため、必要となる最大のバッファサイズが少なくなる。ただし、使用する帯域幅が比較的小さい場合 ($B \leq 8$ Mbps) には、データを細かく分割しているため、ほとんどすべてのデータを保存することがあり、必要なバッファサイズは多くなる。これは、分割数を小さくすることで少なくできる。

例えば、24Mbps の帯域幅を用いて放送する場合、既存手法の中で最も少ないバッファサイズを与える待ち時間重視分割数決定法を用いた交替放送法では、551M バイトのバッファが必要になるが、飛び越し挿入法 ($N = 100$) では 281M バイトのバッファが必要になる。ただし、5Mbps の帯域幅を用いて放送する場合には、単純繰り返し手法ではヘッダ分の 12 バイトのバッファが必要だが、飛び越し挿入法 ($N = 100$) では 1114M バイトのバッファが必要になる。 $N = 2$ の飛び越し挿入法では、563 M バイトのバッファが必要になる。

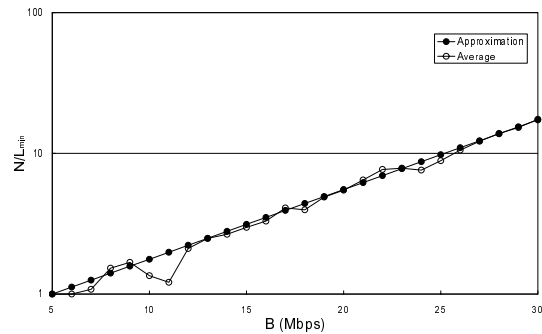


図 12 使用する帯域幅と比例定数 N/L_{min}
Fig. 12 The available bandwidth and the proportionality constant N/L_{min} .

5.3 L の計算式の論拠

L をパラメタとして平均待ち時間を算出し、平均待ち時間の最小値を与える $L (L_{min})$ を求めたところ、 L_{min} は N にほぼ比例することを確認した。さらに、この比例定数 N/L_{min} の平均値の対数値と B が比例することを確認した。この比例定数を最小二乗近似することで、比例定数 0.57 を得た。図 12 に、5Mbps のデータを放送する場合の N/L_{min} の平均値と、傾きが 0.57 の近似値を示す。横軸は B とし、縦軸は N/L_{min} とした。縦軸は対数軸である。‘Average’ が平均値を示し、‘Approximation’ が最小二乗近似の値を示す。比例定数 0.57 を用い、 L が整数であることを考え、次式から式 (1) を導出した。

$$\frac{\ln \frac{N}{L}}{B - 1} = 0.57 \quad (2)$$

5.4 平均待ち時間の周期

図 8 より、平均待ち時間が N に対して周期的に変化していることが分かる。これは、 L を式 (1) で与えているように、除算の結果より小さい最大の整数を L としているためである。よって、 $n (= 2, \dots)$ 回目の周期は分母の指数関数の値を E とすると、

$$[(n+1)E] - [nE] \quad (3)$$

で与えられる。

N が $[nE]$ となる場合に、平均待ち時間はその周期の中で最小になっている。例えば、5Mbps の 30 分のデータを 24Mbps の帯域幅を用いて放送する場合、2 回目の周期は $N = 18$ から 26 で現れている。この周期の平均待ち時間の最小値は $N = 26$ で与えられ、69 秒となる。

5.5 平均待ち時間の近似値

クライアントが S_1 の受信と同時に再生を開始しても最後まで途切れずに再生できると想定し、 S_1 の放送間隔から平均待ち時間の近似値を算出する。ヘッダ

の送信にかかる時間は非常に短いとみなし、無視する。
放送スケジュールで S_1 の間に含まれるセグメントの数は、次のように与えられる。ここで、 $H(n)$ は次式で与えられる調和関数である。

$$H(n) = \sum_{i=1}^n \frac{1}{i} \quad (4)$$

- S_1 の間に $i = \lceil N/L \rceil$ で挿入されたセグメントがある場合、

$$L \cdot H(\lfloor \frac{N}{L} \rfloor) + 1. \quad (5)$$

このような場合は、 $N - L \cdot \lfloor N/L \rfloor$ 回現れる。

- S_1 の間に $i = \lceil N/L \rceil$ で挿入されたセグメントがない場合、

$$L \cdot H(\lfloor \frac{N}{L} \rfloor). \quad (6)$$

このような場合は、 $1 - N + (L + 1) \lfloor N/L \rfloor$ 回現れる。

よって、 S_1 の間に含まれるセグメントの数は、平均すると

$$L \cdot H(\lfloor \frac{N}{L} \rfloor) + \frac{N - L \cdot \lfloor \frac{N}{L} \rfloor}{\lfloor \frac{N}{L} \rfloor + 1} \quad (7)$$

個になる。一個のセグメントの放送にかかる時間は DR/BN であるため、クライアントが S_1 の受信と同時に再生を開始しても最後まで途切れずに再生できる場合の平均待ち時間 W_{ave} は次式で与えられる。

$$W_{ave} = \frac{DR}{2BN} \left(L \cdot H(\lfloor \frac{N}{L} \rfloor) + \frac{N - L \cdot \lfloor \frac{N}{L} \rfloor}{\lfloor \frac{N}{L} \rfloor + 1} \right) \quad (8)$$

5Mbps の 30 分のデータを放送する場合の平均待ち時間のシミュレーション結果と式 (8) で与えられる近似値を図 13 に示す。‘Approximation’ が近似値を示し、‘Simulation’ がシミュレーション結果の平均待ち時間を示す。このグラフより、近似値がシミュレーション結果に近い値を与えていることが分かる。例えば、 $B = 24\text{Mbps}$ 、 $N = 100$ の場合、平均待ち時間のシミュレーション結果は 67 秒、近似値は 62 秒になる。さらに近似を行うことで、平均待ち時間の収束値は

$$\frac{0.285DR}{B \cdot \exp(0.57(\frac{B}{R} - 1))} \left(\frac{B}{R} - 1 \right) \quad (9)$$

と求まる。

筆者らは N が $\lceil 3 \exp(0.57(\frac{B}{R} - 1)) \rceil$ となれば平均待ち時間を十分短縮できると考える。例えば、 $B = 24\text{Mbps}$ の場合、上式は 26 となる。 $N = 26$ の場合、平均待ち時間は 69 秒となる。

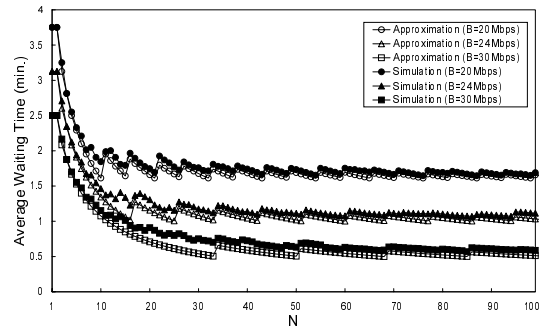


図 13 平均待ち時間の近似値
Fig.13 An approximate average waiting time.

6. 結 論

連続メディアデータを単一のチャネルを用いて放送する場合に平均待ち時間を短縮する幾つかの手法が提案されている。既存の手法では、データを幾つかの部分に等分割しているが、データの分割数を多くすることで、既存の手法よりも待ち時間を短縮できる。本稿では、データを細分割して分割数を多くすることで待ち時間を短縮するスケジューリング手法を提案した。データを細分割する場合、オーバーヘッドの増加による待ち時間の増加が考えられるため、待ち時間の短縮とオーバーヘッドの増加がトレードオフとなる。評価の結果、オーバーヘッドは小さく、既存の手法よりも平均待ち時間を短縮できることを明らかにした。

今後の予定として、早送りを考慮することや、再生中に再生レートが変化する場合のスケジューリング手法の考案が挙げられる。

謝 辞

本研究は、文部科学省 21 世紀 COE プログラム「ネットワーク共生環境を築く情報技術の創出」、文部科学省科学技術振興調整費「モバイル環境向 P2P 型情報共有基盤の確立」、および文部科学省特定領域研究 (C)「Grid 技術を適応した新しい研究手法とデータ管理技術の研究」(課題番号: 13224059) 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) Aggarwal, C.C., Wolf, J.L. and Yu, P.S.: A permutation-based pyramid broadcasting scheme for video-on-demand system, *Proc. IEEE Int. Conf. on Multimedia Computing and Systems (ICMCS '96)*, pp.118-126 (1996).
- 2) Eager, D.L. and Vernon, M.K.: Dynamic skyscraper broadcasts for video-on-demand,

- Proc. 4th Int. Workshop on Multimedia Information Systems (MIS '98)*, pp.18-32 (1998).
- 3) 橋本和彦: デジタル衛星放送の技術と動向, 電子情報通信学会誌, Vol.81, No.1, pp.86-88 (1998).
 - 4) Hua, K.A. and Sheu, S.: Skyscraper broadcasting: a new broadcasting scheme for metropolitan video-on-demand systems, *Proc. ACM SIGCOMM*, pp.89-100 (1997).
 - 5) 藤原洋: 最新 MPEG 教科書, マルチメディア通信研究会, アスキー出版局, 東京 (1997).
 - 6) Janakiraman, R. and Waldvogel, M.: Fuzzy-cast: Efficient Video-on-Demand over Multicast, *Proc. IEEE Infocom* (2002).
 - 7) 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).
 - 8) 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).
 - 9) モバイル放送株式会社: モバイル放送 (2003). <http://www.mbc.co.jp/>
 - 10) Paris, J.-F.: A Simple Low-Bandwidth Broadcasting Protocol for Video-on-Demand, *Proc. Int. Conf. on Computer Communications and Networks (IC3N '99)*, pp.118-123 (1999).
 - 11) Paris, J.-F.: An Interactive Broadcasting Protocol for Video-on-Demand, *Proc. IEEE Int. Performance, Computing, and Communications Conference (IPCCC '01)*, pp.347-353 (2001).
 - 12) Paris, J.-F., Carter, S.W. and Long, D.D.E.: A hybrid broadcasting protocol for video on demand, *Proc. Multimedia Computing and Networking Conference (MMCN '99)*, pp.317-326 (1999).
 - 13) Paris, J.-F., Carter, S.W. and Long, D.D.E.: Efficient Broadcasting Protocols for Video on Demand, *Proc. Int. Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS '98)*, pp.127-132 (1998).
 - 14) 総務省: 情報通信白書平成 15 年版 (2003). <http://www.johotsusintokei.soumu.go.jp/whitepaper/ja/h15/pdf/index.html>
 - 15) 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).
 - 16) 義久智樹, 塚本昌彦, 西尾章治郎: 連続メディアデータ放送における単一チャネルを用いた複数データのスケジューリング手法, 情報処理学会シンポジウムシリーズ マルチメディア, 分散, 協調とモバイルシンポジウム (DICOMO2002) 論文集, Vol.2002, No.9, pp.511-514 (2002).
 - 17) 義久智樹, 塚本昌彦, 西尾章治郎: データの分割に関する制約を考慮した連続メディアデータ放送におけるスケジューリング手法, 情報処理学会論文誌: データベース, Vol. 44, No. SIG3 (TOD17), pp.33-42 (2003).
 - 18) 義久智樹, 塚本昌彦, 西尾章治郎: 連続メディアデータ放送における待ち時間短縮のための分割放送方式, 情報処理学会論文誌, Vol. 44, No. 6, pp.1558-1569 (2003).