

# 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

義久 智樹<sup>†</sup> 塚本 昌彦<sup>†</sup> 西尾 章治郎<sup>†</sup>

近年、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関する研究がさかんに行われている。連続メディアデータ配信では、クライアントがデータを最後まで途切れずに再生できることが重要になる。従来研究では、この条件を考慮したうえで、クライアントの待ち時間を短縮する手法を提案している。これらの手法を用いる場合、クライアントは複数の放送チャンネルから同時にデータを受信しなければならない。しかし、衛星デジタル放送や bluetooth などの放送システムのデータ受信機は、ハードウェアレベルでは複数のチャンネルから同時にデータを受信できない場合が少なくない。そこで本稿では、データの放送に要する時間が再生時間より短くなることに注目し、単一チャンネルを用いてクライアントの待ち時間を短縮するスケジューリング方式の提案および評価を行う。

## A Scheduling Scheme for Waiting Time Reduction on Continuous Media Data Broadcast

TOMOKI YOSHIHISA,<sup>†</sup> MASAHIKO TSUKAMOTO<sup>†</sup> and SHOJIRO NISHIO<sup>†</sup>

Due to the recent proliferation of continuous media applications such as audio and video data, various strategies for broadcasting continuous media data have been studied. Some of them have focused on reducing the waiting time of clients under the continuity condition, i.e., to play data without any intermittence until the end of the data. These strategies usually employ multiple channels to broadcast continuous media data. However, most receivers of broadcast systems such as satellite systems and bluetooth systems cannot receive data from multiple channels concurrently in the hardware level. In this paper, we propose a scheduling scheme to reduce the waiting time of clients with a single channel and evaluate our scheme.

### 1. はじめに

近年、計算機の高速度化やネットワークの広帯域化にともない、ストリーミング配信が普及している<sup>14)</sup>。ストリーミング配信を用いると、クライアントは音声や画像データファイルのダウンロード完了を待たずに、データの受信開始とほぼ同時にそれらを再生できる。このようなストリーミングデータを放送型配信システムで配信する研究がさかんに行われている<sup>1)~12)</sup>。

一般に、放送型配信では、サーバがデータを周期的に繰り返して放送し、クライアントは必要なデータのみ選択して取得する。サーバは同じデータを各クライアントにまとめて配信できるため、クライアント数が増加してもシステムの負荷がほとんど変わらないという特徴がある。しかし、放送型配信システムでは、

ユーザがクライアントにデータの受信要求を出してから、クライアントは目的のデータが放送されるまで待たなければならない。

一方、サーバが放送型配信を用いて、音楽や映画といった連続メディアデータの配信を行う場合には、クライアントがデータを最後まで途切れずに再生できることが重要になる。連続メディアデータの放送型配信に関する研究では、この条件を満たしながら、複数の放送チャンネルを用いたサーバのスケジューリング戦略によって、クライアントの待ち時間を短縮している<sup>1)~8)</sup>。複数チャンネルを用いてデータを配信すると、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し、サーバは複数のチャンネルを管理する必要がある。したがって、単一チャンネルの場合と比べて処理が複雑になる。システム構築の観点から考えると、ハードウェアレベルでは無線 LAN システムは複数のチャンネルから同時にデータを受信できない。場所に特化したコンテンツを放送できる bluetooth でも、多くの受信モジュールでは複数のチャンネルから同時にデータを

<sup>†</sup> 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

受信できない。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デジタル放送受信機器が必要になるが、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。

そこで、本稿では、単一チャンネルを用いてクライアントの待ち時間を短縮できるスケジューリング方式を提案する。提案方式では、連続メディアデータの放送に要する時間が再生時間より短くなることに注目し、データをいくつか等に分割して、データの初めの部分を頻繁に放送することでクライアントの待ち時間を短縮する。この方式では、データの分割数および分割したデータのスケジューリングが待ち時間に影響を与える。本稿では、2等分する場合については数学的な解析を行い、平均待ち時間を最短にする放送スケジュールを導出する。3等分以上の場合については、ヒューリスティックな手法を提案し、評価を行う。

以下、2章では、基本となる考え方を例を用いて説明する。これより、データをいくつか等に分割して、データの初めの部分を頻繁に放送することでクライアントの待ち時間を短縮できることが分かる。3章では想定環境および提案方式について述べる。4章では、データを2等分する場合の解析を行い、データを2等分して放送するスケジュールの中で、平均待ち時間を最短にする放送スケジュールを導出する。5章では、データを3個以上の部分に等分割する場合について、ヒューリスティックなスケジューリング手法を提案し、その評価を行う。6章で関連研究について述べ、最後に7章で本稿のまとめを行う。

## 2. 基本となる考え方

本章では、初めに連続メディアデータを単純に繰り返して放送する場合の平均待ち時間を求め、次に提案方式の基本となる考え方を簡単な例を用いて説明する。

再生時間5分の音楽データの放送に40秒かかる場合を考える。これはMP3規格で符号化した5Mバイトの5分の音楽データを1Mbpsの衛星デジタル放送やbluetoothで配信する場合に相当する。このデータを単純に繰り返して放送する場合(単純繰返し手法)、サーバは図1に示すように40秒ごとに同じデータを放送する。ユーザが受信要求をクライアントにランダムに出すと、ユーザは受信要求を出してから曲の再生が開始されるまで平均して20秒待つことになる。ただし、ここでは受信しながら再生するストリーム型の再生を仮定する。

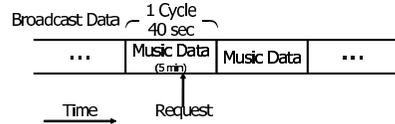


図1 連続メディアデータ放送のための単純繰返し手法  
Fig. 1 A simple repetition scheme for continuous media data broadcast.

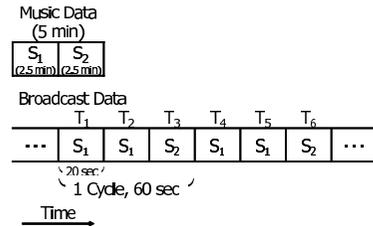


図2 基本となる考え方を説明するための簡単な例  
Fig. 2 A simple example for explaining our basic idea.

提案方式の基本となる考え方を説明するために、簡単な例を示す。この音楽データを再生時間2.5分の2つのデータに等分割する。前半を $S_1$ 、後半を $S_2$ とし、初めの部分を頻繁に放送するように、 $S_1 S_1 S_2$ の順に繰り返し放送する(図2)。 $S_1, S_2$ の放送にはそれぞれ20秒かかり、放送周期は60秒になる。ある放送周期の、最初の $S_1$ の放送区間を $T_1$ 、次の $S_1$ の放送区間を $T_2$ 、最後の $S_2$ の放送区間を $T_3$ と順に呼ぶ。ここで、平均待ち時間を以下のように場合分けして求める。

- $T_1$  内でユーザが受信要求を出す場合  
クライアントは $T_2$ で放送される $S_1$ を受信し、 $S_1$ の再生中に $T_3$ で放送される $S_2$ を受信できる。すなわち、 $T_2$ の開始と同時に再生を開始し、データを最後まで途切れずに再生できる。この場合の平均待ち時間は10秒になる。
- $T_2$  内でユーザが受信要求を出す場合  
クライアントは $T_4$ で放送される $S_1$ を待つ間に、 $T_3$ で放送される $S_2$ を受信できる。この場合の平均待ち時間は30秒になる。
- $T_3$  内でユーザが受信要求を出す場合  
クライアントは $T_4$ の開始と同時に再生できる。この場合の平均待ち時間は10秒になる。

結局、平均待ち時間は約17秒となり、前述の単純繰返し手法より約15%短縮できる。さらにこの場合、図3に示すように、 $S_1 S_1 S_1 S_1 S_1 S_1 S_1 S_2$ の順に繰り返し放送すれば、平均待ち時間は最短の12.5秒となり、単純繰返し手法より平均待ち時間を約38%短縮できる。2等分の例を示したが、3等分以上の場合も初めの部分を頻繁に放送することで、平均待ち時間を短

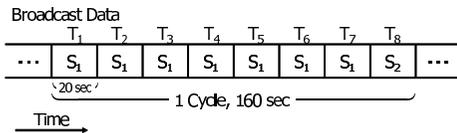


図 3 平均待ち時間を最短にする放送スケジュール (2 等分, 再生時間/放送時間 = 7.5)

Fig. 3 The broadcast schedule which gives the minimum average waiting time (2 division, playback ratio is 7.5).

縮できる。

### 3. 想定環境および提案方式

連続メディアデータの再生時間を  $D$ , そのデータの放送に要する時間を「放送時間」と呼び  $D'$  で表す。本稿ではデータの放送時間が再生時間より短くなることを考慮し, 再生倍率  $a$  を

$$a = \frac{D}{D'} \quad (1)$$

と定義する。たとえば, MPEG Audio Layer 3 (MP3) 規格を使用して音声ファイルを符号化する場合, 5 分の音声データを, 約 5 M バイトのファイルにできる。このデータを 1 Mbps の帯域を用いて放送すると, 40 秒で放送できる。この場合, 再生倍率  $a = 300/40 = 7.5$  になる。音声だけでなく, 動画の放送時間も再生時間より短くできる。MPEG1 規格は 1~1.5 Mbps 程度の通信速度に対応しており, 10 Mbps の帯域を持つ通信網で配信すると, 再生倍率は 6.7~10.0 となる。MPEG2 規格は 4~15 Mbps のデータ通信速度に対応しており, 100 Mbps の帯域を持つ通信網で配信すると, 再生倍率は 6.7~25 となる。さらに, MPEG4 規格は 384 kbps 以下の通信速度にも対応しており, 1 Mbps の衛星デジタル放送や bluetooth で配信すると, 再生倍率は 2.6 以上となる<sup>13)</sup>。

再生倍率は, データの再生レートに依存するが, データの再生レートはサーバが連続メディアデータを符号化する際に知ることができるため,  $a$  の値を算出できる。

#### 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衛星デジタル放送や bluetooth といった, ギャランティ型の通信網を想定する。IP ネットワークにおいても 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。フレームには, そのフレームに含まれるデータのサイズなど, データに関する情

報を付加できる。帯域幅が保証された通信網を想定するため, 通信品質の劣化によるフレーム損失はないとして議論をすすめる。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なければならない。
- 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。
- 放送時間は再生時間より短い ( $D' \leq D$ )。そのため,  $a \geq 1$  である。ギャランティ型の通信網であるため,  $a$  の値は一定。
- 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。

サーバは放送する連続メディアデータの数と同じ数のチャンネルを使用することになる。複数のデータを放送する場合, サーバは複数のチャンネルを使用することになるが, 1 つのチャンネルで 1 つの連続メディアデータが放送されているため, 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する必要がない。クライアントが受信する連続メディアデータを変更するには, 受信するチャンネルを変更するだけでよい。このように, 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。また, 待ち時間短縮のために, フレームに含まれるデータの再生に必要な情報をフレームの初めに付加するため, フレームは途中から受信できないと想定する。以上のようなシステムの例として, 衛星デジタル放送や bluetooth で MPEG4 規格で符号化した動画を配信する場合があげられる。この場合, 受信機は複数のチャンネルから同時にデータを受信できず, 再生倍率は 1 以上となる。

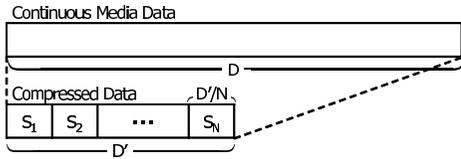


図 4 データの分割

Fig. 4 A division of a data.

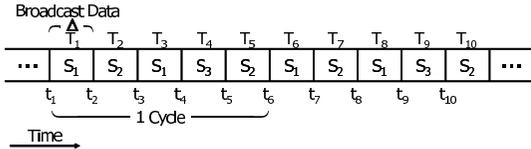


図 5 変数の例 ( $c = 5, N = 3$ )

Fig. 5 An example of variables ( $c = 5, N = 3$ ).

3.2 分割放送方式

単一チャネルを用いてクライアントの待ち時間を短縮する「分割放送方式」を提案する。ここで、待ち時間とは、データを最後まで途切れずに再生できることを満たしたうえで、ユーザがデータの受信要求を出してから、データの再生が開始されるまでの待ち時間を意味する。

分割放送方式では、連続メディアデータを放送時間の等しい  $N$  個の部分に分割し、初めの部分を頻繁に放送するスケジュールを作成する。分割したデータをセグメント  $S_i$  ( $i = 1, \dots, N$ ) と呼ぶ。1つのセグメントの放送時間は  $D'/N$  となる(図4)。1つのセグメントを1つのフレームで放送し、クライアントはフレームを受信し始めると、そのセグメントの番号を知ることができる。たとえば、セグメント  $S_i$  の番号  $i$  をフレームの初めに付加し、すぐに  $i$  の値をアプリケーションに通知することで実現できる。サーバは1周期中に  $c$  個のセグメントを放送するが、1周期中に、同じセグメントを数回放送する場合がある。図5は  $c = 5, N = 3$  の放送スケジュールの一例である。

ここで、セグメントをどのような順序でスケジューリングすれば、クライアントの待ち時間を最短にできるかが問題となる。たとえば、5分のMP3データ ( $D = 5$ ) を1分で放送できる ( $a = 5$ ) 場合、連続メディアデータを2等分し ( $N = 2$ ), 1周期中に4個または5個のセグメントを放送する ( $c = 4, 5$ ) と、表1に示す放送スケジュールが考えられる。 $S_1S_2S_1S_2$  は  $c = 2$  と見なせるため、表中に示していない。表1で最短の平均待ち時間は21秒で、放送スケジュールは  $S_1S_1S_1S_1S_2$  となる。次に、3等分し、1周期中に4個のセグメントを放送すると、表2に示す9通りの放送スケジュールが考えられ、最短の平均待ち時間は

表 1 1周期中に4個または5個のセグメントを放送するスケジュール ( $a = 5, D = 5$  分,  $N = 2$ )

Table 1 Broadcast schedules which include 4 or 5 segments in one broadcast cycle ( $a = 5, D = 5$  min,  $N = 2$ ).

c = 4		c = 5	
放送スケジュール	平均待ち時間 (秒)	放送スケジュール	平均待ち時間 (秒)
$S_1S_1S_1S_2$	23	$S_1S_1S_1S_1S_2$	21
$S_1S_1S_2S_2$	38	$S_1S_1S_1S_2S_2$	33
$S_1S_2S_2S_2$	60	$S_1S_1S_2S_1S_2$	27
		$S_1S_1S_2S_2S_2$	51
		$S_1S_2S_1S_2S_2$	33
		$S_1S_2S_2S_2S_2$	75

表 2 1周期中に4個のセグメントを放送するスケジュール ( $a = 5, D = 5$  分,  $N = 3$ )

Table 2 Broadcast schedules which include 4 segments in one broadcast cycle ( $a = 5, D = 5$  min,  $N = 3$ ).

放送スケジュール	平均待ち時間 (秒)
$S_1S_1S_2S_3$	25
$S_1S_1S_3S_2$	25
$S_1S_2S_1S_3$	19
$S_1S_2S_2S_3$	40
$S_1S_2S_3S_2$	40
$S_1S_3S_2S_2$	40
$S_1S_2S_3S_3$	40
$S_1S_3S_3S_2$	40
$S_1S_3S_2S_3$	40

19秒で、放送スケジュールは  $S_1S_2S_1S_3$  となる。このように、 $a, c, N$  の値を決めると、平均待ち時間を計算によって求めることができるが、これらの値は任意に選択でき、セグメントのスケジューリングが無数にあるため、最短の平均待ち時間を与える放送スケジュールを総当たりで求めることは現実的でない。そこで、4章で2等分の場合の解析を行い、平均待ち時間を最短にする放送スケジュールを導出する。3等分以上の場合については、セグメントのスケジューリングの組合せが多く解析が困難なため、5章でヒューリスティックなスケジューリング手法を提案する。

4. 2等分の場合

本章では、分割放送方式において、連続メディアデータを2等分する場合の解析を行い、データを2等分して放送するあらゆるスケジュールの中で、平均待ち時間を最短にする放送スケジュールを導出する。

4.1 平均待ち時間の分類

フレームの放送時間を  $\Delta$  で表す。フレームに付加されたセグメントの番号や再生開始時刻などの情報のデータサイズは小さいとし、すべてのフレームの放送時間は等しいとする ( $\Delta = D'/2$ )。1つのフレームが放送されている時間をタイムスロットと呼び、 $T_i$

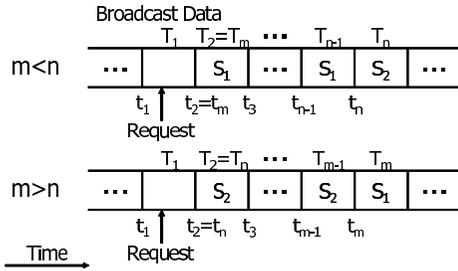


図 6  $m, n$  の値の例

Fig. 6 An example of the value of  $m$  and the value of  $n$ .

( $i = 1, 2, \dots$ ) で表す.  $T_i$  の開始時刻は  $t_i$  で表す.  $t_i = t_1 + (i - 1)\Delta$  となる. 図 5 にこれらの例を示す.

あるタイムスロットでユーザが受信要求を出した場合の平均待ち時間を求めるために、受信要求を出した時刻が含まれるタイムスロットを  $T_1$  とし、その後、 $S_1$  が初めて放送されるタイムスロットを  $T_m$ 、 $S_2$  が初めて放送されるタイムスロットを  $T_n$  とする. 図 6 に  $m < n$  と  $m > n$  の場合の放送スケジュールを示す. 受信要求を出した時刻に放送されているセグメントは初めから受信できず、それが  $S_1, S_2$  のどちらであるかは待ち時間に影響しないため、図 6 には示していない. セグメントは  $S_1$  と  $S_2$  の 2 個しかないため、 $m, n$  のどちらか一方は 2 となり、もう一方は 3 以上となる. ここで、時刻  $t_r$  でデータの受信要求を受信したクライアントが、 $S_1$  を受信するとすぐにデータの再生を開始する場合を考える. クライアントが  $S_1$  を再生し終わる時刻を  $\tau$  とする.  $\tau = t_m + a\Delta$  となる.  $n \leq m + a$  の場合、 $t_n \leq \tau$  となり、クライアントは  $S_1$  を再生している間に  $S_2$  を受信できる.  $n > m + a$  の場合、 $t_n > \tau$  となり、 $S_1$  を再生している間に  $S_2$  を受信できない. 以下に、それぞれの場合の平均待ち時間を求める.

(1)  $n \leq m + a$  の場合

クライアントは  $S_1$  を受信し始めるとすぐに再生を開始し、最後まで途切れずに再生できる. この場合の平均待ち時間  $W^I(m)$  は、

$$W^I(m) = \frac{\Delta}{2} + (m - 2)\Delta \quad (2)$$

となる.

(2)  $n > m + a$  の場合

クライアントは  $t_m$  から再生を始めると、 $S_1$  の再生終了後、続けて  $S_2$  を再生できない. そのため  $S_1$  の再生開始時刻を遅らせる必要がある. そこで、時刻  $t_w$  から  $S_1$  を再生する場合を考える.  $S_1$  の再生終了時刻  $\tau_w$  は、 $t_w + a\Delta$  となる. 時刻  $\tau_w$  までに  $S_2$  が放送されるか、

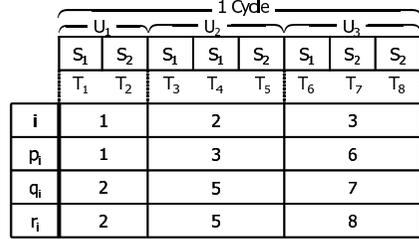


図 7 セグメント列の例

Fig. 7 An example of segment lines.

$\tau_w$  から  $S_2$  が放送されると、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ため、 $t_n \leq \tau_w$  すなわち  $w \geq n - a$  となればよい. 待ち時間は  $w = n - a$  の場合に最短となり、平均待ち時間  $W^{II}(n)$  は、

$$W^{II}(n) = \frac{\Delta}{2} + (n - a - 2)\Delta \quad (3)$$

となる. このとき、最後まで途切れずに再生するために必要な待ち時間に関する情報 ( $t_w - t_m$ ) を  $S_1$  を放送するフレームの初めに付加することで、クライアントはスケジュールを知らなくてもこの待ち時間を知ることができる. この付加情報の受信にかかる時間は非常に短いと考え、議論を行ううえでは無視している.

平均待ち時間を最短にする放送スケジュールを求めるために、分類した平均待ち時間の大小関係を調べる.  $\alpha$  を  $a$  の整数部分、すなわち  $\alpha = [a]$  とすると、 $\alpha \leq a < \alpha + 1$  より

$$\begin{aligned} W^I(2) < W^{II}(\alpha + 3) &\leq W^I(3) \\ &< W^{II}(\alpha + 4) &\leq W^I(4) \\ &< W^{II}(\alpha + 5) &\leq W^I(5) < \dots \end{aligned} \quad (4)$$

が成立する. この式は後に 4.3 節で用いる.

4.2 セグメント列

データを 2 等分する場合の放送スケジュールは一般に  $S_1$  と  $S_2$  の繰返して構成されるため、パターン  $S_2S_1$  が出現する箇所、 $S_2$  と  $S_1$  の間でセグメント列に分ける. すなわち、各セグメント列  $U_i = S_1 \dots S_1 S_2 \dots S_2$  ( $i = 1, \dots, l$ ) とし、放送スケジュールを  $U_1 \dots U_l$  と表す.  $U_i$  で、初めの  $S_1$  が放送されるタイムスロットを  $T_{p_i}$ 、初めの  $S_2$  が放送されるタイムスロットを  $T_{q_i}$ 、最後の  $S_2$  が放送されるタイムスロットを  $T_{r_i}$  とする. ここで、 $i = 1, \dots, l - 1$  に対し、 $r_i + 1 = p_{i+1}$  が成立することになる (図 7). 放送スケジュールをセグメント列に分ける例を表 3 に示す. この表のように、すべての放送スケジュールはいくつかのセグメント列に分けることができる. セグ

表 4 セグメント列に含まれる  $W^I(m), W^{II}(n)$  の数  
Table 4 The number of  $W^I(m)$  or  $W^{II}(n)$  in a segment line.

$U$	$W^I(2)$	$< W^{II}(\alpha+3)$	$\leq W^I(3)$	$< W^{II}(\alpha+4)$	$\leq W^I(4)$	$< \dots$
$S_1 S_2$	1	0	1	0	0	
$S_1 S_2 S_2$	1	0	1	0	1	
$S_1 S_1 S_2$	2	0	1	0	0	
(a) $\overbrace{S_1 \dots S_1}^{\alpha} S_2$	$\alpha$	0	1	0	0	
(b) $\overbrace{S_1 \dots S_1 S_1}^{\alpha+1} S_2$	$\alpha$	1	1	0	0	
$\overbrace{S_1 \dots S_1 S_1 S_1}^{\alpha+2} S_2$	$\alpha$	1	1	1	0	

表 3 放送スケジュールをセグメント列に分ける例  
Table 3 Examples of dividing broadcast schedules into segment lines.

放送スケジュール	セグメント列
$S_1 S_2$	$U_1 = \{S_1 S_2\}$
$S_1 S_1 S_2$	$U_1 = \{S_1 S_1 S_2\}$
$S_1 S_2 S_2$	$U_1 = \{S_1 S_2 S_2\}$
$S_1 S_1 S_1 S_2$	$U_1 = \{S_1 S_1 S_1 S_2\}$
$S_1 S_1 S_1 S_2 S_2 S_2$	$U_1 = \{S_1 S_1 S_1 S_2 S_2 S_2\}$
$S_1 S_2 S_1 S_1 S_2$	$U_1 = \{S_1 S_2\}, U_2 = \{S_1 S_1 S_2\}$
$S_1 S_2 S_2 S_1 S_2$	$U_1 = \{S_1 S_2 S_2\}, U_2 = \{S_1 S_2\}$
$S_1 S_2 S_1 S_1 S_1 S_2$	$U_1 = \{S_1 S_2\}, U_2 = \{S_1 S_1 S_1 S_2\}$
$S_1 S_2 S_2 S_1 S_1 S_2$	$U_1 = \{S_1 S_2 S_2\}, U_2 = \{S_1 S_1 S_2\}$
$S_1 S_2 S_1 S_2 S_1 S_1 S_2$	$U_1 = \{S_1 S_2\}, U_2 = \{S_1 S_2\}$ $U_3 = \{S_1 S_1 S_2\}$

メント列の初めのセグメントが  $S_1$  であり、最後のセグメントが  $S_2$  であるため、放送スケジュールのセグメント列への分け方は一意に定まる。

セグメント列に関して、次の定理が成り立つ。

定理 1 セグメント列  $U_i$  は  $T_{p_i-1}, \dots, T_{r_i-1}$  内で受信要求を出すユーザの平均待ち時間のみ決定する。

たとえば、 $T_{p_i-1}$  内でユーザが受信要求を出す場合、クライアントは  $T_{p_i}$  で放送される  $S_1$  を受信し、 $T_{q_i}$  で放送される  $S_2$  を受信する。この場合の平均待ち時間は  $U_i$  の  $p_i$  や  $q_i$  の値によって決まる。さらに、 $T_{r_i-1}$  内でユーザが受信要求を出す場合、クライアントは  $S_1$  が放送されるのを待つ間に、 $T_{r_i}$  で放送される  $S_2$  を受信し、 $T_{p_{i+1}}$  で次のセグメント列に含まれる  $S_1$  を受信する。この場合、クライアントの待ち時間は  $U_i$  の  $r_i$  の値によって決まる。

以上の理由から、放送スケジュールを構成する各々のセグメント列は平均待ち時間に関して独立しており、もとの放送スケジュールの平均待ち時間は、各々のセグメント列を放送スケジュールとして放送する場合の平均待ち時間の平均で与えられる。たとえば、図 7 において  $a = 2$  の場合、 $U_1$  を放送スケジュールとした場

合の平均待ち時間は  $\Delta$ 、 $U_2$  の平均待ち時間は  $5\Delta/6$ 、 $U_3$  の平均待ち時間は  $3\Delta/2$  となる。これらのセグメント列に含まれるセグメントの数を考慮して平均すると、 $9\Delta/8$  となり、図 7 の平均待ち時間が求まる。さらに、あらゆるセグメント列において、平均待ち時間を最短にするセグメント列を  $U_{min}$  とすると、次の定理が成り立つ。

定理 2 平均待ち時間を最短にする放送スケジュールは  $U_{min}$  を繰り返す放送スケジュールである。

証明を以下に述べる。放送スケジュールをセグメント列  $U_1 \dots U_l$  で構成する。 $U_i \neq U_{min}$  ( $i = 1, \dots, l$ ) とする。 $U_i$  の平均待ち時間は  $U_{min}$  より長いことから、これらのセグメント列のどれか 1 つを  $U_{min}$  に変更すると、定理 1 よりセグメント列は平均待ち時間に関して独立しているため、その放送スケジュールの平均待ち時間は短くなる。同様の議論を繰り返すことで、平均待ち時間を最短にする放送スケジュールは  $U_{min}$  を繰り返す放送スケジュールであることが分かる。したがって、 $U_{min}$  を求めることで平均待ち時間を最短にする放送スケジュールを決定できる。

### 4.3 $U_{min}$ の決定

あるセグメント列  $U$  の平均待ち時間を考える。平均待ち時間の分類から、表 4 のように  $W^I(m), W^{II}(n)$  の個数が決定される。 $U$  はセグメント列であるため、 $S_1 \dots S_1 S_2 \dots S_2$  のように構成される。初めに、 $U = S_1 S_2$  の場合を考える。これは表 4 の 1 行目に示されている。この場合、 $W^I(2)$  の数は 1 個であり、 $W^I(3)$  の数も 1 個である。 $U$  に含まれる  $S_2$  の数を増やすと、 $U$  は 2 行目のようになる。 $S_2$  が増えたことによって、 $W^I(4)$  の数が 1 となる。式 (4) より、 $W^I(3) < W^I(4)$  であるため、 $U$  の平均待ち時間は長くなる。 $S_2$  の数が増えるほど平均待ち時間が長くなるため、 $U$  に含まれる  $S_2$  の数が 1 個の場合が平均待ち時間をより短くする。次に、 $U$  に含まれる  $S_1$  の数を

増やす。\$U\$ は 3 行目のようになる。この場合、\$W^I(2)\$ の数が 2 となる。式 (4) に示すように、\$W^I(2)\$ は最も短いため、\$U\$ の平均待ち時間も短くなる。\$S\_1\$ の数が \$\alpha\$ より小さい限り、\$S\_1\$ の数を増やすほど平均待ち時間は短くなる。\$S\_1\$ の数が \$\alpha\$ の場合、\$U\$ は表 4 に示す (a) となる。\$S\_1\$ の数をさらに増やすことで、\$W^{II}(\alpha+3)\$ の数が 1 となり、表に示す (b) になる (a) の平均待ち時間が \$W^{II}(\alpha+3)\$ より小さい場合、(a) の平均待ち時間は (b) より短くなり、そうでない場合には、(b) の待ち時間が (a) より短くなる。さらに (b) に含まれる \$S\_1\$ の数を増やすと、\$W^{II}(\alpha+4)\$ の数が 1 となる。式 (4) から \$W^I(\alpha+3) < W^I(\alpha+4)\$ であるため、\$U\$ の平均待ち時間は長くなる。すなわち、最短の平均待ち時間を与える放送スケジュール (\$U\_{min}\$) は表 4 の (a) または (b) となる。(a) の平均待ち時間 \$W\_a\$ は

$$W_a = \frac{\alpha + 3}{2\alpha + 2} \Delta \quad (5)$$

と求め、(b) の平均待ち時間 \$W\_b\$ は

$$W_b = \frac{3\alpha - 2\alpha + 6}{2\alpha + 4} \Delta \quad (6)$$

と求まる。式 (5)、(6) から (a) と (b) の平均待ち時間の大小関係を調べると、

$$W_a - W_b = \frac{(\alpha + 1)(\alpha - \alpha) - \alpha}{(\alpha + 1)(\alpha + 2)} \Delta \quad (7)$$

となる。式 (7) の分子を \$A\$ とする。すなわち、

$$A = (a \text{ より大きい最小の整数}) \\ \times (a \text{ の小数部分}) - (a \text{ の整数部分})$$

とすると、平均待ち時間を最短にする放送スケジュールは次のように求まる。

- \$A < 0\$ の場合

平均待ち時間を最短にする放送スケジュールは

$$\overbrace{S_1 \cdots S_1}^{\alpha} S_2 \quad (8)$$

で与えられ、平均待ち時間は式 (5) で与えられる。

- \$A > 0\$ の場合

平均待ち時間を最短にする放送スケジュールは

$$\overbrace{S_1 S_1 \cdots S_1}^{\alpha+1} S_2 \quad (9)$$

で与えられ、平均待ち時間は式 (6) で与えられる。

- \$A = 0\$ の場合

上記 2 つの放送スケジュールの平均待ち時間は等しく、平均待ち時間を最短にする。

以上で、クライアントがデータを最後まで途切れずに再生できることを満たしたうえで、データを 2 等分して放送する場合に、平均待ち時間を最短にする放送

表 5 平均待ち時間が最短の放送スケジュールを求めるまでの計算時間 (秒)

Table 5 Computation times for finding the broadcast schedule which gives the minimum average waiting time (min).

\$N\$	\$c=7\$	\$c=8\$	\$c=9\$	\$c=10\$	\$c=11\$
3	1	2	8	32	114
4	2	17	93	531	2817
5	6	56	503	3691	25753

スケジュールが導出された。

### 5. 3 等分以上の場合

データを 3 個以上の部分に等分割する場合、解析が困難になるため、プログラムによってあらゆる放送スケジュールを考え、平均待ち時間を最短にする放送スケジュールを求めることが考えられる。そこで、データを 3 個以上の部分に等分割して放送する場合に、1 周期中に \$c\$ 個のセグメントを放送するスケジュールを考え、平均待ち時間の最短値を算出するプログラムを作成した。このプログラムを用いて、再生倍率 7.5 の平均待ち時間の最短値を計算した。その計算時間を表 5 に示す (CPU: Pentium III 1.2 GHz, メモリ: 512 M バイト)。計算時間が \$O(N^c)\$ であるため、表 5 の結果から概算すると、4 等分して 1 周期中に 24 個のセグメントを放送するあらゆるスケジュールの中で平均待ち時間を最短にする放送スケジュールを求めるには、3000 年以上費やすことになる。このことから実用レベルのパラメータであってもプログラムで最適な放送スケジュールを求めるのは困難であるといえる。そこで、本稿では、ヒューリスティックなスケジューリング手法を提案する。

#### 5.1 セグメント挿入法

提案手法「セグメント挿入法」について述べる。初期状態では、スケジュール \$A\$ は空であり、最終的に \$A\$ が放送スケジュールとなる。サーバはこの放送スケジュールに従って左側から順にセグメントを放送する。

- (1) 再生倍率 \$a\$ の連続メディアデータを、セグメント \$S\_1, \dots, S\_N\$ に \$N (= 2, 3, \dots)\$ 等分する。
- (2) スケジュール \$A\$ にセグメント \$S\_N\$ を \$n\_N\$ 個加える。
- (3) \$i = N - 1\$ として、スケジュール \$A\$ に含まれている各セグメントの左側に \$S\_i\$ を各々 \$n\_i\$ 個ずつ挿入する。
- (4) \$i\$ を 1 つずつ減らし \$i = 1\$ になるまで手順 (3) を繰り返す。

セグメント挿入法ではセグメントを挿入する数 \$n\_i\$ (\$i = 1, \dots, N\$) を次のように与える。

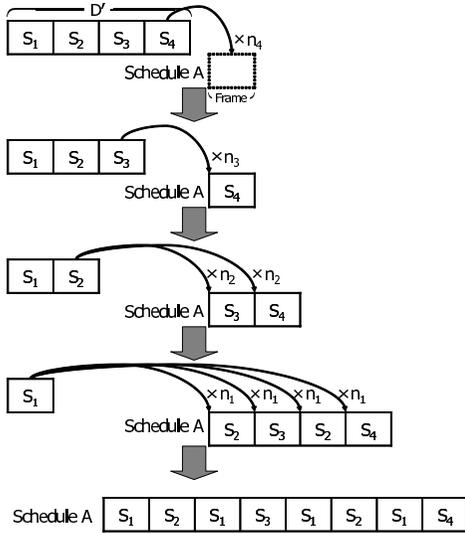


図 8 セグメント挿入法の手順 ( $a = 5, N = 4$ )

Fig. 8 A process of the segment insertion method ( $a = 5, N = 4$ ).

- $i = 1, N = 2$  の場合

$$n_1 = \lfloor a \rfloor \tag{10}$$

- $i = 1, N = 3, 4, \dots$  の場合

$$n_1 = \left\lfloor \frac{(N-2)a + 1}{2^{N-2}} - 1 \right\rfloor \tag{11}$$

- $i = 2, \dots, N$  の場合

$$n_i = 1 \tag{12}$$

$a = 5, N = 4$  の場合のセグメント挿入法を用いたスケジュール作成手順を図 8 に示す. スケジュール A にセグメントを挿入していくことがこの手法の名前の由縁である. セグメント挿入法では, 再生倍率  $a$ , 分割数  $N$  を与えれば放送スケジュールを一意に決定できる.

### 5.2 単純繰返し手法との比較

セグメント挿入法の平均待ち時間の計算結果を図 9, 図 10, 図 11 に示す. 平均待ち時間は放送時間  $D$  に比例するため, 縦軸には平均待ち時間を  $D$  で除した値をとり, 再生倍率  $a$  を横軸とした. たとえば, 5 分の MP3 データ ( $D = 5$ ) を 1 分で放送できる ( $a = 5$ ) 場合に, 連続メディアデータを 3 等分し ( $N = 3$ ), セグメント挿入法を用いる場合, セグメント挿入法で作成される放送スケジュールは  $S_1 S_1 S_2 S_1 S_1 S_3$  となり, 平均待ち時間は 16.7 秒となる. したがって, 図 10 で,  $a = 5$  の値は  $16.7/300 = 0.056$  となっている. 図中で 'SI ( $N = i$ )' ( $i = 2, 3, 4$ ) とは, データを  $i$  等分するセグメント挿入法を用いる場合の平均待ち時間を示す. 'Simple' とは 2 章で初めに述べた単純繰返し手

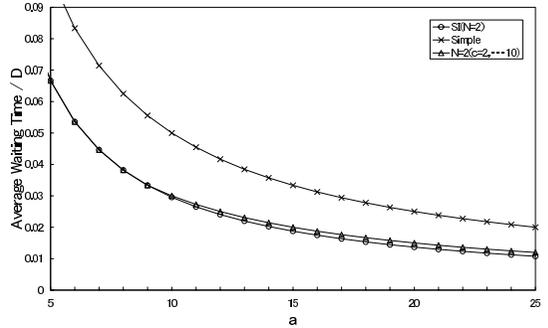


図 9 セグメント挿入法 ( $N = 2$ ) の平均待ち時間

Fig. 9 The average waiting time of the segment insertion method ( $N = 2$ ).

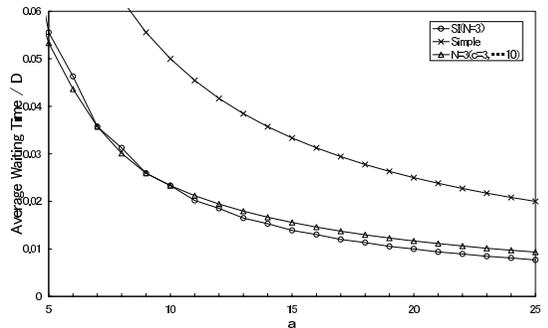


図 10 セグメント挿入法 ( $N = 3$ ) の平均待ち時間

Fig. 10 The average waiting time of the segment insertion method ( $N = 3$ ).

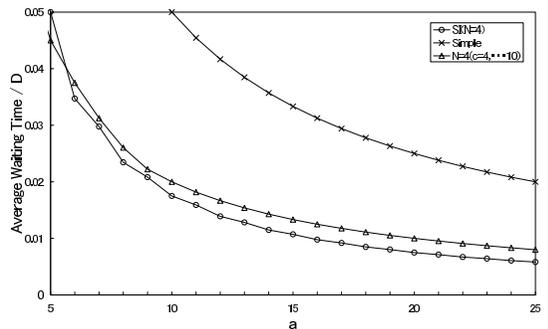


図 11 セグメント挿入法 ( $N = 4$ ) の平均待ち時間

Fig. 11 The average waiting time of the segment insertion method ( $N = 4$ ).

法を意味する. また,  $a, N$  が与えられた場合に, 1 周期中に  $c$  個のセグメントを放送するあらゆるスケジュールを考え, その平均待ち時間の最短値を算出した. すべての  $c$  に対して平均待ち時間を計算することは現実的でないため,  $c$  が 10 以下の場合のみ算出した. 図中で ' $N = i (c = i, \dots, 10)$ ' とは, 1 周期中に  $i$  個から 10 個までのセグメントを放送するあらゆる

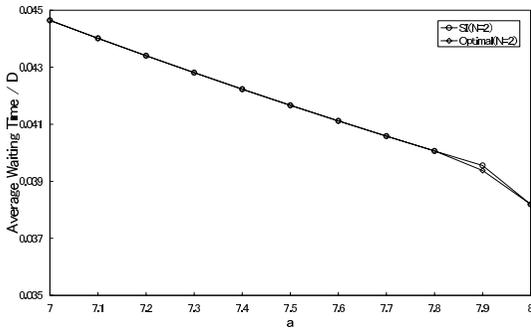


図 12 セグメント挿入法 ( $N = 2$ ) と 2 等分の最短値  
 Fig. 12 The average waiting time of the segment insertion method ( $N = 2$ ) and the minimum average waiting time of 2 divisional broadcast schedules.

スケジュールの平均待ち時間の最短値を意味する． $c$  の範囲を限定したため，セグメント挿入法の平均待ち時間の方がこの最短値より小さくなる場合がある．

5.4 節に述べるように，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．この特徴を保持するために必要な再生倍率の下限が存在するため，放送スケジュールを作成できない場合がある．たとえば， $a < 3.5$ ， $N = 4$  の場合，式 (11) を満たす  $n$  が 0 以下になり，放送スケジュールを作成できない．このような場合，分割数を小さくすれば放送スケジュールを作成できる．

2 章で述べたように，MP3 で符号化した 5 M バイトの 5 分のデータを 1 Mbps の衛星デジタル放送や bluetooth で放送する場合， $a = 7.5$  となる．単純繰返し手法では，平均待ち時間は 20 秒となるが，セグメント挿入法を用い，データを 5 等分して放送すると，8 秒となり，60%短縮できることが分かる．

5.3 2 等分の最短値との比較

2 等分のセグメント挿入法の平均待ち時間と，4 章で示した 2 等分の場合の最短の平均待ち時間を図 12 に示す．‘SI ( $N = 2$ )’ が 2 等分のセグメント挿入法，‘Optimal ( $N = 2$ )’ が 2 等分する場合の最短値を示す．グラフを見やすくするため， $7 < a \leq 8$  の範囲で平均待ち時間を示した．このように 2 等分のセグメント挿入法は，ほとんどの  $a$  において最短の平均待ち時間を与えることが分かる．しかし， $a = 7.9$  の場合のように，最短にならない場合がある．これは，セグメント挿入法の放送スケジュールはクライアントが  $S_1$  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が，2 等分して放送する場合に最短の平均待ち時間を与える放送スケジュールは，そうとは限らないためである．

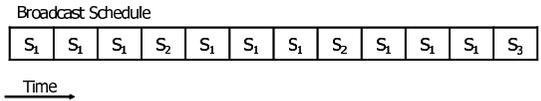


図 13  $n_i$  が任意とした場合の放送スケジュールの例 ( $N = 3$ ， $n_1 = 3$ ， $n_2 = 2$ ， $n_3 = 1$ )  
 Fig. 13 An example of the broadcast schedule in the case of arbitrarily given  $n_i$  ( $N = 3$ ， $n_1 = 3$ ， $n_2 = 2$ ， $n_3 = 1$ ).

5.4 クライアントの再生開始時刻

放送スケジュールの中には，クライアントが  $S_1$  の受信と同時に再生を開始する場合に，連続メディアデータを最後まで途切れずに再生できる場合と，できない場合がある．たとえば，データを 3 等分し  $S_2S_1S_3$  の順に放送する場合， $S_2$  が放送されているタイムスロットでユーザがデータの受信要求を出し，クライアントが  $S_1$  の受信と同時に再生を開始すると， $1 < a < 2$  の場合， $S_1$  の再生終了までに  $S_2$  が放送されておらず，データを最後まで途切れずに再生できない．この場合， $S_1$  を受信してもいくら待ってから再生しなければならず，クライアントは，適切な再生開始時刻を知る必要がある．しかし， $S_1S_2S_3$  の順に放送する場合，受信要求を出す時刻にかかわらず，クライアントは次の放送周期の  $S_1$  の受信と同時に再生を開始し，データを最後まで途切れずに再生できる．そのため，クライアントは放送スケジュールを考慮せず， $S_1$  を受信した時点で再生を開始すればよいという利点がある．

セグメント挿入法では，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ように，また平均待ち時間ができるだけ短縮できるように式 (10)，(11)，(12) で  $n_i$  を決定する．

5.5 セグメント挿入法の平均待ち時間

セグメント挿入法では  $n_i$  ( $i = 1, \dots, N$ ) を式 (10)，(11)，(12) で与えたが，以下の証明では， $n_i$  は任意とする．たとえば， $N = 3$ ， $n_1 = 3$ ， $n_2 = 2$ ， $n_3 = 1$  とした場合の放送スケジュールは図 13 のようになる．

放送スケジュール作成過程において，最後に  $S_1$  をスケジュール A に挿入する前のスケジュール A に含まれるセグメントの数を  $l$  とする． $S_1$  を挿入すると，スケジュール A に含まれるセグメントの数は  $n_1l + l$  となるため， $c = n_1l + l$  である．放送スケジュールに  $S_1$  は  $n_1l$  個含まれる．5.4 節に述べたように，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．そのため，あるタイムスロットで受信要求を出した場合の，そのタイムスロットにおける平均待ち時間は，次のタイムスロットで  $S_1$  が放送されている場合， $\Delta / 2$

となる．平均待ち時間がこのようになるタイムスロットの数は放送スケジュールに含まれる  $S_1$  の数に等しく， $n_1 l$  個となる．次のタイムスロットで  $S_1$  が放送されていない場合，さらに次のタイムスロットで  $S_1$  が放送されているため，平均待ち時間は  $3\Delta/2$  となる．平均待ち時間がこのようになるタイムスロットは  $c - n_1 l = l$  個ある．

よって，セグメント挿入法の平均待ち時間  $W$  は

$$\begin{aligned} W &= \frac{1}{c} \left( n_1 l \times \frac{\Delta}{2} + 3l \times \frac{\Delta}{2} \right) \\ &= \left( \frac{1}{2} + \frac{1}{n_1 + 1} \right) \Delta \end{aligned} \quad (13)$$

で与えられる．これより，分割数が大きいほど，また  $S_1$  の挿入する数  $n_1$  の値が大きいほど，平均待ち時間は短くなることが分かる．単純繰返し手法の平均待ち時間は  $D'/2$  であり，式 (13) との大小関係を調べると，セグメント挿入法の平均待ち時間は単純繰返し手法より短いことが分かる．

## 6. 関連研究

連続メディアデータの放送型配信に関する研究が多数行われている．

Pyramid broadcasting (PB) 法<sup>1)</sup> では  $M$  個のデータを  $K$  個のセグメントに分割し， $K$  個のチャンネルを用いて放送する． $i$  番目のセグメントの再生時間  $D_i$  は  $\alpha^{i-1} D_1$  となるように分割する．このように分割し， $i$  番目のチャンネルでは  $M$  個のデータの  $i$  番目のセグメントを連続して放送する．初めのセグメントの再生時間は短く，頻繁に放送されるため，クライアントの待ち時間を短縮できる．しかし，この手法を用いると，分割数が大きいほど後のセグメントの再生時間が長くなるため，クライアントに必要なバッファ容量が大きくなる．

Skyscraper broadcasting (SB) 法<sup>2)</sup> では，この問題を考慮し，分割すると再生時間が  $D_1 W$  より長くなるセグメントは再生時間が  $W$  になるように分割する．

$W$  を変更することで，必要なバッファの容量や，待ち時間を任意に与えられる．

SB 法において，放送するデータを動的に変更する Dynamic skyscraper broadcasting (DSB) 法<sup>3)</sup> も提案されている．

Permutation-based pyramid broadcasting (PPB) 法<sup>4)</sup> では， $K$  個のチャンネルをさらに  $p$  個のサブチャンネルに分ける． $i$  番目のセグメントを  $p$  個のサブチャンネルで放送する． $p$  個のサブチャンネルの周期が始まる時刻をずらすことで，PB 法より待ち時間を短縮でき

る．PPB 法では，単一のチャンネルで連続メディアデータを配信する時分割多重について言及しているが，単一チャンネルでクライアントの待ち時間を短縮する手法について詳しく述べていない．

待ち時間をさらに短縮する Harmonic broadcasting (HB) 法<sup>5)</sup> がある．HB 法では，データを  $N$  個のセグメントに等分割し，さらに  $i$  番目のセグメントを  $i$  個のサブセグメントに分割する． $i$  番目のセグメントの  $i$  個のサブセグメントを  $i$  番目のチャンネルで放送する．1 番目のチャンネルの帯域幅は，データのビットレートと等しく， $i$  番目のチャンネルの帯域幅は，1 番目のチャンネルの  $1/i$  とする．したがって， $i$  番目のチャンネルの再生倍率は  $1/i$  となる．このように HB 法では再生時間と放送時間の差異を考慮しているが，単一チャンネルでクライアントの待ち時間を短縮する手法について述べていない．

HB 法では，クライアントにバッファが必要になるが，Fast broadcasting (FB) 法<sup>6)</sup> では，バッファがある場合より長く待つことで，バッファがなくてもデータを再生できる．FB 法は，データを  $2^K - 1$  個に等分割し， $K$  個のチャンネルで放送する．FB 法において，使用するチャンネル数を動的に変更する手法も提案されている<sup>9)</sup>．

さらに多くのセグメントを 1 つのチャンネルで放送する Pagoda broadcasting 法も提案されている<sup>7)</sup>．

このように複数のチャンネルを用い，クライアントの待ち時間を短縮する研究はあるが，単一チャンネルに比べ，処理やシステムが複雑になる．また，複数のチャンネルから同時にデータを受信できる機器は少ない．本稿は，単一チャンネルでクライアントの待ち時間を短縮している点が異なる．

## 7. 結論

本稿では，単一チャンネルでクライアントの待ち時間を短縮する「分割放送方式」を提案した．分割放送方式では，クライアントがデータを最後まで途切れずに再生できることを考慮したうえで，データをいくつか等に等分割してデータの初めの部分を頻繁に放送することで，クライアントの待ち時間を短縮する．この場合，データの分割数および分割したデータのスケジューリングが待ち時間に影響を与える．本稿では，2 等分する場合については数学的な解析を行い，平均待ち時間を最短にする放送スケジュールを導出した．この放送スケジュールは，1 周期中に前半のデータを，放送に要する時間と再生時間の比によって決まる回数だけ放送し，次に，後半のデータを一度だけ放送する．MP3

で符号化した連続メディアデータを 1 Mbps の衛星デジタル放送で配信する場合, 単純に繰り返して放送する手法より, 平均待ち時間を約 38% 短縮できる. このように, データを 2 等分して放送するだけで, 平均待ち時間の大幅な短縮が期待できる.

3 等分以上の場合については, ヒューリスティックなスケジューリング手法「セグメント挿入法」を提案し, 評価を行った. 評価の結果, セグメント挿入法はデータを繰り返して放送する手法よりも, 平均待ち時間を短縮でき, 分割数が多いほど平均待ち時間を短縮できることが分かった.

今後, セグメント挿入法とは別のスケジューリング手法の提案や, 単一チャンネルで複数のデータを配信する場合の検討, それらのデータ間に相関性がある場合の検討があげられる.

謝辞 本研究は, 文部科学省 21 世紀 COE プログラム(研究拠点形成費補助金), 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) 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).
- 2) 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).
- 3) 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).
- 4) 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).
- 5) 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).
- 6) 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).
- 7) Paris, J.-F., Carter, S.W. and Long, D.D.E.: A hybrid broadcasting protocol for video on demand, *Proc. Int. Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS '98)*, pp.127-132 (1998).
- 8) Paris, J.-F., Long, D.D.E. and Mantey, P.E.: Zero-delay broadcasting protocols for video-on-demand, *Proc. ACM Int. Multimedia Conf. (Multimedia '99)*, pp.189-197 (1999).
- 9) Tseng, Y.-C., Hsieh, C.-M., Yang, M.-H., Liao, W.-H. and Sheu, J.-P.: Data broadcasting and seamless channel transition for highly-demanded videos, *Proc. IEEE INFOCOM '00*, pp.727-736 (2000).
- 10) Almeroth, K.C. and Ammar, M.H.: On the use of multicast delivery to provide a scalable and interactive video-on-demand service, *IEEE Journal on Selected Areas in Communications*, Vol.14, pp.1110-1122 (1996).
- 11) Dan, A., Sitaram, D. and Shahabuddin, P.: Scheduling policies for an on-demand video server with batching, *Proc. ACM Int. Multimedia Conf. (Multimedia '94)*, pp.15-23 (1994).
- 12) Horn, G.B. and Rasmussen, J.E.: A scalable and reliable paradigm for media on demand, *IEEE Computer*, Vol.34, No.9, pp.40-45 (2001).
- 13) 高橋政雄: 一目で分かる MP3 の概要, INTERFACE 2000 年 8 月号, pp.70-77, CQ 出版, 東京 (2000).
- 14) 次世代放送コンテンツの振興に関する調査研究会: デジタル放送新時代への挑戦 (2000). <http://www.soumu.go.jp/joho-tsusin/pressrelease/japanese/housou/000718j702.html>  
(平成 14 年 4 月 23 日受付)  
(平成 15 年 4 月 3 日採録)



義久 智樹

1979 年生. 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修了. 現在, 同専攻博士後期課程在学中. 地上波デジタル放送に興味を持つ.



塚本 昌彦 (正会員)

1987年京都大学工学部数理工学科卒業。1989年同大学院工学研究科博士前期課程修了。同年、シャープ(株)入社。1995年大阪大学大学院工学研究科情報システム工学専攻講師を経て、1996年同専攻助教授。2002年より同大学院情報科学研究科マルチメディア工学専攻助教授となり、現在に至る。工学博士。ウェアラブルコンピューティング・ユビキタスコンピューティングに興味を持つ。ACM, IEEE等8学会の会員。



西尾章治郎 (正会員)

1975年京都大学工学部数理工学科卒業。1980年同大学院工学研究科博士後期課程修了。工学博士。京都大学工学部助手、大阪大学基礎工学部および情報処理教育センター助教授、大阪大学大学院工学研究科情報システム工学専攻教授を経て、2002年より同大学院情報科学研究科マルチメディア工学専攻教授となり、現在に至る。2000年より大阪大学サイバーメディアセンター長を併任。この間、カナダ・ウォータールー大学、ビクトリア大学客員。データベース、マルチメディアシステムの研究に従事。現在、ACM Trans. on Internet Technology, Data & Knowledge Engineering, Data Mining and Knowledge Discovery等の論文誌編集委員。情報処理学会フェロー含め、ACM, IEEE等9学会の会員。