

情報の性質を考慮した放送スケジューリング方式

カン ギョウビ* 浅田一繁* 白田由香利* 飯沢篤志* 古瀬一隆†

†株式会社次世代情報放送システム研究所

〒111-0035 東京都台東区西浅草 1-1-1 かんぼ浅草ビル 8F

Tel: (03)5326-7385

E-mail: [kan|asada|shirota|izw]@ibl.co.jp

*株式会社リコー

〒112-0002 東京都文京区小石川 1-1-17 とみん日生春日町ビル 7F

Tel: (03)3815-7261

†筑波大学電子・情報工学系

〒305-0087 茨城県つくば市天王台 1-1-1

Tel: (0298)53-6224

†株式会社リコーより株式会社次世代情報放送システム研究所へ兼任出向中。

あらまし

本稿では、デジタル放送を用いたプッシュ型情報配信システムにおける配信スケジューリング方式を提案する。これまでの配信スケジューリングに関する研究では、主に配信時間のみを問題にした1次元的なスケジューリング方式が提案されてきたが、我々は、デジタル放送ならではの特徴に着目し、配信時間のみならず、配信帯域幅も考慮した2次元的なスケジューリング方式を考案した。この方式は、従来方式で用いられてきた配信するデータのアクセス確率だけではなく、開始時間の制限や帯域幅の変化など、より多様なデータ性質を考慮した配信スケジューリングを可能にする。本稿では、考慮すべきデータの性質とそれに基づいたシステムモデル、さらに、このようなモデルにおけるスケジューリングアルゴリズムを設計する際の考え方や方向性を示した。

キーワード スケジューリング、モデル、アルゴリズム、デジタル放送

A Scheduling Mechanism for Broadcast Data Based on Data Properties

Xiaowei Kan** Kazushige Asada†* Yukari Shirota†* Atsushi Iizawa†* Kazutaka Furuse†

†Information Broadcasting Laboratories, Inc.

Kampo Asakusa Build. 8F1-1-1 Nishi-Asakusa, Taito-ku Tokyo, JAPAN 111-0035

Tel: +81-3-5326-7385

E-mail: [kan|asada|shirota|izw]@ibl.co.jp

*RICOH Co., Ltd.

Tomin Nissei Kasugacho Build. 7F1-1-17 Koishikawa, Bunkyo-ku Tokyo, JAPAN 112-0002

Tel: +81-3-3815-7261

†Institute of Information Sciences and Electronics, University of Tsukuba

1-1-1 Tennodai, Tsukuba-shi Ibaraki, JAPAN 305-0087

Tel: +81-0298-53-6224

†The authors are partly on loan from Ricoh Company, Ltd. to Information Broadcasting Laboratories, Inc.

Abstract

In this paper, we propose a mechanism of scheduling for push broadcast services in digital broadcasting. In previous work, the scheduling problem has been considered as a problem of one-dimensional allocation focusing solely on the time dimension. We redefine the problem as that of two-dimensional allocation, with both time and bandwidth of the data taken into account, and propose a mechanism truly designed for digital broadcasting. The proposed mechanism uses not only data access statistics as in previous work but also the properties of the data such as restrictions in the start time and change in the bandwidth. In the paper, we show data properties to be considered, the system model based on those properties and how to use the model in designing a scheduling algorithm for the model.

key words

Scheduling, Model, Algorithm, Digital Broadcasting

1. はじめに

近年、マルチメディアアプリケーションの増加およびデジタル放送時代の到来に伴い、デジタル放送を用いてマルチメディアデータをクライアントに配信する放送型情報システムに関する研究が盛んに行われている[1]。このような放送型(プッシュ型)情報システム(図1)では、サーバーは決められた配信スケジュールに従ってマルチメディアデータベースに蓄積されているデータを広い配信帯域に乗せて配信する。クライアントは必要なデータだけを配信帯域から選択受信する。これまで、我々は、このようなシステムにおけるフィルタリング方式やキャッシング方式などについて提案してきた[2][3]。本稿では、このようなシステムにおける配信スケジューリング方式について提案する。

一般的に、配信スケジューリングに関する研究はいくつかの課題に直面する。まず、適切なシステムモデルの確立が要求される。これまでの配信スケジューリングに関する研究は主に配信時間のみを問題にした1次元的なスケジューリングである。しかし、後述するように(2章参照)、デジタル放送のスケジューリングシステムでは、配信時間のみならず、配信帯域幅も考慮しなければならない。つまり、2次元的なスケジューリングモデルが必要となる。さらに、配信するデータは、従来方式で用いられているデータのアクセス確率に限らず、配信開始時間の制限や帯域幅の変化など、より多様な性質をもつ。このため、必要十分なシステムモデルの確立は容易ではない。次に、確立したモデルにおけるスケジューリングアルゴリズムの考案およびアルゴリズムの効率化が要求される。ここで、所謂クライアント側の response time や tuning time だけではなく、スケジューリング処理自身の処理コストも考慮しなければならない。

これまでも、配信スケジューリングためのモデルやアルゴリズムは数多く報告されている。Su 氏らによる研究[4]は、配信する個々のデータは、アクセス確率以外均質であるというモデルに基づくものであるため、現実の複雑さをモデリングするには不十分と思われる。アクセス確率だけでなく、データ間の相関性まで考慮した方式としては、西尾

氏らによって提案されているCBS法[5]がある。CBS法は、データ間の相関を重み付き完全グラフで表現し、グラフ内のハミルトン閉路のうち、重み総和が最小となる閉路を選定し、その順でデータを配信する。CBS法によって、データ間の相関性が考慮されるようになったが、他の数多くの性質についてはまだ十分にモデリングされていない。なお、ハミルトン問題はNP完全であるので、アルゴリズムの効率化が必要と思われる。一方、Vaiday氏らによる研究[6]は、マルチチャンネル対応ではあるが、基本的には配信時間のみを考慮した1次元的なスケジューリング方式である。清野氏[7]らによって提案されたRAID型放送システムは、マルチチャンネルをまたがってデータのページを分散し配置することにより、受信失敗率を下げることを目指しているが、これも基本的には固定ビットレート(後述)の1次元的なスケジューリングである。

本研究では、デジタル放送ならではの特徴およびデータの非均質性に着目し、データの多様な性質を考慮した2次元的な配信スケジューリングシステムのモデルおよびスケジューリングアルゴリズムを提案し、人手による配信スケジューリングの支援を目指す。

本稿は以下のように構成される。まず、2章において、例を用いてデジタル放送における配信スケジューリングを概説しながら、関連用語を説明する。3章では、配信スケジューリングシステムのモデルを定義し、4章では、定義したモデルにおける配信スケジューリングアルゴリズムについて説明する。5章で、上記のモデルおよびアルゴリズムに関する議論を行った後、最後に今後の課題を示す。

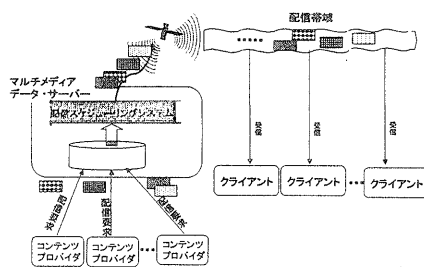


図1 想定するシステム

2. デジタル放送における配信スケジューリング

我々の提案を説明する前に、本節では、デジタル放送における配信スケジューリングについて簡単に説明しながら、関連用語を定義する。

図2に a、b、c、d、e という5つのデータ(箱で表している)に関する配信スケジュールを示す。箱の高さはそのデータを配信するのに必要な配信帯域の幅、箱の長さはそのデータを配信する期間を表す。データ a,d には論理チャンネル番号"1"が振られ、b、c、e には論理チャンネル番号"2"が振られる。クライアントは自分の受信チャンネルをこの論理チャンネル番号に合わせ、そのチャンネルのデータを受信する。配信スケジューリングとは、これらの箱を帯域幅制限を越えない範囲で、しかるべき時刻に積みかさねながら配置することである。ただし、この際、任意の時刻に積みかさねられているデータは別々の論理チャンネルのデータでなければならない。図2で示したように、時刻 t_1 と t_2 からそれぞれbとaの配信が開始された。時刻 t_3 ではbの配信が終わると同時にcの配信が開始されている。ここでは時刻 t_5 に注目していただきたい。この時刻からcと異なる帯域幅をもつeを配信するために、dの帯域幅を圧縮することにした。もしdの帯域幅を圧縮できなければ、dの配信が完全に終了するまでeの配信ができないため、帯域の無駄が生じる。このアナログ放送には不可能な番組ごとの配信帯域幅の伸縮の技術は「可変ビットレート」と呼ばれ、デジタル放送の特技である。可変ビットレート技術は配信帯域の効率的分配によって、より多くのデータの配信を可能にしたと同時に、配信スケジューリングを一層

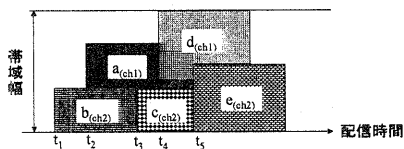


図2. デジタル放送における配信スケジューリング

複雑化させた。

ここまでは配信帯域の分配を中心に説明してきた。次に、どの時刻にどのデータを配信すべきかという配置のポリシーから配信スケジューリングを見てみよう。一般的に、配信スケジュールを組む時には、次に示すような配信するデータの様々な性質を考慮しながら、配置していく。

◇ 所要配信帯域幅

データのフォーマットによって、配信に必要な帯域幅は異なる。例えば、動画なら一定の帯域幅を確保する必要があるが、ゲームソフトなら時間はかかるが狭い帯域幅でも問題はない。さらに、可変ビットレートの場合、図2におけるデータdのように、帯域幅が途中で変化する可能性もある

◇ 所要配信時間

上記の帯域幅で、あるデータを配信するのに必要な時間

◇ 配信開始時刻の制限

制限あるデータはその制限どおりに配置しなければならない。ある固定時刻に配信開始といった制限以外にも、制限の種類はいろいろ考えられる。例えば、「〇〇時以前(or以降)に配信」や「〇〇時から〇〇時の間に配信」などである

◇ 配信中止可能性

必ず配信しなければならないわけではなく、配信帯域が余った場合に配信し、余りがなければ止めてもよいというデータは「中止可能」と言う。逆に必ず配信しなければならないデータは「中止不可」と言う

◇ データ量の変化可能性

中継番組のデータなどでは、番組が延長する可能性がある。この場合は、配信帯域幅は変わらないまま、配信時間の延長によるデータ量の増加となる。逆に、一定時間内に配信するHDTVのデータは途中からSDTVに変わる可能性もある。この場合は、配信時間は変わらないが、配信帯域幅の圧縮によりデータ量は減少する。

実際に配信スケジュールを組む時は、これら以外にも

もっと多くの性質が考慮される。配信するすべてのデータの性質をすべて考慮しながら、互いの制約を調節し、配置しなければならない。さらに、前述の可変ビットレートによる複雑さも加わるので、デジタル放送における配信スケジューリングはかなり複雑な作業となる。残念ながら、既にサービスを開始しているプッシュ型情報システム(e.g. Mega Wave Select[8]など)における配信スケジューリングはシステム化されずに人手に頼っているのが現状である。今後、マルチメディアデータの種類および量の増加に伴ない、人手だけでスケジュールを組むのはますます困難になり、何らかのシステム化による支援が必要と思われる。そこで、本研究はこのシステム化に欠かせない配信スケジューリングモデルおよびそのモデルにおけるスケジューリングアルゴリズムを提案し、人手による配信スケジューリングの支援を目指す。

3. 配信スケジューリングシステムのモデル

本稿では、静的スケジューリングを前提として、まず単純なケースからモデリングを行うことにした。つまり、配信帯域幅も、配信期間も、配信すべきデータもすべてスケジューリングの前に決められている。ただし、スケジューリング後データの量が増減する可能性がある。

3.1. 配信帯域のモデル

まず、プッシュ型情報システムにおける配信帯域のモデルについて説明する。図2に示したように、配信帯域は配信時間と帯域幅からなる2次元空間における矩形で表すことができる。矩形の長さTはスケジューリングの期間で

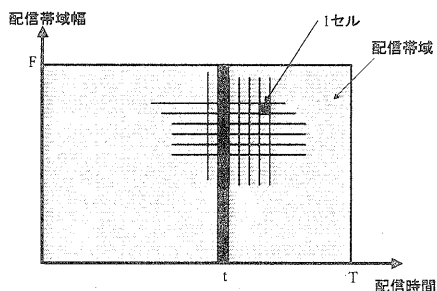


図3. 配信帯域モデル

あり、単位時間数で表す。矩形の高さFは配信帯域の幅であり、単位帯域幅数で表す。ここで、時間軸および帯域幅軸の座標は連続型でなく、離散型となる。さらに、この2次元空間上において、単位時間内における単位帯域幅での伝送容量を1セル(cell)と呼ぶ。配信帯域全体で配信できるデータの量はT*Fセルであり、単位時間tに配信できるデータの量はFセルである。

3.2. 配信データのモデル

上記の配信帯域で配信しようとするデータは $\{i_1, i_2, \dots, i_n\}$ とする。個々のデータは1つのセル大きさをもつ要素(element)何個かがつながったものである。本稿では、下記の表に示す変量を用いて2章で述べた配信スケジューリングの際に考慮すべきデータの性質を表す。

変量	性質	値
τ_i	所要配信時間	単位時間数で表す
$\beta_i(t)$	所要配信帯域幅	$\beta_i(t)$ は時刻tにおけるiの要素数を表す
θ_i	配信中止可能性	$\theta_i=0$:中止可 $\theta_i=1$:中止不可
ϕ_i	論理チャンネルの番号	自然数
$(\delta_{ci}, \delta_{pi}, \delta_{mi})$	データ量変化可能性	δ_{ci} は変化の種類を表し、 δ_{pi}, δ_{mi} は最大変化単位時間(or要素数)を表す
$(\tau_{ci}, \tau_{bi}, \tau_{ei})$	配信開始時刻制限	τ_{ci} は制限の種類を表す。 τ_{bi} と τ_{ei} は制限時刻を表す

データを箱と見なすときに、 $\beta_i(t)$ はその箱の高さを表す。可変ビットレートの場合、箱の高さは時間の関数である。 $\forall t_i, t_j$ に対して $\beta_i(t_i) = \beta_i(t_j)$ が成り立つならそのデータiの帯域幅(i.e.箱の高さ)は固定である。そうでなければ帯域幅は可変となる。一般的に、箱の面積、つまり要素数で表せるデータの量は $\sum_{i=1}^n \beta_i(t)$ となる。

箱の面積の変化可能性は、3つ組み $(\delta_{ci}, \delta_{pi}, \delta_{mi})$ で表す。 δ_{ci} は変化の種類を表し、下記の値を取る。

δ_{ci} 値	意味
0	無変化
1	配信時間の延長可能性がある。最大 δ_{pi} 単位時間の延長が可能である

2	配信時間の短縮可能性がある。最大 δ_{mi} 単位時間の縮小が可能である
3	配信時間の延長・短縮の可能性が共にある。最大 δ_{pi} 単位時間の延長ないし、最大 δ_{mi} 単位時間の短縮が可能である
4	帯域幅の拡大可能性がある。最大要素数 δ_{pi} 個分を大きくする可能性がある
5	帯域幅の圧縮可能性がある。最大要素数 δ_{mi} 個分を小さくする可能性がある
6	帯域幅の拡大・圧縮の可能性が共にある。最大要素数 δ_{pi} 個分の拡大ないし、最大要素数 δ_{mi} 個分の圧縮が可能である

時間軸におけるデータの配置制限は、 $(\tau_{ci}, \tau_{bi}, \tau_{ai})$ で表す。 τ_{ci} は制限の種類を表し、下記の値を取る。

τ_{ci} の値	意味
0	無制限
1	τ_{bi} に配置
2	τ_{bi} 以降の任意時刻に配置
3	τ_{ci} 以前の任意時刻に配置
4	τ_{bi} と τ_{ci} の間の任意時刻に配置

3.3. スケジュールのモデル

配信スケジュールは下記の集合でモデル化する。

$$S_t = \{(i_{n1}, \beta_{n1}(t)), (i_{n2}, \beta_{n2}(t)) \dots (i_{nst}, \beta_{nst}(t)) \mid i_{ny} \in \{i_1, i_2, \dots, i\}\}$$

ここで、 S_t は単位時刻 t に配信するデータの ID (i_y) とその配信帯域幅 ($\beta_y(t)$) の二元組みからなる集合である。

図4にその例を示す。4単位時間内に、5単位帯域幅の配信帯域において、データ a、b、c、d、e は図4に示したようにスケジュールされる。この配信スケジュールは下記の集合で表わされる。

$$S_1 = \{(a,2), (b,1), (c,2)\}$$

$$S_2 = \{(a,1), (b,2), (c,1), (d,1)\}$$

$$S_3 = \{(a,2), (b,1), (d,2)\}$$

$$S_4 = \{(a,1), (e,3)\}$$

一般的には、スケジュールの際に下記の制約条件を満たさなければならない。

◇ 配信容量に関する制約

任意の単位時刻に配信するすべてのデータの帯域幅の合計は F を超えてはならない。つまり、

$$\forall t \in \{1, \dots, T\}, \sum_{i=1}^N \beta_i(t) \leq F$$

が成り立たなければならない。

◇ 論理チャンネルに関する制約

任意の単位時刻に配信するデータは1チャンネル1データに限る。言い換えれば、同じ時刻には同じチャンネルに一種類のデータしか流せない。つまり $\forall t \in \{1, \dots, T\}, \forall l \neq m \in \{1, \dots, st\}$ に対し、

$$\varphi_{il} \neq \varphi_{im}$$

が成り立たなければならない。

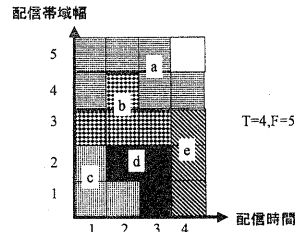


図4. 例

4. 配信スケジューリングアルゴリズム

3章で定義しているモデルでは、多種多様な情報の性質やシステムに関わる要素および制約条件がモデリングされているので、各々の性質を考慮しながら実用的な配信スケジュールを算出するアルゴリズムの設計が可能となる。一般的に、このようなモデルにおける配信スケジューリングアルゴリズムはいろいろ考えられる。例えば、配信帯域を箱とみなし、個々のデータを駒とみなす上、制限条件を満たす前提に、すべての駒ないしできるだけ多くの駒を箱に詰めていく問題として、既存の箱詰めパズル (packing puzzle) のアルゴリズムを利用することなどが考えられる。我々は、本格的にアルゴリズムを考案する前に、まず単純なアルゴリズムを設計することによって、本モデルにおける配信スケジューリングアルゴリズムを設計する際の考え方や方向性を探ることにした。本節では、この単純なアルゴリズムを説明しながら、本モデルにおける配信スケジューリングアルゴリズムを設計する際の考え方や方向性を示す。

ここで、個々のデータが矩形かつ配信時間の延長のみがある場合、つまり、 $\forall t_i, t_j$ に対しては $\beta_i(t_i) = \beta_i(t_j)$ かつ $\forall i \in \{i_1, \dots, i_N\}$ に対して、 $\delta_{ci} \notin \{2, 3, 4, 5, 6\}$ であるとの前提を置くという単純なケースからまず考えることにした。

まず、アルゴリズムの最上位関数 `SchedulingAlgorithm` について説明する。ここで、 F は配信帯域幅の上限であり、 T は配信期間である。 I はすべてのデータの集合を表す。つまり、 $I = \{(i_1, a_1), (i_2, a_2), \dots, (i_N, a_N)\}$ 、 $a_i = (\tau_{is}, \beta_i(t), \theta_{is}, \phi_{is}, (\delta_{ci}, \delta_{pi}, \delta_{mi}), (\tau_{cis}, \tau_{bis}, \tau_{eis}))$ となる。この関数は、基本的に、時間軸に沿って各時刻に配置可能なデータがあるかどうかを探し、配置できるデータがあれば、その時刻を開始時刻としてそのデータを配置する。配置できるデータがなければ次の時刻へいく。最後に、中止不可のデータが残っていたら、スケジューリング不可とし、エラーを返す。中止不可のデータが残っていなければスケジューリング成功とし、その配置結果を返す。最も単純なものであるが、上記のモデルで記述したデータの様々な性質を考慮しながら配置していく必要がある。具体的には、下記の通りである。すべてのデータを中止不可 (i.e. $\theta_i = 1$) と中止可能 (i.e. $\theta_i = 0$) データとの二種類に分ける。まず、中止不可のデータから配置し、次に中止可能なデータを配置する。中止不可のデータに関しては、まず、開始時刻制限が固定であるデータから配置し、次に「〇〇時刻以降」という制限の付いたデータを配置する。引き続き、「〇〇時刻以前」という制限の付いたデータを配置し、最後に「〇〇時刻以降、〇〇時刻以前」という制限の付いたデータを配置する。一通り配置し終えた後、制限があるデータが残っているなら、スケジューリング失敗とする。そうでなければ配置し続ける。最後に中止不可のデータが残っている場合は同じく失敗とする。以下、この関数の擬似コードを示す。

```
SchedulingAlgorithm(I, F, T){
  I' = {i |  $\theta_i = 1, i \in I$ };
  Bt = F; St =  $\Phi$ ; t  $\in$  {1, ..., T};
  te[N] = 0;
  for all 1  $\leq$  t  $\leq$  T do
    for all i  $\in$  I' do
      if ( $\tau_{ci} = 1 \wedge \tau_{bi} = t \wedge \text{Locatable}(i, t)$ )
        Locate(i, t);
  for all 1  $\leq$  t  $\leq$  T do
```

```
    for all i  $\in$  I' do
      if ( $\tau_{ci} = 2 \wedge \tau_{bi} \leq t \wedge \text{Locatable}(i, t)$ )
        Locate(i, t);
    for all 1  $\leq$  t  $\leq$  T do
      for all i  $\in$  I' do
        if ( $\tau_{ci} = 3 \wedge 0 \leq t \leq \tau_{ci} \wedge \text{Locatable}(i, t)$ )
          Locate(i, t);
      for all 1  $\leq$  t  $\leq$  T do
        for all i  $\in$  I' do
          if ( $\tau_{ci} = 4 \wedge \tau_{bi} \leq t \leq \tau_{ci} \wedge \text{Locatable}(i, t)$ )
            Locate(i, t);
      if ( $\exists i \in I', \tau_{ci} \neq 0$ ) failed;
      for all 1  $\leq$  t  $\leq$  T do
        for all i  $\in$  I' do
          if ( $\text{Locatable}(i, t)$ ) Locate(i, t);
      if ( $|I'| \neq 0$ ) failed;
      I' = {i |  $\theta_i = 0, i \in I$ };
      for all 1  $\leq$  t  $\leq$  T do
        for all i  $\in$  I'' do
          if ( $\text{Locatable}(i, t)$ ) Locate(i, t);
      return (St, t  $\in$  {1, ..., T});
}
```

次に、`Locate` 関数について説明する。上位関数 `SchedulingAlgorithm` で配置可能なデータが見つかったら、関数 `Locate` を呼び出し、そのデータをスケジュールに加える。`Locate` はとても簡単な関数である。まず、配置するデータ i の配信開始時刻から終了時刻までの間の各単位時刻におけるセル数からデータ i の帯域幅分を除き、データ i をその単位時刻の S_t に加える。最後にデータ i の終了時刻を記録した上、データ i を配信すべき集合から除く。以下に、この関数の擬似コードを示す。

```
Locate(i, t){
  for all t  $\leq$  t'  $\leq$  t +  $\tau_i$  do
    Bt' = Bt -  $\beta_i(t)$ 
    St' = St  $\cup$  {(i,  $\beta_i(t)$ );
  te[i] = t +  $\tau_i$ ;
  I' = I' - {i};
  return;
}
```

最後に、`Locatable` 関数について説明する。上位関数 `SchedulingAlgorithm` において、配置すべきデータ集合から配置しようとするデータを選択した後、関数 `Locatable` を呼び出し、配置可能かどうかを確認する。この関数の動作を簡単に説明すると、次のようになる。まず、配信帯域が足りるかどうかを確認する。具体的には、配置しようとするデータ i の配信開始時刻から終了時刻までの間の各単位時刻における残り配信帯域幅のセル数がデータ i の所要帯域幅分に足りるかどうかを確認する。足りな

れば、「配置不可」を返す。足りるなら次の確認へ進む。次には、論理チャンネルの制約条件を満たすかを確認する。具体的には、配信開始時刻から終了時刻までの間の各単位時刻において、既に配置したデータ中にデータ i と同じ論理チャンネル番号をもつものがあるかを確認する。あれば、「配置不可」を返す。なければ次の確認へ進む。配信時間の延長可能性のあるデータが延長するときでも、スケジューラが破綻しないために、延長可能性のあるデータの直後には中止不可かつ開始時刻固定のデータを配置しないという最も単純な方法をとる。そこで、この関数の最後に、配置しようとするデータ i が開始時刻固定かつ中止不可であるなら、関数 `Extendable` を呼び出して、直前のデータが「配信時間の延長可能」かどうかを確認する。関数 `Extendable` は、配置しようとする時刻の1つ前の時刻を引数とし、その時刻を配信終了時刻とするデータは配信時間の延長可能かどうかを調べる。直前のデータが可変かつ配置しようとする時刻から最大延長時間までの間には2つデータの所要帯域幅分の配信帯域が残っていないければ、「延びあり」を返す。そうでなければ「延びなし」を返す。

以下に、関数 `Locatable` の擬似コードを示す。

```

Locatable(i,t){
  for all  $t \leq t' \leq t + \tau_i$  do
    if( $B_{t'} - \beta_i(t) < 0$ ) return NO;
  for all  $t \leq t' \leq t + \tau_i$  do
    for all  $j \in S_{t'}$  do
      if( $\phi_j = \phi_i$ ) return NO;
  if( $\tau_{c_i} \neq 0 \wedge \theta_{i=1} \wedge \text{Extendable}(t)$ ) return NO;
  return YES;
}
Extendable(t){
  for all  $t \in [j] = t-1 \wedge \delta_{c_j} = 1$  do
    for all  $t \leq t' \leq t + \delta_{p_j}$  do
      if( $B_{t'} < \beta_j(t) + \beta_i(t)$ ) return YES;
  return NO;
}

```

5. 議論

本節では、我々の提案したシステムモデルおよびスケジューリングアルゴリズムに関する議論を示す。

5.1. システムモデルについて

本稿で提案した配信スケジューリングシステムのモデル

は、システムにおける最も基本的な要素および制約条件をモデル化したものの、配信スケジューリングシステムのモデリングにおける一つの考え方を示したに過ぎない。実際の配信スケジューリングシステムのためのより高度なモデリングを求めるには、さらに工夫が必要と思われる。ここで、いくつかの例を挙げながら、その方向性を示す。

まず、3. 1に述べた配信帯域モデルについて考えよう。

ここでは、配信帯域をセルの集合としてモデル化しただけであるが、実際の配信スケジューリングの際には、下記のことを考慮しながらスケジュールを組む必要がある。

- ◇ 「午後2時～5までの時間帯には主婦の視聴者が多い」、
- ◇ 「午後7時～9時までは視聴者が最も多い」

要するに、配信帯域にも複雑な性質を持ち、これらを含めてモデリングする必要がある。これに対して、これらの性質をセルの属性としてモデリングする方法が考えられる。まず、個々のセルに特徴ベクトル $c_i = (ca_{i1}, ca_{i2}, \dots, ca_{ia})$ を定義し、その部分の配信帯域の性質を表すことができる。さらに、個々のデータにも特徴ベクトル $d_i = (da_{i1}, da_{i2}, \dots, da_{ia})$ を定義しておけば、 c_i と d_i のベクトル演算結果に基づいてデータを配置し、配信帯域の性質を考慮したスケジューリングが可能となる。

次に、3. 2に述べたデータモデルでは、「データ A を配信する直後にデータ B を配信してはいけない」や、「同じ論理チャンネルに同じ種類データを続けて配信してはいけない」といったデータの隣接制限がモデリングされていない。さらに、西尾氏らの考慮したデータ間の相関性もモデリングされていない。これらに対しても、ベクトルを用いてモデリングする方法が考えられる。つまり、個々のデータの隣接制限は隣接禁止データの特徴ベクトルを要素とする集合 $\omega_i = \{d_{iw1}, d_{iw2}, \dots, d_{iwi}\}$ を用いてモデル化することが可能である。データ間の相関性もデータの特徴ベクトル間の cosine measure などによる類似度を用いてモデル化することが可能である。

配信帯域のモデルに関しても、データモデルに関しても、ベクトルモデルを採用するのは、モデルの柔軟性を向上させることが狙いである。もちろん、このようなデータモ

デルの拡張に応じて、スケジューリングアルゴリズムの拡張も必要となる。

5.2. アルゴリズムについて

4章に示すアルゴリズムは、最も単純なものである。これを用いて、可能なスケジュールを求めることができるが、最適であるとはかぎらない。例えば、簡単なバックトラックアルゴリズムを導入することによって、4章で示すアルゴリズムで「スケジュール不可」と「宣言」されたデータセットのスケジューリングが可能になったり、配置しきれない中止可能なデータが配置できたりすることが十分に有り得ると考えられる。ただし、バックトラックアルゴリズムの導入に伴う計算量の急増を考慮し、アルゴリズムの効率化が必要と思われる。そもそも、本稿では、このモデルにおける最適化用の評価関数については提示してない。今後の課題として検討する予定があるが、ここでは、いくつかの例を挙げておく。

- ◇ 空きセルの数
- ◇ スケジューリングできたデータの数

一方、このアルゴリズムにおいて、ある時刻に配置可能なデータが複数ある場合、データの選択はランダムに行われているが、これも、選択関数を加えることによって、より最適なスケジュールの算出が可能となる。例えば、データのアクセス確率の大きい順に配置候補から適切なデータを選択する方法が考えられる(本稿では、「データのアクセス確率」がモデリングされていないが、拡張は容易であると思われる)。

6. まとめ

本稿では、デジタル放送を用いたプッシュ型情報配信システムにおける配信スケジューリング方式を提案する。これまでの配信スケジューリングに関する研究では主に配信時間のみを問題にした1次元的なスケジューリング方式が提案されてきたが、我々は、デジタル放送ならではの特徴に着目し、配信時間のみならず、配信帯域幅も考慮した2次元的なスケジューリング方式を考案した。この方式は、従来方式で用いられてきた配信するデータのアクセス確率だけでなく、開始時間の制限や帯域幅の変化など、

より多様なデータ性質を考慮した配信スケジューリングを可能にする。本稿では、考慮すべきデータの性質とそれに基づいたシステムモデル、さらに、このようなモデルにおけるスケジューリングアルゴリズムを設計する際の考え方や方向性を示した。

今後、以下の3点を課題として研究を進める所存である。

- ◇ アルゴリズムの探索
本稿で示した単純なアルゴリズムをベースとして本稿で提案したモデルにおけるより高度なスケジューリングアルゴリズムを考案する。
- ◇ 性能評価
シミュレーションを用いて、本稿で提案した配信スケジューリングモデルおよびアルゴリズムの有効性を示す。
- ◇ 動的配信スケジューリングへの拡張
本稿で想定しているのは、静的な配信スケジューリングであるが、動的な配信スケジューリングにも対応できるモデルを考案する。

7. 参考文献

- [1] 角谷和俊, 宮部義幸:「放送型情報配信のためのモデルとシステム」, 情報処理学会論文誌:データベース, Vol. 40, No.TOD4, 1999.
- [2] 大和田俊和, 浅田一繁, 飯沢篤志, 古瀬一隆:「デジタル放送のためのインデックス情報の断片化方式に関する検討」, 情報処理学会研究報告, 98-DBS-116-29, 1998.
- [3] 西山揚子, 大和田俊和, 浅田一繁, 飯沢篤志, 古瀬一隆:「距離の公理に基づいた情報放送フィルタリング方式」, 情報処理学会研究報告, 99-DBS-119, 1999.
- [4] C. J. Su, L. Tassiulas. "Joint Broadcast Scheduling and User's Cache Management for Efficient Information Delivery", ACM Journal on Wireless Networks, 1999.
- [5] 矢島悦子, 原隆浩, 塚本昌彦, 西尾章治郎:「データ間の相関性を考慮した放送データの送受信方式について」, データベースシステム, 116-13, 1998.
- [6] S. Hameed and N. H. Vaidya. "Log-Time Algorithms for Scheduling Single and Multiple Channel Data Broadcast", MOBICOM'97, Budapest, September, 1997.
- [7] 青野正宏, 辻順一郎, 渡辺尚, 水野忠則:「RAID 型放送システムの提案」, 情報処理学会論文誌, Vol.40, No.7, 1999.
- [8] 「衛星マルチキャスト配信サービス Mega Wave Select」, <http://www.megawave.ne.jp/taiou/mwsrel.html>