

2次元的な放送モデルにおける配信間隔と配信スケジューリング

カン ギョウビ^{†,††} 浅田 一 繁^{†,††}
飯沢 篤志^{†,††} 古瀬 一 隆^{†††}

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、本稿で提案する方式は、デジタル放送ならではの特徴に着目した、配信時間のみならず配信帯域幅も考慮した2次元的なものである。本稿では、このような2次元的なスケジューリングの際に、システムの全体平均応答時間(overall mean response time)が最短となるような、各データの最適配信間隔を求める方式を示す。さらに、求めた配信間隔を守りながら、配信スケジュールを決定するアルゴリズムを提案する。最後に、性能評価を行い、本方式の有効性を示す。

On Broadcast Interval and Scheduling of Two-dimensional Broadcasting Model

XIAOWEI KAN,^{†,††} KAZUSHIGE ASADA,^{†,††} ATSUSHI IIZAWA^{†,††}
and KAZUTAKA FURUSE^{†††}

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 are taken into account, and propose a mechanism truly designed for digital broadcasting. In this paper, we discuss a method to determine the optimal broadcast interval for each data in such kind of two-dimensional scheduling. Moreover, we propose a scheduling algorithm based on the estimated optimal broadcast intervals. Results from performance evaluation verify the effectiveness of the proposed mechanism.

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ともない、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配信システムに関する研究がさかに行われている¹⁾。このような放送型(プッシュ型)情報配信システム(図1)では、サーバ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配信帯域に乗せて周期的に放送し、クライアントは必要なデータのみを配信帯域から選択受信する。具体的な例としては、下記のようなシステムが考えられる。

なシステムが考えられる。

- テマパークの利用者へのスポット、イベント、店舗などの情報配信サービス
- 空港、商店街の利用者への店舗、交通、気象、ニュースなどの情報配信サービス

このような配信システムでは、データを繰り返し配信することにより、各クライアントから離散的に発生した同じデータに対するアクセス要求を、1周期配信す

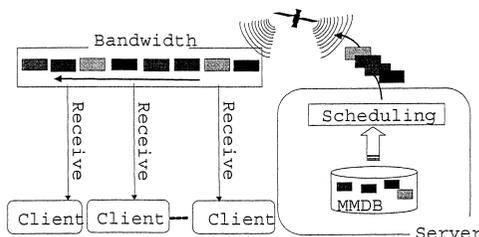


図1 放送型情報配信システム

Fig. 1 Push-based information delivery system.

† 株式会社次世代情報放送システム研究所
Information Broadcasting Laboratories, Inc.

†† 株式会社リコー
RICOH, Co., Ltd.

††† 筑波大学電子・情報工学系
Institute of Information Sciences and Electronics, University of Tsukuba

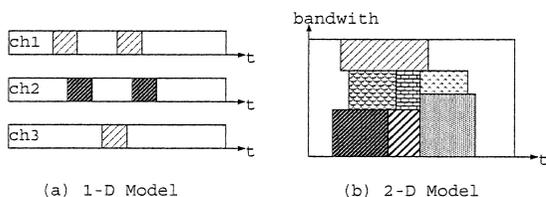


図 2 1 次元的な放送モデル vs. 2 次元的な放送モデル
Fig. 2 1-dimensional broadcasting model vs. 2-dimensional broadcasting model.

るたびにまとめて満たすことができる。このため、クライアント数が増加しても、システム全体の負荷コストがほとんど変わらず、システムのスケーラビリティが優れている。一方、データに対するアクセス要求をまとめて満たすため、各クライアントにとっては、あるデータを要求してからそのデータの配信が開始されるまでの待ち時間の平均値〔本稿では“平均応答時間 (mean response time)”と呼ぶ〕、ならびにすべてのデータの待ち時間の平均値〔本稿では“全体平均応答時間 (overall mean response time)”と呼ぶ〕がシステムの有効性を大きく左右することになる。そこで、この全体平均応答時間を短縮するために、サーバがどのような順番および頻度でデータを配信するかを定める配信スケジューリング方式に関する研究が数多く報告されている^{2)~17)}。

これまでの研究では、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が考えられてきた。すなわち、図 2 (a) に示すように、同じビットレート (配信に要する配信帯域の幅) のデータを時間軸に沿って 1 つずつ逐次的に配信していくという放送モデルにおける配信スケジューリングである。システムの全体平均応答時間を短縮するために、一部の研究では、複数のチャンネルにそれぞれ同じデータセットをずらしながら配信するというマルチチャンネルモデルを提案しているが、すべてのチャンネルの帯域幅および配信データのビットレートが同じであることが仮定されており、これも基本的には 1 次元的なスケジューリングと見なすことができる。

しかし、実際のデジタル放送では、可変ビットレートや多重化の技術によって、複数のデータを並列に配信することや、個々のデータを異なる帯域幅で配信することが可能になっている。たとえば、CS デジタル放送にも採用されている MPEG-2 システムでは、1 つのトランスポートストリーム (transport stream) に複数のデータを時分割多重化し、まとめて 1 つのストリームとして配信する。また、このトランスポートストリームに含まれている個々のデータのビットレ

トは可変である。たとえば、鮮明な映像を配信するには大きな帯域幅を用い、音声のみのデータを配信するときには小さな帯域幅を用いる、といったことができる。受信機側は DMUX (Demultiplexer, 分離部) を用いて、同時に配信してきた個々のデータを識別・分離し、受信者にとって必要なデータを選択受信する。したがって、実際のデジタル放送における配信スケジューリングは、図 2 (b) で示されているような 2 次元的な放送モデルに基づくことになる。このモデルでは、データを 1 つずつ逐次的に配信するのではなく、複数のデータを並行に配信することができるので、用途やデータの内容にあわせて柔軟かつ効率良くデータを配信することが可能である。

なお、デジタル放送を用いたシステムの受信形態としては、受信したデータを受信機に蓄積して利用する蓄積型と、受信しながら再生するストリーミング型がある。本研究が対象としているのは後者である。DirecPC¹⁸⁾ のようなシステムでは、固定端末が受信機となるため、蓄積型受信形態が一般に用いられている。これに対して、本研究では、利用者が受信機を持ち歩きながらマルチメディアデータを受信するようなシステムを対象としている。携帯端末には限られている容量のメモリしか内蔵できないので、受信形態としてはストリーミング型を用いるのが一般的である。

筆者らはこれまでに、このような情報配信サービスシステムおよびデジタル放送ならではの特徴に着目し、配信時間と配信帯域をともに考慮する 2 次元的な放送モデルを提案した¹⁹⁾。本稿では、このような 2 次元的な放送モデルにおいて、全体平均応答時間を短縮するには個々のデータをどのような間隔で配信すべきかについて議論し、最適な配信間隔を算出する計算式を示す。この配信間隔は、配信するデータのビットレートやアクセス確率などを元に算出されるものであり、すべてのデータについてこの配信間隔を守って配信を行えば、理論上全体平均応答時間が最も小さくなる。しかし、実際のデータを用いてスケジューリングを行う際には、すべてのデータについて最適な配信間隔を守るように配置するのは実質的に不可能である。そこで、本稿では、この配信間隔をできるだけ守りながら配信を行うようなスケジューリングを決定するアルゴリズムを提案し、このスケジューリングの有効性について、性能評価を用いた検証を行う。

本稿は以下のように構成される。まず、2 章におい

2000 年 12 月に NTT ドコモが開始した映像配信サービス “M-stage visual” で採用されていた動画/静止画兼用の閲覧端末 “eggy” はその一例である。

て関連研究について述べる。3章では、2次元的な配信スケジューリングモデルを定義し、そのモデルにおける最適な配信間隔の計算式を提案する。4章では、上記の配信間隔を用いて具体的に配信スケジュールを組むためのアルゴリズムを示す。5章では、本提案の有効性を評価する。最後に、今後の課題について述べる。

2. 関連研究

80年代初期に放送型情報配信システム Teletex^{2)~4)}が提案されて以来、配信スケジューリングに関する研究がはさかに行われるようになった⁵⁾。これまでに提案された方式は、配信順序や頻度を定める基準によって、大きく2種類に分けることができる。

(1) 利用者からの個々のアクセス要求をベースにする^{9),11),12)}。

たとえば、たくさんのアクセス要求が寄せられるデータや待ち時間が長いデータは、より優先的に、頻繁に配信する。これはすべてのクライアントからサーバへアップリンクがあることを前提とする。個々のクライアントの利用者はこのアップリンクを通して、自分のアクセス要求をサーバへ送る。

(2) データへのアクセス確率の予測をベースにする^{10),13),15)}。

たとえば、アクセス確率の高い、つまり頻繁にアクセスされるだろうと予測されるデータを優先的に、頻繁に配信する。逆に、ほとんどアクセスされないだろうと思われるデータは低い頻度で配信する。この方式ではアップリンクは不要である。

本稿で提案する方式は後者に分類される。

データのアクセス確率に基づく配信スケジューリングアルゴリズムは Gecsei によって提案された²⁾。Gecsei は、データ i の配信間隔 S_i を下記の式 (1) で算出する値としたときに最小全体平均応答時間を得られることを示した。

$$s_i = \left(\sum_{j=1}^N \sqrt{p_j} \right) \frac{1}{\sqrt{p_i}} \quad (1)$$

ここで、 p_i はデータ i のアクセス確率を表し、 N は配信するデータの総数を表す。これは、アクセス確率の高いデータ(人気のあるデータ)はより短い間隔で、より頻繁に放送されるということの意味する。

式 (1) では、データはすべて同じ大きさであることが前提となっている。しかし、実際の配信システムにおけるデータはすべて同じ大きさとは限らない。そこ

で、Hameed らは式 (1) を可変長のデータまで拡張した(式 (2))⁰⁾。

$$s_i = \left(\sum_{j=1}^N \sqrt{p_j l_j} \right) \sqrt{\frac{l_i}{p_i}} \quad (2)$$

ここで、 l_i はデータ i の長さを表し、その他は式 (1) と同じ意味を持つ。すなわち、アクセス確率が高く、かつ放送所要時間が短いデータはより頻繁に放送するということになる。

さらに、Hameed らはこの式を配信失敗率を考慮するように拡張した(式 (3))⁰⁾。

$$s_i = \left(\sum_{j=1}^N \sqrt{p_j l_j \left(\frac{1+E(j)}{1-E(j)} \right)} \right) \times \sqrt{\frac{l_i}{p_i} \left(\frac{1-E(i)}{1+E(i)} \right)} \quad (3)$$

ここでは、 $E(i)$ はデータ i の配信失敗率を表し、その他は式 (2) と同じ意味を持つ。つまり、アクセス確率が高くかつ放送に所要する時間が短いデータは、配信失敗率が大きければ大きいほどより短い間隔で、頻繁に放送する。しかし、Hameed らの研究ではデータの更新を考慮していない。

青野らは、配信するデータの更新率を考慮するように、配信間隔の算出式を提案した(式 (4))⁵⁾。

$$s_i = \left(\sum_{j=1}^N p_j^\alpha q_j^\beta \right) \frac{1}{p_i^\alpha q_i^\beta} \quad (4)$$

ここで、 q_i はデータ i の更新確率を表す。また、 α と β はそれぞれ p_i と q_i に関するべき乗指数であり、更新確率とアクセス確率の影響を制御する。すなわち、頻繁に更新されかつアクセス確率の高いデータほどより短い間隔で、頻繁に放送することによって、全体平均応答時間を短縮するとともに、更新の伝播遅延を短縮することを図る。ちなみに、 $\alpha = 0.5$ かつ $\beta = 0$ (これは、更新率の影響を考慮しないことを意味する) の場合、この式は式 (1) と等しくなる。

前述のように、これらの研究は各データを同じビットレートで逐次的に配信することを前提にした、1次元的な配信スケジューリングである。しかし、2次元的な放送モデルでは、各データのビットレートは必ずしも等しくなく、かつ、データが並列的に配信されるため、従来研究で得られたこれらの配信間隔の算出式をそのまま適用することができない。したがって、2次元的な配信スケジューリングにあわせた拡張が必要となる。

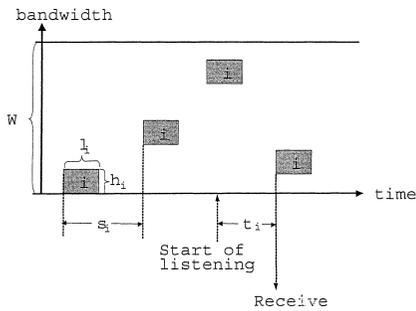


図3 放送モデル

Fig.3 Broadcasting model.

3. 最適な配信間隔

本章では、2次元的な配信スケジューリングにおいて、システム全体平均応答時間を最小にする最適な配信間隔算出方法について述べる。

3.1 放送モデル

まず、本研究における放送モデルを定義する。

配信帯域は図3に示す2次元空間で表す。横軸 *time* は配信時間であり、単位配信時間の離散値をとる。縦軸 *bandwidth* は配信帯域幅であり、単位配信帯域幅の離散値をとる。配信しようとするデータの総数は N 個とする。個々のデータの配信に必要な単位配信時間数は、このデータの長さ l_i とし、配信に必要な単位配信帯域幅はこのデータの高さ h_i とする。異なるデータの長さおよび高さは同じである必要がない。たとえば、MPEG-2の動画を配信する場合、データの高さは5 Mpbsが必要であるが、MP3の音楽ならデータの高さは192 kpbsで十分である。ただし、データの長さと高さは配信時間にもなう変化はないとする。すなわち、 l_i と h_i は配信時刻 t の関数ではなく、定数とする。DirectPCのような典型的なデータ放送システムでは、蓄積型の受信形態を前提としているため、ソフトウェアのようなデータはもちろん、動画や音声データでも、長さおよび高さが可変である。しかし、1章でも説明したように、本稿の想定する放送システムでは、ストリーミング型の受信形態を前提としているため、動画や音声のデータをスムーズに再生するには、一定のビットレートによる配信が必要である。データ i の配信間隔 s_i は、 k 回目の配信開始時刻から $k+1$ 回目の配信開始時刻までの単位配信時間数である。配信中、データへのアクセス要求はランダムに発生すると仮定する。データ i へのアクセス確率は p_i とし、 $\sum_{i=1}^N p_i = 1$ である。各 p_i は事前調査によって入手し、配信の最中にこのアクセス確率は変化しないものとする。データ i の平均応答時間

(mean response time) t_i は、データ i のアクセス要求が発生した時刻から、すなわち、利用者が受信を開始してから実際にデータ i の配信が開始されるまでの単位配信時間数の平均値とする。また、システムの全体平均応答時間(overall mean response time)は $\sum_{i=1}^N p_i t_i$ とする。本稿では、単純化のため、配信失敗および更新については考慮しない。

3.2 配信間隔の最適値

前述のような2次元的な放送モデルにおいて、各データをどのような間隔で配信すればシステムの全体平均応答時間が最も短くなるか、すなわち、各データの配信間隔の最適値はどのように算出できるかという問題については、これまでの研究では提案されなかった。本稿では、Hameedらのモデル¹⁰⁾をベースに、1次元的な放送モデルにおける成果を上記の2次元的な放送モデルに拡張した。

各データの配信間隔 s_i をつねに同じであるように維持すれば最適な全体平均応答時間が得られることは、従来の研究によって明らかにされている^{7)~9),15)}。実際には、すべてのデータをつねに一定の配信間隔を保ちながら配信するのは容易ではないが、ここでは最適な配信間隔の計算式を導出するために、理論上各データが等間隔で配信されると仮定する。

最適な配信間隔を決める基本的な考え方は、「アクセス確率の高いデータほど頻繁に配信する」「放送帯域を消費しないデータほど頻繁に配信する」という点にある。このことより、次が導き出される。

定理1 各データが等間隔で配信されると仮定する。すべてのデータを式(5)で算出される配信間隔で配信するとき、全体平均応答時間は最小となる。

$$s_i = \frac{1}{W} \left(\sum_{j=1}^N \sqrt{p_j l_j h_j} \right) \sqrt{\frac{l_i h_i}{p_i}} \quad (5)$$

ここで、 p_i はデータ i のアクセス確率を表し、 W は配信帯域の幅を表す。すなわち、アクセス確率が高くかつサイズの小さいデータをより短い間隔で頻繁に放送する。定理1の証明は付録に示す。

データのアクセス要求がランダムに発生すると仮定すれば、各データの平均応答時間 t_i は配信間隔 s_i の半分となる。すなわち、 $t_i = s_i/2$ 。したがって、システムの全体平均応答時間 $t_{overall}$ は次の式(6)で計算できる。

$$t_{overall} = \sum_{i=1}^N p_i t_i = \frac{1}{2} \sum_{i=1}^N p_i s_i \quad (6)$$

式(5)を式(6)を代入すると、システムの全体平均

応答時間の最適値 $t_{optimal}$ が得られる．

$$t_{optimal} = \frac{1}{2W} \left(\sum_{i=1}^N \sqrt{p_i l_i h_i} \right)^2 \quad (7)$$

前述のように，すべてのデータをつねに各自の最適な配信間隔 s_i を保ちながら配信するのは現実には不可能である．したがって，式 (7) は，全体平均応答時間の下限を与える．実際にデータを配信する際には，各自の配信間隔を s_i に保てば保つほど，全体平均応答時間はこの最適値に近づく．

4. スケジューリングアルゴリズム

各データの最適な配信間隔を式 (5) によって求め，なるべくこの値に近くなるように配信すべきデータを選定すれば，式 (7) の $t_{optimal}$ に近い全体平均応答時間が得られる．本章では，式 (5) で算出した配信間隔をできるだけ保ちながら，配信すべきデータを選定するスケジューリングアルゴリズムを示す．

一般的には，配送中にリアルタイムに次の配信データを決める（動的配信スケジューリング）方式と，配信前に配信スケジュールをすべて決める（静的配信スケジューリング）方式との2種類がある．動的配信スケジューリングは，柔軟な配信スケジュールを可能とする代わりに，受信機に，いつ，どのデータが配信されるかが不明なため，必要なデータを入手するまで，電源を入れ続けて受信し続けるしかない，という欠点がある．一方，静的配信スケジューリングは，必要なデータが配信される時刻にあわせて受信機の電源を入れ，受信を開始できるので，モバイル環境において消費電力を低減できるという面に優れている．本稿で提案する方式は，配信帯域幅，配信するデータの h_i ， l_i ， p_i ，配信するデータ総数 N はすべてスケジューリング時に決められていることを前提とする．この前提が成り立つ場合，本方式は動的配信スケジューリング方式にも，静的な配信スケジューリング方式にも適用できる．

ここで示すアルゴリズムの基本的な考え方は，「スケジューリング時刻 t において，最適な配信間隔を保つために次に配信すべきデータを選び，そのデータを配信するのに必要な帯域が確保できる最も早い時刻 $t' (\geq t)$ に配信する」というものである．そして，選んだデータの次回に配信すべき時刻として，そのデータの最適配信間隔 s_i を加算した値をとることによって，最適な配信間隔を保つという手法を用いる．ただし，本稿の用いている2次元的な配信モデルでは，データによって配信に必要な帯域幅（高さ）が異なるため，

時刻 t の段階で残っている未使用の帯域幅では次に配信すべきデータを配信できない場合が生じる可能性がある．時刻 t に放送できないデータを除外して次に配信するデータを選定してしまうと，いつまでも配信できないデータが存在してしまう可能性がある．このような状況を避けるため，最も早く配信すべきものとして選定されたデータを配信するために必要な配信帯域が時刻 t では足りない場合， t より先の最も早く配信できる時刻 $t' (t \leq t')$ の配信帯域を予約するというアプローチをとっている．

アルゴリズムの入力は， W ， N ， p_i ， l_i および h_i である．アルゴリズムの出力，すなわち，配信スケジュールは，データの識別子と配信する時刻の組の集合 $S \subseteq \{(i, t) | 1 \leq i \leq N, 0 \leq t\}$ で表す．なお，スケジューリングアルゴリズムでは，以下の変数を用いる．

- t : スケジューリングの現在時刻
- w_t : 時刻 t に未使用の配信帯域幅
- α_i : 次にデータ i の配信をはじめめるべき時刻
- β_i : $\beta_i = \alpha_i + s_i$
- R_t : 時刻 t に配信すべきデータの集合

図4にスケジューリングアルゴリズムを示す．このアルゴリズムの概要は，以下のとおりである．

まず，(1) で各データのアクセス確率 p_i ，長さ l_i ，および高さ h_i を用いて，式 (5) に従って，このデータの最適配信間隔 s_i を算出する．また，(2)～(4) において各変数の初期化を行う．

次に，(5) で， $\alpha_i \leq t$ となるデータの集合 R_t を求める． R_t が空であれば，現在時刻 t を単位時間分進め，集合 R_t を再度求める．

R_t が空でなければ，すなわち，時刻 t において選択可能なデータが存在すれば，(7)～(8) において R_t の中で最も β_i の値が小さいものを1つ選び， j とする（そのような j が複数存在する場合には，それらのうちの1つを任意に選択する）．

(9) では，データ j を配信するために必要な帯域が確保できる最も早い時刻 t' を求める．(10) では，この t' を用いて (j, t') をスケジュールとして出力する．

続いて，(11) において，データ j の配信で消費した分の帯域を w_t から減ずる．

最後に，(12) において，データ j の次に配信すべき時刻を設定し，(5) に戻る．

5. 性能評価

本方式の有効性を検証するため，1次元的な配信スケジューリングと2次元的な配信スケジューリングとの比較や，システムの各種パラメータが全体平均応答

- (1) determine optimal spacing s_i , for each i , using Equation (5)
- (2) $S \leftarrow \phi$
- (3) $\alpha_i \leftarrow 0; \beta_i \leftarrow s_i$, for each i
- (4) $w_t \leftarrow W$, for $t \geq 0$
- (5) $R_t \leftarrow \{i \mid 1 \leq i \leq N, \alpha_i \leq t\}$
- (6) if $R_t = \phi$, $t \leftarrow t + 1$; goto (5)
- (7) $\beta_{\min} \leftarrow \min_{i \in R_t} \beta_i$
- (8) $j \leftarrow i \in R_t$ such that $\beta_i = \beta_{\min}$
- (9) $t' \leftarrow \min(\{t'' \mid t'' \geq t, \text{for all } \hat{t} \text{ such that } t'' \leq \hat{t} \leq t'' + l_j, w_{\hat{t}} \geq h_j\})$
- (10) $S \leftarrow S \cup \{(j, t')\}$
- (11) $w_i \leftarrow w_i - h_j$, for each \hat{t} such that $t' \leq \hat{t} \leq t' + l_j$
- (12) $\alpha_j \leftarrow \beta_j; \beta_j \leftarrow \alpha_j + s_j$
- (13) goto (5)

図4 スケジューリングアルゴリズム

Fig.4 Scheduling algorithm.

時間に対する影響など、2次元方式自身の性質を示すことによって性能評価を行う。1次元的な方式との比較対象としては、Hameedらの方式¹⁰⁾(3チャンネル)を採用する。評価用の配信データには自動生成した疑似データを用いる。評価では、まず各種のアルゴリズムを用いて、それぞれの配信スケジュールを算出する。算出された配信スケジュールにおける実際の全体平均応答時間は確率計算によって求める。

5.1 評価モデル

性能評価に用いたシステムの各種パラメータを以下に示す。

- 配信するデータの数 N を 100 とし、各データの長さ l_i は 1~10 (単位時間) の一様分布に従ってランダムに選定する。各データの長さ h_i は平均 5, 分散 1 の正規分布に従ってランダムに選定する。ただし、正值のみを採用する。
- 配信帯域の幅 W は 30 単位配信帯域幅とし、スケジューリング期間 T は 1,000,000 単位時間とする。
- 利用者のデータへのアクセス確率 p_i は次の Zipf 分布²⁰⁾ に従うものとする。

$$p_i = \frac{(1/i)^\theta}{\sum_{i=1}^N (1/i)^\theta}, \quad 1 \leq i \leq N \quad (8)$$

ここで、 θ は分布の偏りを制御するために用いる。 θ の値が大きくなるほど、アクセス確率の偏りは大きくなる。デフォルト値は 0.5 とする。Zipf 分布はデータのアクセス確率を表現する分布として適していると考えられており、多くの従来研究に利用されている^{5),7),9),10)}。

- 比較対象とする Hameed らの方式に関しては、利用者はすべてのチャンネルを受信するものとする。つまり $\pi_1 = \pi_2 = 0, \pi_3 = 1$ とする。なお、2次元方式と同じ帯域幅を利用する前提で比較するために、3つのチャンネル幅は各々10単位配信帯域幅とする。その他のパラメータは2次元方式と同様とする。

このパラメータ設定に基づいて決定された配信スケジュールにおける全体平均応答時間は、下記のように確率計算で算出することにする。まず、仮に、スケジュール期間 T の間にデータ i ($i = 1, 2, \dots, N$) が配信された回数を k_i とする。そして、 j 番目の配信開始時刻から $j+1$ 番目の配信開始までの間の単位配信時間数(つまり、 j 番目の配信間隔)を s_{ij} ($j = 1, 2, \dots, k_i$) とすると、 $\sum_{j=1}^{k_i} (s_{ij}) = T$ である。次に、データへのアクセスがランダムに発生するものとする、データ i に対する任意のアクセスが s_{ij} の間に発生する確率は s_{ij}/T である。したがって、この s_{ij} に入ったデータ i に対するすべてのアクセスについて、次にデータ i が配信されるまでの平均時間は $s_{ij}/2$ である。これにより、すべての s_{ij} 、すなわち、全スケジュール期間中のデータ i に対するアクセスの平均待ち時間 t_i は、式 (9) で算出することができる。

$$t_i = \sum_{j=1}^{k_i} \left(\frac{s_{ij}}{T} \right) \left(\frac{s_{ij}}{2} \right) = \frac{1}{2T} \sum_{j=1}^{k_i} (s_{ij})^2 \quad (9)$$

したがって、全体平均応答時間 $t_{overall}$ は式 (10) で算出することができる。

$$\begin{aligned}
 t_{\text{overall}} &= \sum_{i=1}^N (p_i t_i) \\
 &= \sum_{i=1}^N p_i \left(\frac{1}{2T} \sum_{j=1}^{k_i} (s_{ij})^2 \right) \\
 &= \frac{1}{2T} \sum_{i=1}^N p_i \sum_{j=1}^{k_i} (s_{ij})^2 \quad (10)
 \end{aligned}$$

ここで、 p_i はデータ i に対するアクセス確率であり、前述のように式 (8) で算出する。なお、すべてのデータが等間隔で配信される場合、つまり $\forall s_{ij}, s_{ij} = T/k_i = s_i$ なら、この式 (10) は式 (6) と等価である。

5.2 1次元方式と2次元方式の比較

まず、1次元的なスケジューリング方式(以下「1次元方式」という)と2次元的なスケジューリング方式(以下「2次元方式」という)を比べることによって、2次元方式の有効性を評価する。

図5は両者の比較結果を示す。横軸は式(8)のZipf分布における θ の値を表し、縦軸は全体平均応答時間を表しており、式(10)による計算結果である。つまり、この図では、上に行くほど、全体平均待ち時間が長く、性能が悪いことを意味する。なお、1次元方式の実際の全体平均応答時間と理論上の最適全体平均応答時間を、それぞれ“1D”および“1D_opt”と表している。2次元方式については、それぞれ“2D”および“2D_opt”と表している。また、各データの高さをすべて10に固定した場合の2次元方式の実際の全体平均時間と最適値を、それぞれ“2D(CBR)”および“2D_opt(CBR)”と表している。

この結果から、1次元方式に比べて、2次元方式の全体平均応答時間は30~50%程度短縮されることが分かる。パラメータとして用いているデータの高さは平均5、分散1の正規分布であるので、ほとんどのデータの高さは10未満となる。配信帯域幅 W を30と設定しているので、2次元方式では、帯域幅に余裕があれば同時に4つ以上のデータを配信することができるが、1次元マルチチャンネル方式では、チャンネル数以上のデータを同時に配信することはできないので、当然のことながら2次元方式の方が有利となる。もちろん、データの高さが10に近づくにつれ、1次元方式における配信帯域(各チャンネル10単位配信帯域幅)の空きが少なくなり、2次元方式における効率的な配置ができにくくなるため、両者の全体応答時間の差は小さくなる。しかし、“2D(CBR)”と“2D_opt(CBR)”から分かるように、各データの高さがすべて10である場合でも、2次元方式は1次元方式とほぼ同じ全体

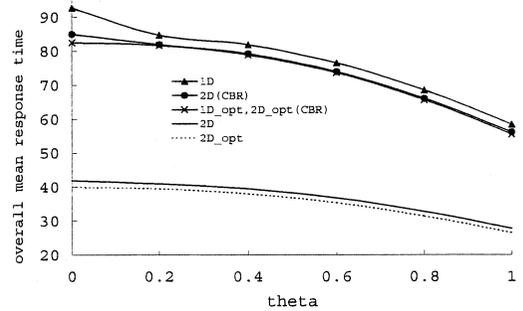


図5 1次元方式 vs. 2次元方式

Fig. 5 One-dimensional vs. two-dimensional.

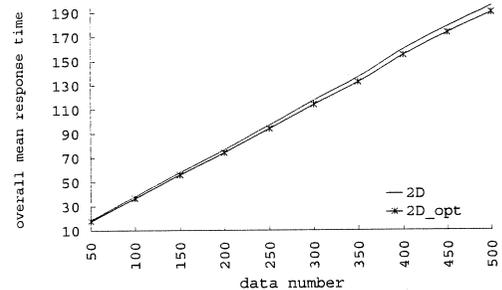


図6 配信するデータ数の影響

Fig. 6 Effects of the data number.

平均応答時間を得られている。したがって、固定ビットレートの場合でも、本稿の提案する2次元方式が既存の1次元方式よりも性能上劣るということはない。

5.3 2次元方式の性質

システムの各種パラメータが全体平均応答時間に及ぼす影響など、2次元方式自身の性質について調べる。まず、利用者のアクセスパターンと全体平均応答時間の関係については、図5に示したように、 θ の値の増加、つまりデータへアクセスの偏りの増加につれ、全体平均応答時間が短くなることが分かる。

次に、配信するデータの個数と全体平均応答時間の関係について調べる。図6は配信データの数が50~500まで変化する場合の全体平均応答時間を示している。横軸は配信するデータの個数を表し、縦軸は全体平均応答時間を表す。図には、式(10)で算出した実際の全体平均応答時間“2D”と、式(7)で算出した最適全体平均応答時間“2D_opt”が示されている。この結果から分かるように、配信するデータ個数の増加にともない、全体平均応答時間は線形的に増加する。なお、データの個数の増加にともない、実際値と最適値の差も線形的に広がる傾向にある。

さらに、配信するデータの高さと全体平均応答時間の変化について調べる。この結果を図7に示す。横軸

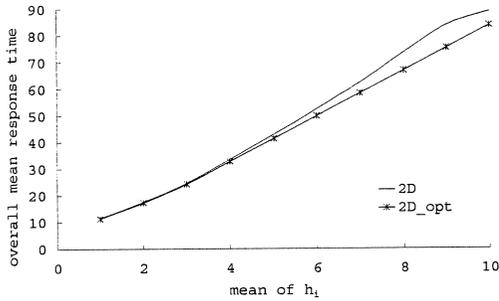


図7 配信するデータ高さの影響

Fig. 7 Effects of the bandwidth for broadcasting a data.

は配信するデータの高さ h_i を生成する際に使う正規分布の平均値, 縦軸は全体平均応答時間を表す. 正規分布の分散を 1 とするとき算出した実際の全体応答時間と理論上の最適値は, それぞれ “2D” と “2D_opt” で表されている. この結果から分かるように, 全体の配信帯域幅が変わらない場合, データの高さと所要全体応答時間がほぼ比例する. いずれの場合においても, 提案するスケジューリング方式は理論上の最適な全体平均応答時間に近い全体平均応答時間を実現している.

本来, すべてのデータをつねに式 (5) で算出した各自の最適配信間隔 s_i で配信できれば, 全体平均応答時間は式 (7) に示す最適全体平均応答時間になるはずである. しかし, 一般的にいえば, すでに配置したデータの長さや高さの関係で, すべてのデータをつねに最適配信間隔で配信することは不可能である. このため, 全体平均待ち時間と最適全体平均時間との間につねに差が生じる. さらに詳細に分析すると, この差の生じる原因は 2 つに分けることができる (1) 実際の配信間隔が最適値からずれている (2) 配信間隔が周期的になっていない (1) については, すべてのデータがその最適配信間隔より s ずれるとすると, 式 (6) は, $t_{overall} = \sum_{i=1}^N p_i(s_i + s)$ となる. したがって, ずれ s の増加にともない, 全体平均応答時間 $t_{optimal}$ は線形的に増加することが分かる (2) については, 実際の配信間隔 s_{ij} の分散と平均応答時間の伸びの関係を図 8 に示す. 横軸は各データの実際の配信間隔 s_{ij} の分散 $\sigma^2 = \frac{1}{k_i - 1} \sum_{j=1}^{k_i} (s_{ij} - \frac{T}{k_i})^2$ を表す. 縦軸は, 式 (9) で算出した実際の応答時間 t_i と最適応答時間 t_{iopt} (つまり, $s_{ij} = s_i$ の場合 t_i の値) の差が t_{iopt} に対する割合 $\delta = (t_i - t_{iopt})/t_{iopt}$ を表す. この図から分かるように, 90% のデータの実際の配信間隔の分散は 17~40 の間であり, 40 を超えるものは 10% しかない. さらに, 実際の配信間隔の分散が 40 までの間では, 分散が大きくなるにともない, そのデータの平均応答時間の伸びが緩やかに増加することが分かる.

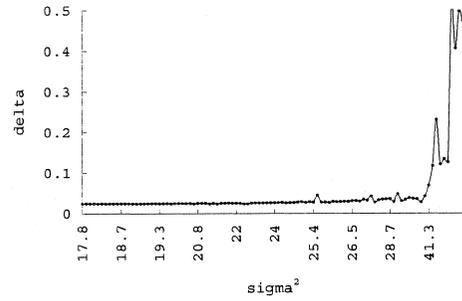


図8 配信間隔の分散の影響

Fig. 8 Effects of the dispersion of broadcasting interval.

分散が 40 以上になると, 平均応答時間の伸びが急速に増加するのは, その 10% のデータの分散の差が大きいのが原因である.

6. まとめ

本稿では, 2次元的な配信モデルにおける最適な配信間隔を示すとともに, それを用いたスケジューリングアルゴリズムを提案した. 従来の 1次元的なモデルとは異なり, 本稿のモデルでは異なるビットレートを持つデータを配信可能にしている. 性能評価の結果により, 本稿の提案方式が理論上の最適値に近い性能を実現することを確認した. この方式は, デジタル放送における可変ビットレート, すなわちデータの高さが不均一であるような場合に有効であると考えられる.

提案した方式は 2次元配信モデルにおけるスケジューリングの基本的な機能を実現していると考えられるが, より実的なものとするためにはさまざまな課題がある. たとえば, 本稿で提案した配置モデルでは配信の失敗やデータの更新については考慮していないが, 実際の配信システムでは応用によってはこれらを考慮することがシステムの有効性に影響することもあると思われるので, 配信モデルの拡張が必要である.

本提案の配信モデルでは, 個々のデータに対するアクセスについては独立した事象として扱っているが, 実際に配信されるデータの間には何らかの相関関係が存在する 경우가多い. 文献 14) に提案されているように, データ間の相関を考慮し, 関連のあるデータをまとめて配信する方式は, 2次元配信モデルにおいても有効であると考えられる. また, ハイパーメディアデータを配信する場合, ハイパーリンクを順にたどるようなアクセスが考えられるので, これを考慮に入れた配信モデルを構築することも, 今後の課題である.

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.SIG5(TOD2), pp.141-157 (1999).
- 2) Gecsei, J.: *The Architecture of Videotex System*, Prentice-Hall, Englewood Cliffs, NJ (1983).
- 3) Alber, A.F.: *Video/Teletex: Principles and Practices*, McGraw-Hill, NY, New York (1985).
- 4) Storey, J.R., Vincent, A. and Fitzgerald, R.: A Description of the Broadcast Telidon System, *IEEE Trans. Consumer Electron*, Vol.CE-26, No.3 (1980).
- 5) Wong, J.W.: Broadcast Delivery, *Proc. IEEE*, Vol. 76, No.12, pp.1566-1577 (1988).
- 6) Imielinski, T., et al.: Energy Efficient Indexing on Air, *Proc. ACM SIGMOD Conference*, Minneapolis, MN, pp.25-36, ACM (1994).
- 7) Acharya, S., Alonso, R., Franklin, M. and Zdonik, S.: Broadcast Disks: Data Management for Asymmetric Communication Environments, *Proc. ACM SIGMOD Conference*, San Jose, CA, pp.199-210, ACM (1994).
- 8) Acharya, S., Franklin, M. and Zdonik, S.: Dissemination-based data delivery using broadcast disks, *IEEE Personal Communication*, Vol.2, No.6, pp.50-60 (1995).
- 9) Su, C.-J. and Tassiulas, L.: Broadcast Scheduling for Information Distribution, *Proc. INFOCOM'97* (1997).
- 10) Hameed, S. and Vaidya, N.H.: Efficient Algorithms for Scheduling Data Broadcast, *ACM Wireless and Network* (1999).
- 11) Dykman, H.D., Ammar, M. and Wong, J.W.: Scheduling Algorithms for Videotex Systems Under Broadcast Delivery, Technical Report GIT-ICS-88/25, Georgia Institute of Technology (1988).
- 12) Aksoy, D. and Franklin, M.: RxW: A Scheduling Approach for Large-Scale On-Demand Data Broadcast, *IEEE/ACM Trans. Networking*, Vol.7, No.6, pp.846-860 (1999).
- 13) 矢島悦子, 原 隆浩, 塚本昌彦, 西尾章治郎: 相関性を持つデータ間の放送時間間隔について, 情報処理学会論文誌, Vol.40, No.1, pp.188-196 (1999).
- 14) 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スケジューリング法およびキャッシング法, 情報処理学会論文誌, Vol.40, No.9, pp.3577-3585 (1999).
- 15) 青野正宏, 田窪昭夫, 渡辺 尚, 水野忠則: データ放送におけるスケジューリング決定法「二重循環法」の提案と評価, 情報処理学会論文誌, Vol.40, No.3, pp.1267-1275 (1999).
- 16) 青野正宏, 渡辺 尚, 水野忠則: プッシュ型とプル型通信の動的統合による応答時間の短縮, 情報処理学会 MBL 研究会, モバイルコンピューティング 10-2, pp.9-16 (1999).
- 17) 箱森 聡, 田辺雅則, 石川裕治, 井上 潮: 放送型通信とオンデマンド型通信を統合した情報提供システム, 情報処理学会論文誌, Vol.40, No.10, pp.3772-3781 (1999).
- 18) DirecPC: <http://www.direcpc.com> (1996).
- 19) カンギョウビ, 浅田一繁, 白田由香利, 飯沢篤志, 古瀬一隆: 情報の性質を考慮した放送スケジューリング方式, 情報処理学会データベースシステム研究報告, 2000-DBS-122, pp.89-96 (2000).
- 20) Zipf, G.K.: *Human Behavior and the Principle of Least Effort Reading*, Addison-Wesley, MA (1949).

付 録

【定理 1 の証明】

$l_i h_i / s_i = r_i$ とおく. すると, 式 (6) は次の式 (11) に書き換えることができる.

$$t = \frac{1}{2} \sum_{i=1}^N \frac{p_i l_i h_i}{r_i} \quad (11)$$

各データが等間隔で配信されると仮定した場合, r_i はデータ i に対して割り当てられた最適な帯域幅と見なすことができるので, $\sum_{i=1}^N r_i = W$ となる. よって, N 個の r_i のうち, $N-1$ 個だけが独立に変化可能ということになる.

r_i が t を最小とする値であるためには, $\forall i, \frac{\partial t}{\partial r_i} = 0$ である必要がある. よって, 以下の関係が成り立つ.

$$\begin{aligned} 0 &= \frac{\partial t}{\partial r_1} = \frac{1}{2} \frac{\partial}{\partial r_1} \left(\sum_{i=1}^N \frac{p_i l_i h_i}{r_i} \right) \\ &= \frac{1}{2} \frac{\partial}{\partial r_1} \left(\frac{p_1 l_1 h_1}{r_1} + \sum_{i=2}^{N-1} \frac{p_i l_i h_i}{r_i} + \frac{p_N l_N h_N}{(W - \sum_{i=1}^{N-1} r_i)} \right) \\ &= \frac{1}{2} \left(-\frac{p_1 l_1 h_1}{r_1^2} + \frac{p_N l_N h_N}{(W - \sum_{i=1}^{N-1} r_i)^2} \right) \end{aligned}$$

これより, 次の式 (12) が得られる.

$$\frac{p_1 l_1 h_1}{r_1^2} = \frac{p_N l_N h_N}{(W - \sum_{i=1}^{N-1} r_i)^2} \quad (12)$$

同様にして, 式 (13) が得られる.

$$\frac{p_2 l_2 h_2}{r_2^2} = \frac{p_N l_N h_N}{(W - \sum_{i=1}^{N-1} r_i)^2} \quad (13)$$

式(12)と式(13)より, 次の関係が成り立つ.

$$\frac{p_1 l_1 h_1}{r_1^2} = \frac{p_2 l_2 h_2}{r_2^2} \implies \frac{r_1}{r_2} = \sqrt{\frac{p_1 l_1 h_1}{p_2 l_2 h_2}}$$

したがって, 任意の i, j について, 次の式が成り立つ.

$$\frac{r_i}{r_j} = \sqrt{\frac{p_i l_i h_i}{p_j l_j h_j}}$$

これは, r_i が $\sqrt{p_i l_i h_i}$ に比例することを意味する. すなわち, $r_i = \alpha \sqrt{p_i l_i h_i}$ である. ここで,

$$\alpha = \frac{W}{\sum_{j=1}^N \sqrt{p_j l_j h_j}} \quad (14)$$

である場合に限り, $\forall i \frac{\partial t}{\partial r_i} = 0$ かつ $\sum_i r_i = W$ が成り立つ. なお, この問題の性質から, t の非負の最小値が必ず存在する. したがって, 上記の α は最小値 t を得られる唯一の解である. 式(14)と $r_i = \alpha \sqrt{p_i l_i h_i}$ を $l_i h_i / s_i = r_i$ に代入すれば, 式(5)が得られる.

(平成 13 年 4 月 7 日受付)

(平成 13 年 6 月 13 日採録)



カン ギョウビ (正会員)

1992 年筑波大学工学研究科博士課程後期を中退し, 筑波大学電子・情報工学系準研究員. 1993 年(株)リコー入社, ソフトウェア研究所でデータベース・システムの研究開発に従事. 現在に至る. 1999 年から(株)次世代情報放送システム研究所へ兼任出向し, 放送機構を用いた超大規模データベースシステムの研究開発に従事. IEEE 会員. 博士(工学).



浅田 一繁

1987 年早稲田大学理工学部電気工学科卒業. 同年(株)リコー入社. 現在同社ソフトウェア研究所に勤務. データベースシステムの研究に従事. 1997 年から(株)次世代情報放送システム研究所へ兼任出向し, 放送機構を用いた超大規模データベースシステムの研究開発に従事.



飯沢 篤志 (正会員)

1982 年東京大学大学院理学系研究科情報科学専門課程修士課程修了. 1985 年(株)リコー入社. 現在同社ソフトウェア研究所に勤務. データベース・システムの研究に従事. 1997 年から(株)次世代情報放送システム研究所へ兼任出向し, 放送機構を用いた超大規模データベースシステムの研究開発に従事. 電子情報通信学会, ソフトウェア科学学会, IEEE 各会員.



古瀬 一隆 (正会員)

1993 年筑波大学大学院工学研究科修了. 同年(株)リコーソフトウェア研究所. 1996 年茨城大学工学部情報工学科助手. 1999 年筑波大学電子・情報工学系助手. データベースシステムの問合せ処理に関する研究に従事. 電子情報通信学会, ACM, IEEE, CS 各会員. 博士(工学).