

## P2P ストリーミング環境における 再生途切れ時間短縮方式

坂下 卓<sup>†1</sup> 義久 智 樹<sup>†2</sup>  
原 隆 浩<sup>†1</sup> 西尾 章 治 郎<sup>†1</sup>

近年、インターネットを介した映像配信サービスの普及にともない、P2P (Peer-to-Peer) ストリーミング環境に注目が集まっている。P2P ストリーミング環境では、再生端末間で映像や音楽などのストリーミングデータを分割して送受信することで、サーバに発生する負荷を軽減できる。しかし、分割データの受信が再生に間に合わない場合、再生に途切れが発生する。そこで本論文では、P2P ストリーミング環境における再生途切れ時間短縮方式を提案する。提案方式では、分割データの再生位置と、P2P ネットワーク内に存在する分割データの数を考慮して分割データの重要度を計算する。重要度に基づいて分割データを受信することで、再生中に発生する途切れ時間を短縮する。

### A Method to Reduce Interruption Time in P2P Streaming Environments

SUGURU SAKASHITA,<sup>†1</sup> TOMOKI YOSHIHISA,<sup>†2</sup>  
TAKAHIRO HARA<sup>†1</sup> and SHOJIRO NISHIO<sup>†1</sup>

Due to the recent prevalence of Internet streaming services, there has been an increasing interest in P2P (Peer-to-Peer) streaming environments. In P2P streaming environments, the load of the server can be reduced by transmitting divided streaming data such as movies and musics among clients. However, if clients cannot receive a piece of the divided data until the time to play it, interruption of the play occurs. In this paper, we propose a method to reduce interruption time in P2P streaming environments. Our proposed method calculates the importance of each piece of the divided data considering its playing position and the number of the pieces distributed in the P2P network. Our proposed method can reduce the interruption time by receiving the divided data based on the importance.

#### 1. はじめに

近年、マルチメディアコンテンツのデジタル化にともない、ビデオオンデマンド<sup>1)</sup>やインターネットラジオ、Web 放送<sup>2)</sup>などのストリーミング配信サービスが広く普及している。ストリーミング配信では、ユーザはデータを再生すると同時に次のデータを受信する。従来のサーバ・クライアント型ではユーザ数の増加にともない、サーバの通信負荷が増大するという問題点がある。そこで、P2P ストリーミング環境<sup>3)-5)</sup>に注目が集まっている。P2P ストリーミング環境では、ユーザ間でデータを送受信することにより、サーバ・クライアント型と比較してサーバの負荷を低減させることができる。

一般的な P2P ストリーミング環境では、データはピースと呼ばれるいくつかのセグメントに分割される。P2P ネットワーク上の再生端末 (以降、ピア) は、他のピアからピースを受信する。ピアは再生時刻までにピースを受信することで、データを途切れることなく視聴できる。しかし、ピースの受信が再生に間に合わない場合、コンテンツの再生が途切れるという問題が発生する。再生が長時間途切れると、ユーザはコンテンツの展開や内容に集中できず、ストレスを感じるため、ストリーミング配信の品質が低下する。ここで、P2P ストリーミング環境には、先頭のピースを受信すると同時に再生を開始する方式と、端末側でバッファを行い、ある程度の時間を待って再生を開始する方式がある。本研究では、前者の方式を対象とする。ここで、前者の方式における再生途切れ時間の合計を  $w$  とすると、後者の方式において、初めに  $w$  だけ待ってその間に受信できるピースをバッファに貯めることでコンテンツを途切れなく再生できる。つまり、前者の方式において  $w$  を短くすることは、後者の方式においてバッファにかかる時間を短縮することに相当する。

P2P ストリーミング環境において再生途切れ時間を短縮するためには、以下の 2 点を考慮することが重要である。

- ピースの緊急性

現在の再生位置から近い将来に再生されるピースほど緊急性が高く、ピアはより早くそのピースを受信しなければ再生途切れ時間が長くなる。たとえば、ピースの再生時間が

<sup>†1</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

<sup>†2</sup> 大阪大学サイバーメディアセンター

Cybermedia Center, Osaka University

4 秒の場合を考える．ピアはコンテンツの最初の 4 秒を再生すると，最初のピースの再生を終了する．ピアは 2 番目のピースを持っていると続けて再生できる．しかし，2 番目と 3 番目のピースを受信していない場合，2 番目のピースが 3 番目のピースよりも緊急性が高いといえる．最初のピースの再生開始から 4 秒以内に 2 番目のピースの受信が完了しない場合，コンテンツの再生が途切れる．4 秒以内に 2 番目のピースを受信すると，再生は途切れない．一方，3 番目のピースは再生開始まで 8 秒の余裕がある．このように，前半のピースほど緊急性が高く，緊急性の高いピースを優先して受信する方が再生途切れ時間を短縮できる．

- ピースの希少性

P2P ネットワーク内に存在するピースの数に偏りがある場合，P2P ネットワーク内の数が少ないピースを保持するピアに受信要求が集中するため，受信に時間がかかり，再生途切れ時間が長くなる．したがって，数が少ないピースほど P2P ネットワーク内に早く分散させ，P2P ネットワーク内に存在するピースの数を増加させるほど再生途切れ時間を短縮できる．

これまでに再生途切れ時間を短縮する研究は行われているが，ピースの緊急性と希少性のどちらかしか考慮されていないという問題があった．

そこで本論文では，P2P ストリーミング環境における再生途切れ時間を短縮するためのピース受信方式である BIS (BiToS + Immediacy and Scarcity) 方式を提案する．BIS 方式では，各ピースに対して，ピースの緊急性と P2P ネットワーク内のピースの希少性を考慮して重要度を計算する．重要度の高いピースを優先して受信することで，再生途切れ時間を短縮する．本研究の新規性は，緊急性と希少性を考慮することであり，既存手法より再生途切れ時間を短縮できる点で P2P ストリーミング配信の分野で貢献できる．シミュレーション実験により，BIS 方式を用いることで，既存手法と比較して P2P ネットワーク内におけるコンテンツの再生中に発生する途切れ時間を短縮できるだけでなく，特にピアの到着間隔が短い場合に，再生途切れ時間を短縮できることを確認する．

以下，2 章で関連研究について説明し，3 章で一般的な P2P ストリーミング環境について説明し，4 章で筆者らが提案した BIS 方式について説明する．5 章で BIS 方式について評価を行い，6 章で考察を行う．最後に 7 章で本論文をまとめる．

## 2. 関連研究

P2P ネットワーク上でファイル共有を行うアプリケーションである BitTorrent<sup>(6),(7)</sup> では，

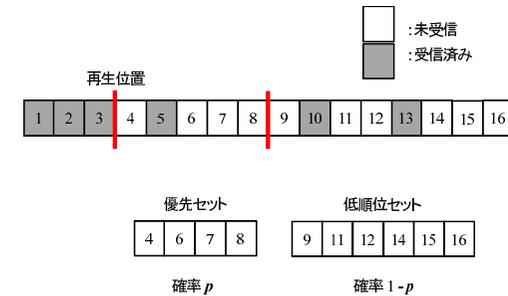


図 1 BiToS におけるセット選択  
Fig. 1 Piece selection under BiToS.

データをピース単位に分割し，レアレストファスト方式を用いてピアどうしでピースを送受信する．レアレストファスト方式では，ピアは P2P ネットワーク内の数が少ないピースを優先的に受信する．しかし，ストリーミング配信の場合，レアレストファスト方式では再生位置を考慮しないため再生途切れ時間が長くなる可能性がある．そこで，再生途切れ時間を短縮する手法がいくつか提案されている．

BASS (BitTorrent Assisted Streaming System for Video-on-Demand)<sup>(8)</sup> は，専用のストリーミングサーバとピアによってシステムを構築する．ピアはレアレストファスト方式を用いてピースを受信し，再生開始までに P2P ネットワークからピースを受信できない場合，サーバからピースを受信することで，再生途切れ時間を短縮する．

BiToS (Enhancing BitTorrent for Supporting Streaming Applications)<sup>(9)</sup> は，BitTorrent のレアレストファスト方式を改良してコンテンツ再生中の途切れ時間を短縮する手法である．BiToS では，図 1 に示すようにコンテンツの未受信のピースを，再生位置に近い優先セットとそれ以外の低順位セットに区別する．ピアは確率  $p$  を用いてピース選択を行うセットを決定し，選択されたセットからレアレストファスト方式を用いて受信ピースを選択する．ここで，確率  $p$  は，コンテンツ再生のために早急に必要と，P2P ネットワーク内に存在する数が少ないピースのどちらを優先するかのバランスを表す．BiToS では，各セット内ではレアレストファスト方式を用いて受信するピースを選択するため，優先セット内のピースのうち，再生位置に近いにもかかわらず P2P ネットワーク内に存在する数が多いピースが長時間受信されない場合がある．

これらの手法はピースの希少性のみ考慮しており，ピースの緊急性を考慮していない．ピー

スの緊急性を考慮した手法として XebCast<sup>10)</sup> と文献 11) の手法, DAW<sup>12)</sup> が提案されている。XebCast は, オリジナルコンテンツを保持するピアを先頭にピアが直列に接続し, ピースを先頭から受信することによって再生途切れ時間を短縮する。ピアは, ピースの受信に費やせる時間の余裕である余裕時間を計算し, 自身の余裕時間が大きくなるように送信経路を作成することで再生途切れ時間を短縮する。しかし, 通信帯域の狭いピアが送信経路に割り込むとネットワーク全体の受信速度が低下する場合がある。

文献 11) では, VBR (Variable Bit Rate) のコンテンツを配信する P2P ストリーミング環境において途切れ時間を短縮する手法を提案している。この手法では, 未受信のピースを, 再生位置に近いピースで構成される再生バッファとそれ以外のピースで構成されるピッキングレンジに分割する。まず, 再生バッファ内のピースを順番に受信し, 再生バッファのピースがすべて受信済み, または受信中の場合にピッキングレンジからピースを受信する。ピッキングレンジからピースを受信する際, データの再生時刻までに最も早く受信できるピースを選択する。この手法は, 再生時刻に近いピースを順番に受信する場合に等しく, P2P ネットワーク内に存在する数が少ないピースが分散しにくいいため, 途切れ時間が長くなる。

DAW (The distance-availability weighted method)<sup>12)</sup> では, 文献 11) と同様に, コンテンツの未受信のピースを 2 つのセットに区別する。まずは, 再生バッファからピースを順番に受信し, 再生バッファ内のピースがすべて受信済みであるか受信中である場合, ピッキングレンジからピースを受信する。ピッキングレンジからピースを選択する場合, P2P ネットワーク内に存在するピース数とピースが再生されるまでの時間を考慮して以下のような重要度 *Priority* を定義し, 最も重要度が高いピースを受信する。

$$Priority = \frac{1}{((P_i - P_c) \times m_i)} \quad (1)$$

ここで,  $P_i$  はピース  $i$  のピース ID,  $P_c$  は再生バッファ内の最後のピース ID,  $m_i$  はピース  $i$  を保持するピア数を示す。5 章の評価から, 提案方式は DAW よりも再生途切れ時間を短縮できることが分かる。

一方, ストリーミング配信における耐障害性を確保するために, erasure coding などの技術を用いる手法が提案されている<sup>13)–15)</sup>。erasure coding では, 誤り訂正符号を加えたオリジナルのデータを  $n$  個のブロックに分割し, 少なくとも  $m$  個のブロック ( $n \geq m$ ) が揃えばオリジナルのデータを復元できる。これは, データの一部が受信できなくてもオリジナルのデータを復元できるようにする障害対策のための分割方法である。erasure coding では,  $m$  個のブロックを受信するまでデータを復元できないため, 単純に  $m$  個のブロックを順番

に受信することで, データを受信しながら再生するストリーミング配信が可能になるわけではない。一方, 提案方式におけるピース分割は, データを分割して受信した部分から再生できるようにする, ストリーミング配信のための分割方法である。ピース分割と erasure coding は, データを分割する点では同じだが, 目的が異なり, 併用が可能である。実際に文献 13)–15) では, ピース内で erasure coding などを行い耐障害性を確保している手法が提案されている。提案方式でも同様に, erasure coding を適用することでコンテンツのデータの耐障害性を確保できる。

### 3. P2P ストリーミング環境

本章では, 本論文で想定する P2P ストリーミング環境について説明する。

図 2 に示すように, P2P ストリーミング環境にはコンテンツを共有するピアとそのネットワークを管理するサーバがある。サーバは, P2P ネットワークに参加しているピアの通信速度とコンテンツの取得状況を管理している。サーバはデータを配信せず, P2P ネットワークを管理するだけであるため通信負荷が小さい。各ピアは定期的にサーバにアクセスし, P2P ネットワークに参加しているピアの情報を取得する。オリジナルのコンテンツを保持するピアは最初 1 つだけである。ピアはまず視聴するコンテンツの配信サービス元を Web ページなどから調べ, データの情報とサーバのアドレスを取得する。コンテンツを視聴したいときは, サーバに接続してピースを保持するピアのリストを取得する。ピアはピアリストを参照して, 通信ピアを選択しピースを受信する。ピースの受信が完了すると, そのピースを再生できる。これは, BitTorrent と同様の環境であり, 現実的な想定である。

#### 3.1 ピース選択

P2P ストリーミング環境において, ピアは接続するピアを選択した後, サーバからピアが保持するピースの情報を取得し, 必要なピースを受信する。ピアがピースを受信する順序は, 再生途切れ時間に影響を与える。たとえば, P2P ネットワーク内の数が少ないピースを保持するピアに対して多くのピアが受信要求する場合, ピースの受信速度は低下する。また, ピアの通信負荷を分散させるためピアあたりの同時に通信可能なピア数は制限されており, 各ピアは通信可能なピアにしかピースを送信できない。制限数以上の受信要求が発生する場合, 他のピアが受信を完了するまでピースの受信を待機しなければならない。

そこで, P2P ストリーミング環境では, ピアは 2 つの方法を利用して, 定期的にピアとの接続の更新を試みる。1 つは送信速度の速いピアと接続する方法である。P2P ネットワーク内のピア間の通信帯域は異なるため, 送信速度の速いピアと接続することでピースを早く

受信でき、再生途切れ時間を短縮できる。しかし、通信速度の速いピアとのみ接続しようとするだけでは、通信を行うピアに偏りが起こる。そこで、2つ目の方法では、ピアリストからランダムにあるピアを選択し接続する。たとえば、ピアが4個のピアと接続するとき、各ピアは3個のピアを1つ目の方法から選択し、残りの1個のピアを2つ目の方法から選択する。

### 3.2 コンテンツの再生

扱うデータは映画やドラマなどのコンテンツである。データが音楽ライブやスポーツ中継のような生放送の場合、ピアはあらかじめデータを受信できないため、配信されるコンテンツは配信開始時点でファイルとして完成しているものとする。コンテンツは  $n$  個のピースに分割される。本研究では、ピアはコンテンツの先頭ピースを受信し終えた時点で、コンテンツを先頭からピース単位で再生するものとし、早送りや巻き戻しは行わない。また、コンテンツの配信を効率的に行うために、再生後のピースもすべてキャッシュする方式を想定する。

図3に、ピアがピースを受信し、それを再生する様子を示す。図の上段は、コンテンツを要求してからピースを受信する様子を表す。下段は、受信したピースを一定の再生レートで再生する様子を示す。各ピースにはピースIDが割り当てられており、ピアはピースの受信が完了するとそれを再生できる。しかし、再生開始までピースを受信できない場合、そのピースの受信が完了するまでコンテンツの再生が途切れる。本論文では、最初のピースを受信するまでの時間とコンテンツの再生中に発生する途切れ時間の和を再生途切れ時間と定義する。ピアはコンテンツの再生を終えると、P2Pネットワークから離脱する。

## 4. 提案方式

本章では、P2Pストリーミング環境において再生途切れ時間を短縮するBIS (BiToS + Immediacy and Scarcity) 方式を提案する。BIS方式では重要度を定義し、ピアは接続ピアからピースを受信する際、各ピースの重要度を計算し、最も重要度の高いピースを受信する。

### 4.1 BIS (BiToS + Immediacy and Scarcity) 方式

BIS方式では、BiToSと同様に未受信のピースを2つのセットに区別する。優先セットから受信ピースを選択する際、ピアはセット内の全ピースに対する重要度を算出し、重要度の高いピースを選択して受信する。低順位セットが選択された場合、セット内のピースは再生されるまでに余裕があるため、緊急性を考慮する必要はない。そこで、低順位セットが選

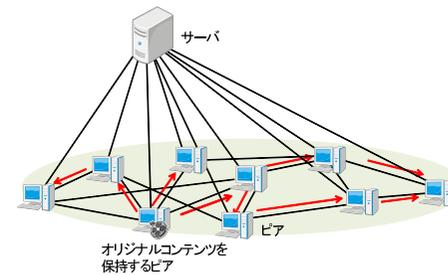


図2 P2P ストリーミング環境

Fig. 2 A P2P streaming environment.

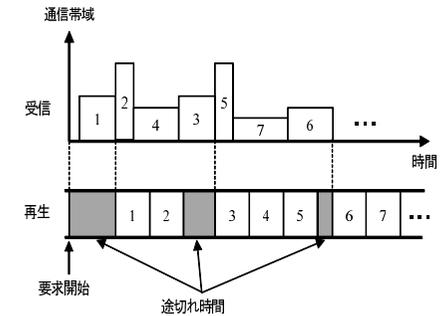


図3 コンテンツ再生の様子

Fig. 3 An image to play a content.

択された場合、BiToSと同様にレアルタイムファスト方式を用いて受信ピースを選択する。

ピースの重要度は、1章で説明したように、緊急性と希少性の2点が重要である。そこで、ピース  $i$  ( $1 \leq i \leq n_p$ ) に対する重要度  $D_i$  は以下のように定義する。 $n_p$  は優先セット内のピースの数を表す。

$$D_i = cI_i + (1 - c)S_i \quad (2)$$

ここで、 $I_i$  はピース  $i$  に対する各ピアの緊急性、 $S_i$  はピース  $i$  のP2Pネットワーク内における希少性、 $c$  ( $0 < c \leq 1.0$ ) は重み係数を表す。 $c = 0$  の場合、BIS方式はBiToSと等しくなる。重要度の最も高いピースが複数存在する場合には、再生位置に近いピースを選択する。

以下では、ピース  $i$  に対する各ピアの緊急性  $I_i$  と、ピース  $i$  のP2Pネットワーク内における希少性  $S_i$  の定義について、それぞれ詳しく説明する。

#### 4.1.1 各ピアの緊急性 $I_i$

ピースの再生はコンテンツの再生順に従って行われるため、現在の再生位置から近い将来に再生されるピースほど緊急性が高く、再生されるまで余裕のあるコンテンツ後半のピースほど緊急性が低い。したがって、ピース  $i$  に対する各ピアの緊急性  $I_i$  を、次式のような線形関数で定義する。本研究では、簡単化のため、 $I_i$  を単純な線形関数で定義している。

$$I_i = \begin{cases} 0 & (i = 0, \dots, n, \quad i \in RP) \\ 1 - \frac{i - h}{n} & (i = h, \dots, n, \quad i \notin RP) \end{cases} \quad (3)$$

ここで、 $h$  は次に再生されるピースID、 $n$  はコンテンツの最後に再生されるピースID、 $RP$

は受信済みのピースの集合を表す。各ピアは、コンテンツの再生位置から  $h$  を把握できる。また、 $n$  はコンテンツの受信要求をサーバに問い合わせた際に取得できる。

#### 4.1.2 P2P ネットワーク内における希少性 $S_i$

ピース  $i$  の P2P ネットワーク内における希少性  $S_i$  は、ピース  $i$  を保持している P2P ネットワーク内のピア数を表す。希少性の高いピースに受信要求が集中すると、ピースの受信速度が低下してしまう。そこで、P2P ネットワーク内に存在する数が少ないピースほど優先して選択されるように、P2P ネットワーク内におけるピース  $i$  の希少性  $S_i$  は次式のように定義することで、数が少ないピースが P2P ネットワーク内により早く分散される。

$$S_i = \frac{N - m}{N} \quad (4)$$

ここで、 $N$  は P2P ネットワークに接続している全ピア数、 $m$  はピース  $i$  を保持しているピア数を表す。 $N$ 、 $m$  は、サーバから取得できる。

#### 4.2 具体例

ここでは、ピアが重要度を計算する方法を具体例を用いて説明する。

配信されるコンテンツは 60 分、2 Mbps のデータとする。ピースサイズが 1,024 Kbyte の場合、コンテンツは 900 Mbyte であるため、900 個のピースに分割できる。全ピアの重み係数は  $c = 0.5$  とする。ここで、 $P_i$  はピース  $i$  ( $i = 0, \dots, 899$ ) を表す。

まず、初めてピアがコンテンツを受信する場合、サーバからコンテンツのピースを保持するピアのリストを受信する。最初は P2P ネットワーク内にオリジナルのコンテンツを保持するピアは 1 つしか存在しないため、そのピアからピースを受信し、ピアは全ピースに対する重要度を計算する。この場合、ピアはまだピースを 1 つも受信していないため、先頭のピースの緊急性は  $I_0 = 1 - \frac{0-0}{900} = 1$  と求められる。先頭のピースの希少性は、オリジナルのコンテンツを保持するピア以外にピースを保持するピアが存在しないため、 $S_0 = \frac{1-1}{1} = 0$  と計算される。したがって、この例では、 $D_0$  は  $1 \times 0.5 + 0 \times (1 - 0.5) = 1$  である。同様に、 $I_1 = 0.998$ 、 $S_1 = 0$  から  $D_1 = 0.998$  と計算できる。このように全ピースに対する重要度  $D_i$  を計算する。その結果、 $D_0$  の重要度が一番高いため、ピアは先頭ピース  $P_0$  を受信する。 $P_0$  を受信完了すると、同様に新しいピースの重要度を計算し、選択する。

しばらくして、P2P ネットワーク内に 50 ピアが存在し、次に  $P_{50}$  を再生するピアがいる場合を考える。そのピアは接続ピアから  $P_{50}$ 、 $P_{51}$ 、 $P_{52}$  を受信できるとする。各ピースを保持するピア数は  $P_{50} = 40$ 、 $P_{51} = 10$ 、 $P_{52} = 30$  とする。このとき、重要度は  $D_{50} = 0.510$ 、 $D_{51} = 0.889$ 、 $D_{52} = 0.698$  である。その結果、ピアは最も重要度が高い  $P_{51}$  を選択して受

信する。このように、ピアは各ピースに対する重要度を計算し、全ピースを受信するまでピースを受信する。全ピースの受信が完了すると、接続ピアとの通信を切断し、P2P ネットワークから離脱する。

## 5. 性能評価

本章では、性能評価のために行ったシミュレーション実験について説明する。シミュレーション実験では、GPS<sup>16)</sup> と呼ばれるシミュレータを利用した。GPS は P2P ネットワークにおけるコンテンツ配信を評価するためにいくつかの研究で利用されている<sup>17),18)</sup>。

### 5.1 シミュレーション環境

有線 LAN でインターネットにつながった端末で構築する P2P ネットワークを想定し、表 1 に示すパラメータを用いてシミュレーション実験を行った。ピアの到着は他のピアの到着に依存せずポアソン過程であると考え、ピアの到着間隔はポアソン分布で与える。ここで、ピアは到着順にピア ID が割り当てられるものとする。たとえば、ピア 0 は P2P ネットワークに最初に参加し、次に参加するピアはピア 1 となる。実験では、400 個のピアが到着するまでをシミュレーションした。また、ピア間の最大通信帯域はすべて一定とし、複数のピアと同時に通信する場合、この最大通信帯域を等分割して各ピアに割り当てる。ピアはコンテンツを受信するのに十分な記憶容量を保持するものとし、他のピアにピースを送信するために再生後のピースも保持する。これは、近年のコンピュータがハードディスクのような大容量の記憶容量を保持していることから現実的である。3.1 節で説明したように、ピアは定期的に通信するピアのリストを更新する。ピアはコンテンツを再生終了後、P2P ネットワークから離脱する。簡単化のため、1 つの P2P ネットワークでは、1 つのコンテンツ

表 1 実験のパラメータ  
Table 1 Experimental parameters.

パラメータ	値
コンテンツ長	2,700 [秒]
ビットレート	2 [Mbps]
ピースサイズ	1 [Mbyte]
オリジナルのコンテンツ数	1 [個]
ピア数	400 [個]
最大通信帯域	8 [Mbps]
同時に通信可能なピア数 $N_s$	4 [個]
ピアの平均要求到着間隔 $r$	30 [秒]
通信ピアの切り替え間隔	10 [秒]

が共有されるものとする．

## 5.2 評価指標

P2P ストリーミング環境では、評価指標として、システム全体の負荷や、特定のピアへの通信負荷、再生途切れ時間などが考えられる．ここで、システム全体の負荷や、特定のノードへの負荷の集中は、結果的に再生途切れ時間が長くなることにつながる．システム全体の負荷が大きくなると、帯域幅が減少してピースの送受信に長い時間を要する．その結果、再生開始時刻までにピースを受信完了できる確率が低くなり、再生途切れ時間が長くなる．また、特定のピアへの負荷が集中することは、そのピアが希少なピースを保持していることを表している．この場合、P2P ネットワーク内でのピースの配布に長い時間を要するため、結果として、再生途切れ時間が長くなる．システム設計者の観点では、再生途切れ時間が長くなるとサービスの品質が低下するため、本論文では直接的な性能指標として、平均途切れ時間について評価を行う．

P2P ストリーミング環境におけるほとんどの従来手法では、システム全体の負荷や特定のノードへの負荷の集中を軽減することで再生途切れ時間を短縮している．たとえば、本論文で想定している BitTorrent の環境では、レアレストファスト方式を用いている．レアレストファスト方式は、数の少ないピースを優先的に受信する方式であり、希少なピースを持つピアに通信負荷が集中しないようにしている．提案方式においても、確率的にピースセットを選択し、低順位セット内のピース選択にレアレストファスト方式を用いることで、特定のピアに通信負荷が集中しないようにしている．また、同時に通信可能なピア数に制限を設けることで、システム全体の負荷を下げている．

2章で述べたように、データの分割方法として、障害対策のために erasure coding などを用いる方法があるが、本論文の主題は再生途切れ時間を短縮することである．これらの障害対策を施したうえで評価を行うと、障害対策の手法の影響が出てしまい、提案方式の効果を純粋に判断できなくなる．このため、本論文では、erasure coding などの技術を用いず、ピース分割のみを行ったうえで再生途切れ時間の評価を行う．

## 5.3 比較方式

シミュレーション実験では、提案方式である BIS 方式と、BiToS<sup>9)</sup>、DAW<sup>12)</sup> を比較する．オリジナルの BiToS では、ピースの受信がコンテンツの再生に間に合わない場合、対応するピースの再生をスキップし、次のピースを再生する．再生を飛ばしたピースに対しては以降、受信要求を出さない．しかし、本研究では再生途切れ時間について評価するため、ピースの受信がコンテンツの再生に間に合わない場合、そのピースの受信が完了するまで待

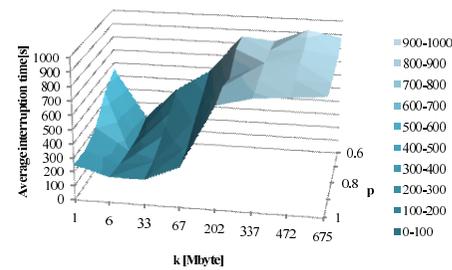


図4 重み係数  $c = 0.0$  の場合の平均途切れ時間  
Fig. 4 Average interruption time when  $c = 0.0$ .

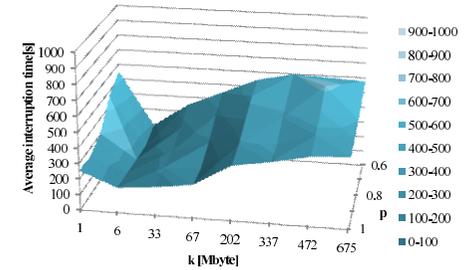


図5 重み係数  $c = 0.4$  の場合の平均途切れ時間  
Fig. 5 Average interruption time when  $c = 0.4$ .

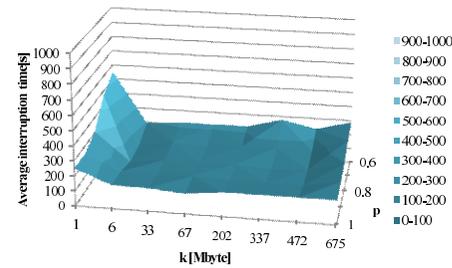


図6 重み係数  $c = 0.8$  の場合の平均途切れ時間  
Fig. 6 Average interruption time when  $c = 0.8$ .

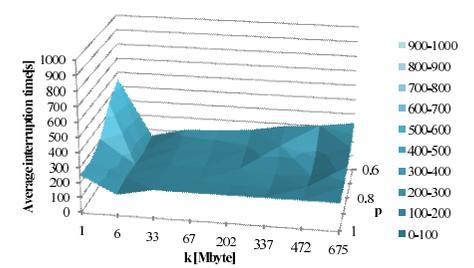


図7 重み係数  $c = 1.0$  の場合の平均途切れ時間  
Fig. 7 Average interruption time when  $c = 1.0$ .

機するものとする．そのため、BiToS においてもピースの再生をスキップせず、該当ピースの受信が終わるまで待機するものとする．BIS 方式、DAW においても同様に、ピースの再生をスキップしない．なお、BIS 方式において、 $c = 0$  の場合は BiToS、 $c = 1.0$  の場合は文献 11) の手法に相当する．

## 5.4 パラメータの影響

BIS 方式では優先セットのデータサイズと優先セットの選択確率によって再生途切れ時間が変化する．そこで、優先セットのデータサイズ  $k$  Mbyte と優先セットの選択確率  $p$  を変化した場合の平均途切れ時間について、重み係数  $c = 0.0, 0.4, 0.8, 1.0$  の結果をそれぞれ、図 4、図 5、図 6、図 7 に示す．

$k$  の最適値は、コンテンツ長や通信帯域、ピアの送信能力などのシステム環境に依存して変化する．ここで、本研究では最大通信帯域を、接続しているピアの数に等分割して通信す

るように想定しているため、使用可能な通信帯域は頻繁に変化する。いくつかのピアと接続するかを事前に予測することが困難なため、使用可能な通信帯域を予測することも困難である。さらに、各ピアの送信能力（ハードウェア的な帯域幅や処理能力）もピアごとに異なり、他のアプリケーションに処理能力が割かれるなど、頻繁に変化するものと考えられる。このほかにも、セット内のピースを持つピア数や通信帯域の揺れといった要因も  $k$  の適切な値に影響するが、これらも予測不能である。以上のような理由からコンテンツ長やシステム環境に応じて、最適な  $k$  の値を解析的に求めることは困難である。そのため、本評価では  $k$  の値を変化させて、与えられたシミュレーション環境における  $k$  の最適値を総当りに調べた。なお、コンテンツ長やシステム環境が変わると、適切な  $k$  の値も変化するため、本評価で得られた  $k$  の最適値の値そのものに絶対的な意味があるわけではない。

まず、優先セットのデータサイズ  $k$  Mbyte と重み係数  $c$  の影響に関して（図の横軸方向）、図 4、図 5 より、 $c$  が比較的小さい場合は、 $k \geq 6$  Mbyte において、 $k$  が小さいほど平均途切れ時間が短縮されることが分かる。これは、 $k$  が大きい場合、優先セット内のピースの数が多くなり、優先セットが選択された際、現在の再生位置からあまり必要としない、P2P ネットワーク内に存在する数が少ないピースを優先的に受信しようとするためである。一方、図 6、図 7 より、 $c$  が比較的大きい場合、 $k \geq 6$  Mbyte において、 $k$  が変化してもあまり平均途切れ時間が変化しないことが分かる。これは、再生位置に近いピースを順番に受信しようとするため、 $k$  が大きい場合でも、選択されるピースの順番があまり変化しないためである。また、すべての場合において、 $k \leq 6$  Mbyte の場合は、 $k$  が小さくなるほど平均途切れ時間が増加することが分かる。これは、優先セット内のピースの数が少なくなり、ピースの希少性の影響が少なくなるためである。さらに、優先セットからピースを受信する際、セット内の全ピースを他のピアから受信中で、ピースを受信できないことがあるためである。

次に、優先セットの選択確率  $p$  と重み係数  $c$  の影響に関して（図の縦軸方向）考察する。図より、 $p$  が 0.6 から大きくなると徐々に平均途切れ時間が短くなり、1.0 に近づくと増加することが分かる。これは、 $p$  が大きくなると、優先セットが選択される確率が高くなり、再生位置に近いピースを受信できるためである。一方、1.0 に近くなると、低順位セットが選択されないため、コンテンツ後半のピースが P2P ネットワーク内に分散しない。その結果、特定のピースに受信要求が集中してしまい、平均途切れ時間が増加する。パラメータの決定方法については、後に 6.1 節で説明する。

### 5.5 要求到着間隔の影響

ピアの要求到着間隔  $r$  が変化すると、P2P ネットワーク内に存在するピア数が変化する

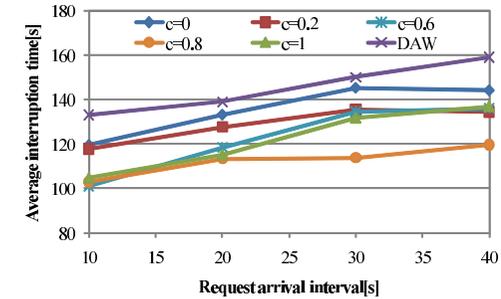


図 8 ピアの要求到着間隔の影響

Fig. 8 Influence of the request arrival interval.

ため、P2P ネットワーク内のピースの分布が変化し、平均途切れ時間が変化する。そこで、BIS 方式における重み定数  $c$  を 0.0 から 1.0 まで変化させ、ピアの要求到着間隔  $r$  が 10 秒から 40 秒まで 10 秒ずつ異なる場合の平均途切れ時間の変化を図 8 に示す。 $k$  と  $p$  は、5.4 節と同様にシミュレーションを行い、平均途切れ時間が最短になるように与えた。

図 8 より、BIS 方式は  $c = 0.0$  で示される BiToS、DAW と比較して平均途切れ時間が短く、重み係数  $c$  が大きい場合平均途切れ時間が短くなること分かる。たとえば、要求到着間隔が 30 秒の場合、平均途切れ時間は 111.7 秒であり、BiToS よりも 23.1%短く、DAW よりも 26.3%短い。これは、優先セットが選択された際、BIS 方式は BiToS と比較して再生位置に近いピースを受信できるためである。DAW は再生位置に近いセットである再生バッファ内のピースをまず順番に受信するため、BIS 方式、BiToS と比較してピースが P2P ネットワーク内に分散しにくく途切れ時間が長くなる。特に要求到着間隔が長い場合、P2P ネットワーク内に存在するピア数が少ないため、よりピースが分散しにくく途切れ時間が長くなる。一方、BIS 方式において  $c = 1.0$  の場合（文献 11）の手法に相当）、優先セットが選択された際ピースを順番に受信しようとするため、P2P ネットワークにピースが分散しにくく、全体的に平均途切れ時間が長くなる。ただし、要求到着間隔が短い場合、平均途切れ時間が短くなっている。これは、P2P ネットワーク内に存在するピア数が多く、P2P ネットワーク内にピースが分散されやすいためである。

### 5.6 同時に通信可能なピア数 $N_s$ の影響

ノードあたりの同時に通信可能なピア数  $N_s$  が変化すると、送受信するピースが変化し、P2P ネットワーク内のピースの分布が変化するため、平均途切れ時間が変化する。そこで、

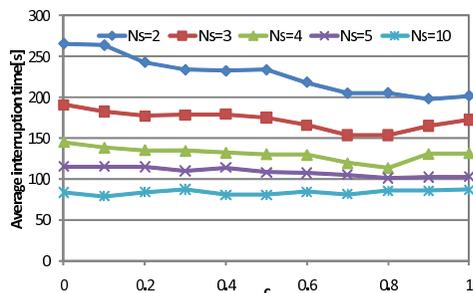


図 9 同時に通信可能なピア数と平均途切れ時間

Fig. 9 The maximum number of peers communicating simultaneously and the average interruption time.

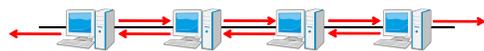


図 10  $N_s = 2$  の場合に考えられるピアの送信経路  
Fig. 10 Peer connections when  $N_s = 2$ .

BIS 方式における重み定数  $c$  を 0.0 から 1.0 まで変化させ、 $N_s$  が 2 から 10 まで異なる場合の平均途切れ時間の変化を調べた結果を図 9 に示す。 $k$  と  $p$  は、5.4 節と同様にシミュレーションを行い、平均途切れ時間が最短になるように与えた。

図 9 から、BIS 方式は BiToS と比較して、 $N_s$  が少ない場合に、平均途切れ時間を大きく短縮できることが分かる。これは、BIS 方式では、優先セットからピースを受信する際、再生位置に近いピースと P2P ネットワーク内に存在するピースの数を考慮して受信できるためだと考えられる。 $N_s$  が少ない場合、各ピアに割り当てられる通信帯域は大きくなるが、P2P ネットワーク内にピースが分散しにくくなるため、重み定数  $c$  が変化することによって平均途切れ時間が短くなる。特に  $N_s = 2$  の場合、ピア間で双方向のピース交換を行うことにより、ピアが直列に送信経路を構築する傾向がある(図 10)。この場合、ピースが分散しにくくなるため、再生位置に近いピースを優先的に受信する方が途切れ時間が短くなる。一方、 $N_s$  が多い場合、通信ピア数が多くなり P2P ネットワーク内にピースが分散しやすいが、各ピアに割り当てられる通信帯域が小さくなり、再生開始までにピースが受信できない場合があるため、重み定数  $c$  による影響が小さくなる。

### 5.7 コンテンツのビットレートの影響

コンテンツのビットレートが変化すると、コンテンツのピースの数が変化し、ピアが P2P

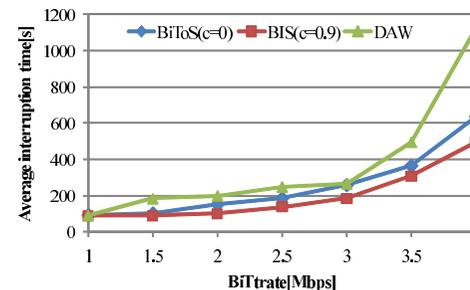


図 11 コンテンツのビットレートの影響

Fig. 11 Influence of bitrate of the streaming data.

ネットワーク内に存在する時間が変化するため、平均途切れ時間が変化する。そこで、ピアの要求到着間隔が 60 秒の場合において、コンテンツのビットレートを 1 Mbps から 2.5 Mbps まで 0.5 Mbps ずつ変化させた場合の平均途切れ時間の結果を図 11 に示す。ここで、 $c$  と  $k$  と  $p$  は、5.4 節と同様にシミュレーションを行い、平均途切れ時間が最短になるように与えた。

図 11 より、ビットレートが増加するほど、平均途切れ時間が増加するが、BiToS、DAW と比較して、BIS 方式の平均途切れ時間が短くなるのが分かる。これは、コンテンツのビットレートが増加するほど、P2P ネットワーク内で送受信するピースの種類が増加するが、優先セットが選択された際、BIS 方式の方が緊急性と希少性を両方考慮して受信できるためである。BiToS においても、 $k$  の値を小さくすることで再生位置に近いピースを受信できるようになるが、優先セットに含まれるピースの数が少なくなる。このため、自身が保持しておらず通信ピアが保持しているピースが優先セット内にある確率が低くなり、平均途切れ時間が長くなると考えられる。DAW は、BIS 方式、BiToS と比較して P2P ネットワーク内にピースが分散しにくいいため、ビットレートが増加するほど、途切れ時間が増加しやすい。

## 6. 考 察

### 6.1 パラメータ $c, k, p$ の決め方

シミュレーション実験より、通信帯域などが変化すると、再生途切れ時間が最も短くなるようなパラメータも変化することが分かる。実環境においても、 $c, k, p$  の値によって再生途切れ時間が変化するため、再生途切れ時間を短縮できるように適切な値を設定する必要がある。

ある．具体的には，定期的に  $c, k, p$  の値を変えながらシステムを運用し，運営者の所望の再生途切れ時間になるように設定する．実際の途切れ時間は，システムの運用時にネットワークを管理するサーバが把握できる．

運営者が，再生途切れ時間を最短にしたいわけではなく，通信帯域が多少変化しても再生途切れ時間が大きく変化しないようにしたい場合，図 6 の平らな部分のように， $k$  や  $p$  が多少変化しても再生途切れ時間が大きく変わらない部分，たとえば  $k = 0.1, p = 0.8$  などに設定することが考えられる．必ずしも再生途切れ時間を最短にする  $c, k, p$  を与える必要はなく，上記のように  $c, k, p$  を適切に設定することで，既存手法より再生途切れ時間を短縮できる．

## 6.2 配信するコンテンツの数

本論文では，P2P ネットワーク内で配信されるコンテンツは 1 種類のみと想定して評価を行ったが，P2P ネットワーク内に複数のコンテンツを配信する場合，コンテンツごとに P2P ネットワークを構築することで可能になる．P2P ネットワークでは，コンテンツごとにピアの挙動が異なるため，本研究の想定のようにコンテンツごとに P2P ネットワークを構築することが現実的であると考えられる．

## 7. 結 論

本論文では，P2P ストリーミング環境において，コンテンツ再生中の途切れ時間を短縮するための分割データ受信方式を提案した．提案方式では，再生位置に近い分割データを優先しつつ，P2P ネットワーク内に存在する数が少ない分割データを拡散させることで，P2P ネットワーク内のユーザに対してコンテンツの再生途切れ時間を短縮する．シミュレーション実験より，既存手法と比較して再生途切れ時間を短縮できていることを確認した．たとえば，コンテンツを受信するピアの要求到着間隔が 30 秒の場合，提案方式の平均途切れ時間は 111.7 秒で，既存手法よりも 23.1% 平均途切れ時間を短縮できる．

謝辞 本研究の一部は，科学研究費補助金若手研究 (B)「端末伝送型インターネット放送におけるコンテンツ配信方式」(課題番号：21700108) によるものである．ここに記して謝意を表す．

## 参 考 文 献

- 1) BBTv. <http://www.bbtv.com/>
- 2) YouTube. <http://jp.youtube.com/>

- 3) Hefeeda, M., Habib, A., Botev, B., Xu, D. and Bhargava, B.: Promise: A peer-to-peer media streaming system, *Proc. MULTIMEDIA'03*, pp.45–54 (Nov. 2003).
- 4) Guo, Y., Suh, K., Kurose, J. and Towsley, D.: A peer-to-peer on-demand streaming service and its performance evaluation, *Proc. ICME'03*, Vol.2, pp.649–652 (July 2003).
- 5) Zhang, X., Liu, J., Li, B. and Yum, T.-S.P.: CoolStreaming/DoNet: A data-driven overlay network for efficient live media streaming, *Proc. INFOCOM'05*, Vol.3, pp.2102–2111 (Mar. 2005).
- 6) BitTorrent. <http://www.bittorrent.com/>
- 7) Cohen, B.: Incentives build robustness in BitTorrent, *Proc. P2PECON'03* (June 2003).
- 8) Dana, C., Li, D., Harrison, D. and Chuah, C.-N.: BASS: Bittorrent assisted streaming system for video-on-demand, *Proc. MMS'05*, pp.1–4 (Oct. 2005).
- 9) Vlavianos, A., Iliofotou, M. and Faloutsos, M.: BiToS: Enhancing BitTorrent for supporting streaming applications, *Proc. INFOCOM'06*, pp.1–6 (Apr. 2006).
- 10) 義久智樹, 西尾章治郎: 端末伝送型インターネット放送における再生中断時間短縮手法, *情報処理学会論文誌*, Vol.50, No.4, pp.1–10 (2009).
- 11) Erman, D., Vogeeler, K.-D. and Popescu, A.: On piece selection for streaming BitTorrent, *Proc. Swedish National Computer Networking Workshop* (Apr. 2008).
- 12) Sandvik, P. and Neovius, M.: The distance-availability weighted piece selection method for BitTorrent: A BitTorrent piece selection method for on-demand streaming, *Proc. AP2PS'09*, pp.198–202 (Oct. 2009).
- 13) Li, J.: PeerStreaming: A practical receiver-driven peer-to-peer media streaming system, Technical Report, Microsoft Research TR-2004-101 (Sep. 2004).
- 14) Wu, C. and Li, B.: Optimal peer selection for minimum-delay peer-to-peer streaming with rateless codes, *Proc. P2PMMS'05*, pp.69–78 (Nov. 2005).
- 15) Padmanabhan, V.-N., Wang, H.-J. and Chou, P.-A.: Distributing streaming media content using cooperative networking, *Proc. NOSSDAV'02*, pp.177–186 (May 2002).
- 16) Yang, W. and Ghazaleh, N.-A.: GPS: A general peer-to-peer simulator and its use for modeling BitTorrent, *Proc. MASCOTS'05*, pp.425–432 (Sep. 2005).
- 17) Kangasharju, J., Schmidt, U., Bradler, D. and Schroder-Bernhardi, J.: ChunkSim: Simulating peer-to-peer content distribution, *Proc. SpringSim'09*, pp.25–32 (2009).
- 18) Shah, P., Rasheed, J. and Paris, J.-F.: Performance study of unstructured P2P overlay streaming systems, *Proc. ICCCN'08* (Aug. 2008).

(平成 22 年 5 月 17 日受付)

(平成 22 年 12 月 1 日採録)



坂下 卓

2009年大阪大学工学部電子情報エネルギー工学科卒業。現在、同大学大学院情報科学研究科博士前期課程在学中。P2P ストリーミング環境およびインターネット放送に興味を持つ。日本データベース学会の学生会員。



義久 智樹 (正会員)

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を修了し、2005年同専攻博士後期課程修了。博士(情報科学)。2005年京都大学学術情報メディアセンター助教。2008年大阪大学サイバーメディアセンター講師を経て2009年より同准教授となり、現在に至る。この間、カリフォルニア大学アーバイン校客員研究員。P2P ストリーミング環境およびインターネット放送に興味を持つ。電子情報通信学会、IEEE 各会員。



原 隆浩 (正会員)

1995年大阪大学工学部情報システム工学科卒業。1997年同大学大学院工学研究科博士前期課程修了。同年同大学院工学研究科博士後期課程中退後、同大学院工学研究科情報システム工学専攻助手、2002年同大学院情報科学研究科マルチメディア工学専攻助手、2004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准教授となり、現在に至る。工学博士。1996年本学会山下記念研究賞受賞。2000年電気通信普及財団テレコムシステム技術賞受賞。2003年本学会研究開発奨励賞受賞。2008年、2009年本学会論文賞受賞。モバイルコンピューティング、ネットワーク環境におけるデータ管理技術に関する研究に従事。IEEE、ACM、電子情報通信学会、日本データベース学会の各会員。



西尾章治郎 (フェロー)

1975年京都大学工学部数理工学科卒業。1980年同大学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、2003年より大阪大学大学院情報科学研究科長、その後2007年より大阪大学理事・副学長に就任。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、Data & Knowledge Engineering等の論文誌編集委員。本会理事を歴任。本会論文賞を受賞。電子情報通信学会フェローを含め、ACM、IEEE等8学会の各会員。