

様々な符号化形式に対応可能なP2P型 オンデマンドビデオ配信方式

武田 光平 渡辺 真太郎 三部 靖夫

株式会社 NTT データ 技術開発本部

〒104-0033 東京都中央区新川 1-21-2 茅場町タワー

あらまし

本稿では、Multiple Description Coding とビデオソースの時間方向分割を併用した P2P オンデマンドビデオ配信方式を提案する。本方式を用いることにより、安定したピア間の接続が期待できない環境でも再生品質を確保でき、さらに様々なクライアントの要求形式に対応可能な配信システムが構築できる。シミュレーションの結果、本方式はユーザリクエストが頻繁な環境下において、再生品質を大きく改善することを確認した。

和文キーワード オンデマンドビデオ, P2P, マルチプルデスクリプションコーディング, トランスコーディング

A method for video-on-demand services using a P2P framework adapting to various video codecs

Kohei Takeda Shintaro Watanabe Yasuo Sambe

R&D Headquarters, NTT DATA Corporation

1-21-2 Shinkawa, Chuo-ku, Tokyo, Japan

Abstract

In this paper, we propose a new P2P on-demand video distribution method, which employs Multiple Description Coding (MDC) and periodical video source division. In our method, as video streams are transmitted through multiple paths, video playback can be made more robust. Furthermore, by generating those video streams dynamically, various client requirements for video format can be satisfied. Our simulation results show that our method significantly improves playback quality, especially when user requests occur frequently.

英文 key words video-on-demand, P2P, multiple description coding, transcoding

1 はじめに

高速な ADSL 回線の普及により、インターネット上で高精細なビデオデータを配信することが可能になった。特にビデオのオンデマンド配信は、テレビなどの既存放送手段では実現しにくいこともあり、ブロードバンド時代のキラアプリケーションとして囑望されている。

しかし、一般にインターネット上のデータ伝送では混雑によるパケットの欠落・遅延が生じるため、サービス品質を確保することは困難である。また、高精細なビデオデータの配信には多大なネットワーク帯域を要し、配信コストの増大が問題となっている。

配信サーバのネットワーク帯域を軽減する方法の一つに、P2P 型の配信がある。P2P 型の配信は、視聴ピア自身がさらに別のピアに配信を行うことを特徴としている。これによってサーバへのトラフィック集中が回避され、また視聴ピア数が増加してもシステムの全体性能が低下しない利点がある。しかしながら、P2P ネットワークでは頻繁なピアの加入・脱退が繰り返されるためにピア間の接続が安定せず、データの安定した伝送という点では課題が残る。

安定したデータ伝送ができない環境下でロバストなビデオストリーム再生を行う方法の一つに、Multiple Description Coding (MDC) がある。MDC とは、ビデオデータを互いに相関する複数のストリーム（デスクリプション）に符号化する手法の総称である。

本稿では、配信コストの低減とビデオストリーム再生のロバスト化を共に図るため、MDC とソースビデオデータの時間方向分割を併用した P2P オンデマンドビデオ配信方式を提案する。

2 従来方式とその課題

昨今のネットワーク接続環境の広帯域化を受け、より大量かつ高ビットレートのビデオコンテンツがインターネット上で配信されるようになった。しかしながら、配信サーバの帯域幅や処理能力に対する要求も同様に増大したため、これを軽減するための種々の方式が提案されている。

Chawathe ら [1] は、インターネット上に複数設置されたストリーミングプロキシ間を信頼性の高い TCP 接続で結び、プロキシから視聴ピアに対して IP マルチキャストによる配信を行う方式を提案した。また、Deshpande ら [2] は視聴ピア間を P2P ネットワークで結び、その上でアプリケーションレベルのマルチキャストによりストリーム配信を行う方式を提案した。さらに Padmanabhan ら [3] はビデオデータに MDC を適用し、各デスクリプションを異なるアプリケーションレベルのマルチキャストツリーに流すことでロバストな配信を可能とする方式を提案している。

以上の方式はいずれも、ライブストリーミングやニアビデオオンデマンド (NVOD) におけるサーバの帯域

幅と処理負荷を大きく削減することができる。しかしこれらの方式はマルチキャストを用いる性質上、完全な VOD のように多種類のコンテンツを個別に要求される状況では、それほどサーバの帯域・処理負荷を削減できない課題がある。

一方、サーバの帯域幅と処理負荷を軽減する別のアプローチも存在する。Hafeeda ら [4] の方式や EdgeBurst 社 [5] の方式では、コンテンツをあらかじめ時間軸方向に分割して多数の断片（セグメント）を作成し、それらを配信ピア群にキャッシュさせる。このアプローチではクライアントがそれぞれ独立にセグメントをリクエストして再生を行うため、オンデマンド配信を容易に実現できる。しかしその反面、クライアントがあるセグメントの再生を終了して次をリクエストする際、リクエスト先のピアがすでに他のクライアントに占有されている可能性がある。あるいは、すでにピアがオフライン状態に移行した後であることも考えられる。このような場合は再生途中で映像が途切れるため、視聴者に不快感を与えるという課題がある。

3 提案方式

2 章において述べた通り、マルチキャスト型のビデオ配信方式ではオンデマンド配信時にサーバの帯域・処理負荷を削減することができない。またセグメントキャッシュ型のビデオ配信方式では、あるセグメントの再生から次のセグメントの再生に移る際、映像に途切れの生じる可能性がある。

そこで本稿では、セグメントキャッシュ型の P2P 配信をベースにすることでオンデマンド配信時のサーバ帯域・処理負荷削減効果を高め、セグメントを動的に MDC のデスクリプションに変換しながら配信することで映像を途切れにくくする方式を提案する。

提案方式では、一つのセグメントを複数のデスクリプションにエンコードし、それぞれを異なるピアから同時に送信する。これにより、クライアントはいずれかが欠けても他のデスクリプションで補って再生を継続できる。

さらに提案方式では、デスクリプションを作成する際にリサイズやフレームレート、ビットレートの変更などを行うことも可能である。これにより、単一のビデオソースを用いて、様々なクライアントの要求フォーマットに対応可能なビデオ配信システムを実現することができる。

この提案方式の具体的な手順について、図 1 を用いて説明する。

STEP 1 クライアントピアは、視聴するコンテンツのセグメントもしくはデスクリプションを探索し、その結果に基づいてサーバピアにデスクリプションをリクエストする（図 1 の (1)）

STEP 2 リクエストを受けた各サーバピアは、要求され

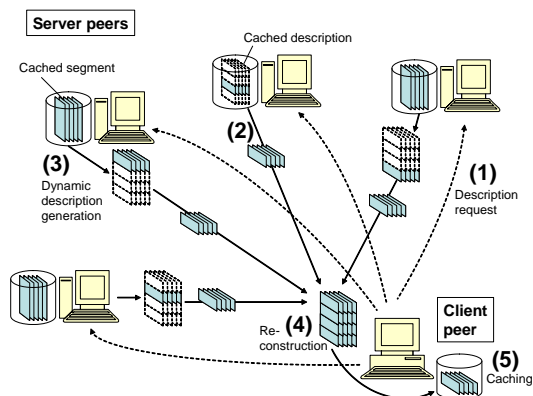


図 1: ストリーム再生の概観

た形式のデスクリプションを実際に所持しているか確認し、もし所持していればそのまま送信する。(図 1 の (2))

STEP 3 サーバピアがデスクリプションを所持していない場合は、セグメントを要求された形式にリアルタイムで変換しながら送信する(図 1 の (3))

STEP 4 クライアントピアは受信できているだけのデスクリプションをデコード・合成し、画面に表示する(図 1 の (4))

またデスクリプションを受信できなかったサーバピアへの接続を切断し、再度別のサーバピアに対してリクエストを行う。

STEP 5 クライアントピアは、完全に受信に成功したデスクリプションをストレージにキャッシュし、将来自分自身がリクエストを受ける場合に備える。(図 1 の (5))

以上により、安定したデータ伝送ができない環境下でもロバストなストリーム再生を行うことができ、かつ様々なクライアントの要求フォーマットに対応可能な P2P 配信方式が実現できる。

以下、3.1 でセグメント及びデスクリプションの生成方法 (STEP3) について、3.2 でサーバピアにおけるストリーム送信手順 (STEP2,3) について、また 3.3 でクライアントピアにおけるデスクリプションの受信と再生手順 (1,4,5) について詳細に述べる。

3.1 セグメント及びデスクリプションの生成

本節では、ソースビデオデータからセグメント及びデスクリプションを生成する方法について述べる。図 2 は、ソースビデオデータとセグメント、デスクリプションの関係を示した模式図である。

一般に圧縮ビデオデータにはシークを可能とするため、途中からデコードを開始できる点がいくつか設定

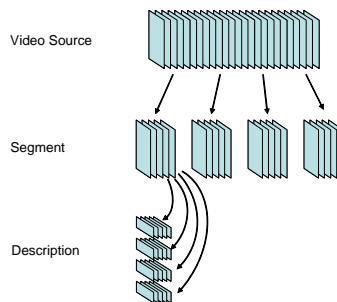


図 2: ビデオデータの分割方法

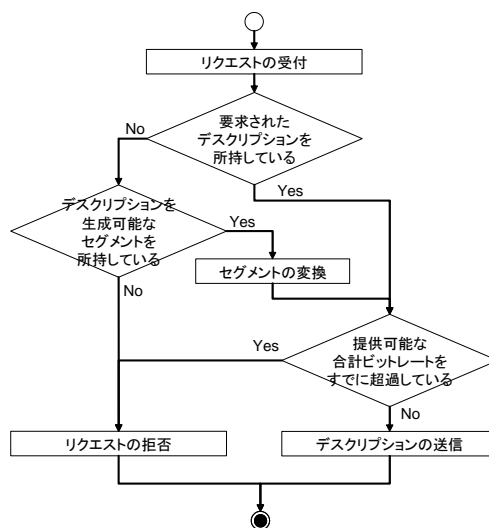


図 3: サーバピアの動作手順

されている。例えば、MPEG1/2 における Closed GOP, MPEG4 における Closed GOV などがそうである。提案方式ではこの位置でソースビデオデータを分割することにより、各セグメントを独立にデコードできるようにする。

また、デスクリプションを生成するには、一旦セグメントをデコードし、リサイズやフレームレートの変更を施した上で再度 MDC を用いてエンコードする。MDC の方式としては、ブロック変換係数の一次変換によるもの [6], Wavelet 変換によるもの [7] などが提案されているが、提案方式では MDC の方式は問わない。

3.2 サーバピアの動作手順

本節では、サーバピアの動作について述べる。ここで言うサーバピアとは、あるコンテンツ再生セッションにおいて、デスクリプションを送信する役割を持つピアのことである。

図 3 に、サーバピアがリクエストを受けた後のフローを示す。大まかな動作の流れとしては、リクエストを受け付けるとまず送信の可否を判定し、可能ならデスク

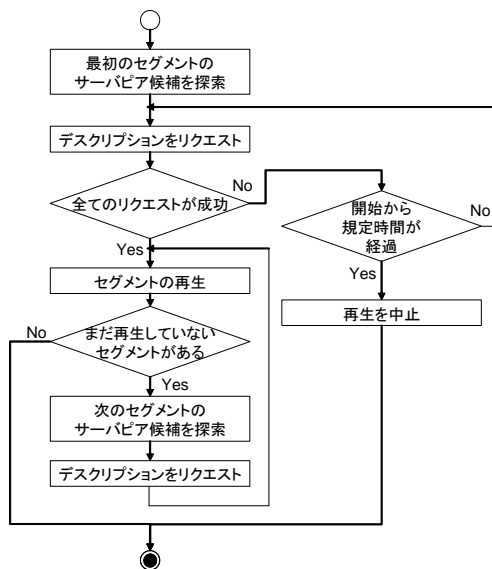


図 4: クライアントピアの動作手順

リプションを送信，不可能なら拒否メッセージを返す手順になる。

3.3 クライアントピアの動作手順

本節では，クライアントピアの動作手順について述べる．クライアントピアとは，あるコンテンツ再生セッションにおいて，デスクリプションを受信・再生する役割を持つピアのことである．

クライアントピアのフローを図 4 に，再生時のブロック図を図 5 に示す．大まかな動作の流れとしては，先頭セグメントについてのみ全てのデスクリプションのリクエストが成功してから再生を開始し，それ以降は探索とリクエストを繰り返しながら再生を行う手順になる．

再生中は，一定レートでデスクリプションをデコードし，それらを合成して表示する．表示タイミングにおいて必要なデスクリプションが受信できていなかった場合は，他のデスクリプションを用いて補間する．また，そのピアとの接続を打ち切って別の候補に再度リクエストを行う．

クライアントピアがデスクリプションを無傷で受信し終えた場合，そのまま自身のストレージにキャッシュする．キャッシュしたデータは将来自身がリクエストを受けた際に用いる．

4 シミュレーション

P2P ネットワーク環境における提案方式のロバスト性を検証するためには，実際にビデオ配信を行ってどれだけのデスクリプションを同時に受信できるか，またどれだけの再生途絶時間が生じるのかを調査することが

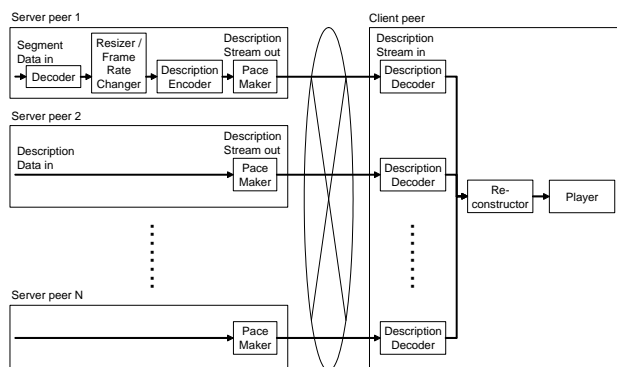


図 5: 再生時のブロック図

表 1: ピアの条件

ピア数	20
トポロジー	スター型
バンド幅	上り 0.8 (Mbps) 下り 4 (Mbps)
一つのピアが提供可能な合計ビットレート	0.6 (Mbps)
ネットワーク遅延	10 (ms)
データ伝送プロトコル	TCP
ピア寿命	平均 120 (s) の指数分布
再生要求の生起間隔	40 (s) ~ 70 (s) の等間隔
総シミュレーション時間	6000 (s)

必要である．しかしながら，現実には多数のコンピュータを揃えて配信実験を行うことは困難であるため，シミュレーションにより提案方式の有効性を評価した．

4.1 シミュレーション条件

コンピュータ間の通信とビデオ再生を模擬するため，ネットワークシミュレータ ns-2[8] を用いた．シミュレーションに用いるピアの条件は，現在の ADSL 環境を想定して表 1 のように定めた．また，P2P ネットワークでは頻繁なピアの脱退が起るため，これを模擬する手段としてピアに寿命を定めた．サーバピアはデスクリプションの送信を開始してから寿命分の時間が経過すると通信を切断するものとした．配信するビデオコンテンツの条件については，表 2 のように定めた．

表 2: 配信コンテンツの条件

コンテンツのセグメント数	10
一セグメントあたりの再生時間	60 (s)
一セグメントから生成される デスクリプション数 (N)	$N = 1 \sim 4$
デスクリプションの 再生ビットレート	$600 / N$ (Kbps)
デスクリプションの 送信ビットレート	$600 / N$ (Kbps)
再生フレームレート	29.97 (fps)

4.2 シミュレーション結果と考察

本節では、提案方式のシミュレーション結果を、従来方式と比較しながら示す。なお、提案方式において生成デスクリプション数を1とするとMDCを用いない従来のセグメントキャッシュ型配信方式と等しくなるため、これを比較対象として用いた。

図6に再生時間全体に占める、動画再生が途絶していた時間の割合を示す。また図7に送信されたデスクリプションが実際にどれだけ再生に用いられたかを示す。

図6によれば、従来方式ではユーザの要求が頻繁に起こるほど再生途絶時間が長くなるが、提案方式では最悪でも1割以下の再生途絶しか起こらないことが分かる。これは、MDCを用いることにより、全体としてのストリーム受信状態が改善されたからである。また図7からは、再生途絶が無いだけでなく、送信されたデスクリプションが効率的に利用されていることが分かる。なお、ユーザリクエスト間隔が60秒と70秒の場合に従来方式の性能が逆転しているように見える。これはセグメントの再生時間がちょうど60秒であり、リクエスト間隔が同じ60秒の場合に再生終了と次のリクエストがうまく前後して起きたためであると考えられる。

再生途絶が起こらないもうひとつの理由として、提案方式のストリーム再生開始条件が過剰なリクエストを遮断する機構として働いていることが考えられる。

そこで、ユーザのコンテンツ再生要求回数に占める、実際に再生が行われた回数の割合を調べた。これを図8に示す。これを見ると、提案方式がユーザの再生要求を受け入れる確率は従来方式に比べて低いことが分かる。その原因として、提案方式では最初に全てのデスクリプションのリクエストが成功しなければ再生を開始しないことが挙げられる。つまり、提案方式ではシステム的能力を超えたリクエストをあらかじめ遮断することによって再生を安定させていると考えられる。

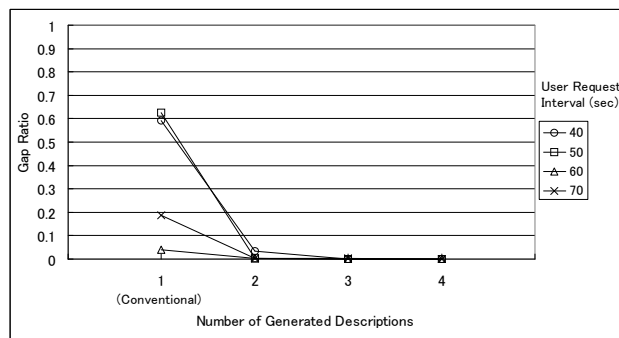


図 6: 全再生時間に占める再生途絶時間の割合

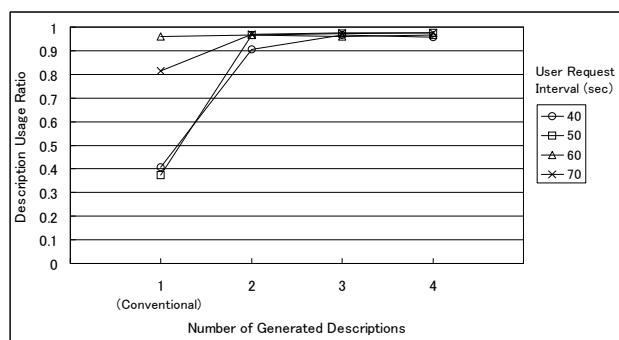


図 7: 送信されたデスクリプション数に占める、実際に再生に用いられたデスクリプション数の割合

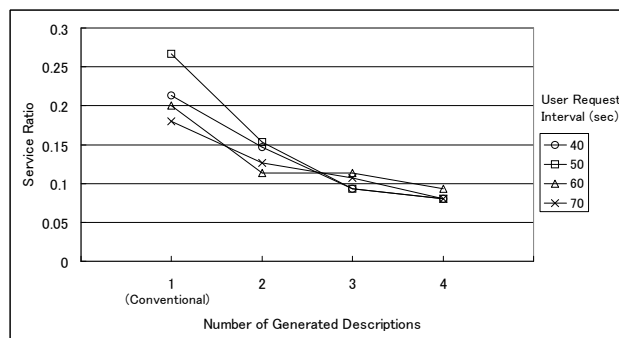


図 8: 全ユーザ再生要求回数に占める、実際に再生が行われた回数の割合

さらにリクエスト遮断の影響を確かめるため、図9, 10, 11に、リクエスト間隔が40秒の場合について再生開始条件を「半数以上のデスクリプション要求が成功すること」とした結果を示す。

これによれば、確かに再生が開始される確率が高まると、再生品質が低下することが分かる。しかしながらデスクリプション数2の場合を見ると、再生回数が従来手法を上回っているにもかかわらず、再生途絶時間、デスクリプション利用率とも従来手法より良好である。すなわち、リクエスト遮断の効果を除いてもなお、MDCによる再生品質の改善効果があることが確認できた。

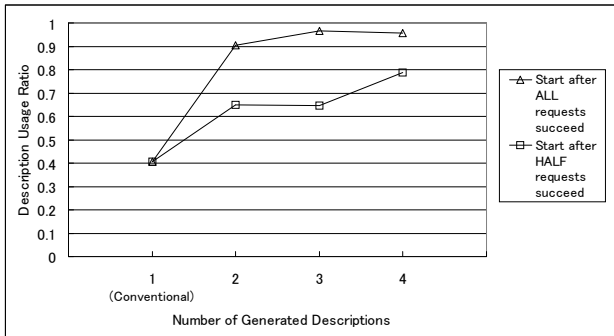


図 9: 再生開始条件の違いが、再生途絶時間に与える影響

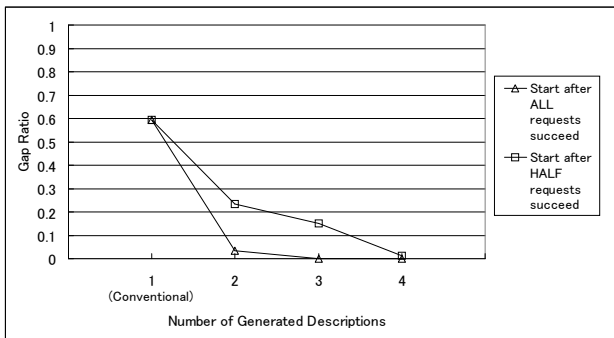


図 10: 再生開始条件の違いが、再生に利用されるデスク립ション数に与える影響

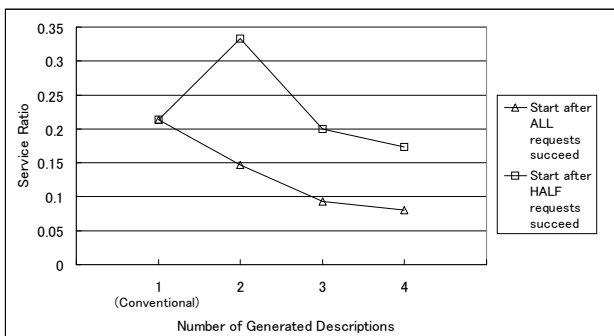


図 11: 再生開始条件の違いが、再生要求受け入れ数に与える影響

5 おわりに

本稿では、P2P 型オンデマンドビデオ配信における再生途絶の問題を解決するため、MDC とビデオソースの時間分割を利用した方式を提案した。

そしてシミュレーション結果より、提案方式は頻りにコンテンツの再生要求が起こる環境下で再生品質を大きく改善することが確認された。しかしその一方で、デスク립ション生成数を大きくすると再生要求の受け入れ率が低下する現象も見られた。

今後は、実際のサービスにおいて重要な性能指標となる再生要求の受け入れ率と再生品質の釣り合いをどう取るかについて、検討していく必要がある。

参考文献

- [1] Y. Chawathe, S. McCanne, and E. A. Brewer, "An architecture for internet content distribution as an infrastructure service", February 2000. Unpublished work.
- [2] H. Deshpande, M. Bawa, and H. Garcia-Molina, "Streaming live media over a peer-to-peer network," in Work at CS-Stanford. Submitted for publication, 2002
- [3] V. N. Padmanabhan, H. J. Wang, P. A. Chou, and K. Sripanidkulchai, "Distributing streaming media content using cooperative networking", in ACM/IEEE NOSSDAV, Miami, FL, USA, May 12-14 2002.
- [4] M. Hefeeda, B. Bhargava, and D. Yau, "On-demand media streaming over the internet", Technical report, CERIAS TR 2002-20, Purdue University, June 2002.
- [5] EdgeBurst,
<http://jibe.ws/aboutJibe/press-102102-EdgeburstDeliverySystem.htm>
- [6] Y. Wang, M. T. Orchard, and A. R. Reibman, "Multiple description image coding for noisy channels by pairing transform coefficients", In Proc. IEEE Workshop on Multimedia Sig. Proc., pp. 419-24, June 1997.
- [7] S. Servetto, K. Ramchandran, V. Vaishampayan, and K. Nahrstedt, "Multiple-Description Wavelet Based Image Coding", In Proceedings of the IEEE International Conference on Image Processing, Chicago, IL, 1998.
- [8] The Network Simulator ns-2,
<http://www.isi.edu/nsnam/ns/>

株式会社 NTT データ 技術開発本部
〒 104-0033 東京都中央区新川 1-21-2 茅場町タワー
Tel.03-3523-8121 Fax.03-3523-8131