

MPEG-4 動画配信のための 品質調整機能を組み込んだプロキシキャッシングシステムの実装と評価

谷口 義明[†] 上岡 功司^{††} 若宮 直紀[†] 村田 正幸[†] 野田 文雄^{††}

[†] 大阪大学 大学院情報科学研究科 〒 560-8531 大阪府豊中市待兼山町 1-3

^{††} (株)日立製作所 システム開発研究所 〒 215-0013 神奈川県川崎市麻生区王禅寺 1099 番地

E-mail: [†]{y-tanigu,wakamiya,murata}@ist.osaka-u.ac.jp, ^{††}{at-ueoka,noda}@sdl.hitachi.co.jp

あらまし WWW システムで広く用いられているプロキシ技術を適用することにより、システムに大きな負荷を与えることなく、実時間で応答性の高い動画配信サービスが実現できると考えられる。さらに、プロキシで蓄積データを適切に品質調整することにより、ネットワークへの接続形態、システム性能、再生動画像に対する利用者の好みなどによりさまざまに異なる要求品質にあわせた動画配信が実現可能である。本稿では、MPEG-4 動画配信のための品質調整機能を組み込んだプロキシキャッシングシステムを提案、設計、実装し、トラフィック量、再生動画像品質の観点から評価を行った。その結果、本システムを用いることで、利用可能な帯域にあわせて、実時間で動画像品質調整を行う動画配信が提供可能であることを示した。

キーワード 品質調整, 動画配信サービス, プロキシキャッシング, MPEG-4

Implementation and Evaluation of Proxy Caching System for MPEG-4 Video Streaming with Quality Adjustment Mechanism

Yoshiaki TANIGUCHI[†], Atsushi UEOKA^{††}, Naoki WAKAMIYA[†],

Masayuki MURATA[†], and Fumio NODA^{††}

[†] Graduate School of Information Science and Technology, Osaka University

1-3 Machikaneyama, Toyonaka, Osaka 560-8531, Japan

^{††} Systems Development Laboratory, Hitachi, Ltd.

1099 Ohzenji, Asao-ku, Kawasaki, Kanagawa 244-0817, Japan

E-mail: [†]{y-tanigu,wakamiya,murata}@ist.osaka-u.ac.jp, ^{††}{at-ueoka,noda}@sdl.hitachi.co.jp

Abstract The proxy mechanism widely used in WWW systems offers low-delay and scalable delivery of data by means of a “proxy server”. By applying proxy mechanism to video streaming systems, high-quality and low-delay video distribution can be accomplished without imposing extra load on the system. In addition, the quality of cached video data can be adapted appropriately in the proxy when clients are heterogeneous, in terms of the available bandwidth, end-system performance, and user preferences on the perceived video quality. In this paper, we propose, design, implement, and evaluate a proxy caching system for MPEG-4 video streaming services. Through evaluations conducted from several performance aspects, we proved that our proxy caching system can provide users with a continuous and high-quality video streaming service in a heterogeneous environment.

Key words Quality Adjustment, Video Streaming Service, Proxy Caching, MPEG-4

1. はじめに

近年、ストリーミングサービスなどの動画データ配信を行うマルチメディアアプリケーションの利用が増加してきている。しかしながら、現在のインターネットは、エンド間の帯域、遅延、パケットロス率といった通信品質を保証しないため、ストリーミングサービスにおいては、動画データの転送遅延や欠落により、再生の停止や途切れ、ちらつきなどが発生する。また、一般的なサーバ/クライアント型の動画配信サービスでは、利用者数の増加や利用者の環境、すなわちネットワークへの接続回線の容量や動画サーバへの経路の輻輳状態、クライ

アントシステムの処理能力などへの適応性が低い。

ネットワークやサーバの負荷を軽減し、応答性の高い動画配信サービスを提供するためには、WWW システムで今日広く用いられているプロキシ技術を動画配信システムに適用することが有効である [1-4]。さらに、プロキシにおいて蓄積データを適切かつ柔軟に品質調整することができれば、ネットワークへの接続形態や輻輳状態、システム性能、再生動画像に対する利用者の好みなどが様々に異なるそれぞれのクライアントに、適切な品質の動画を効率よく配信することが可能になる。

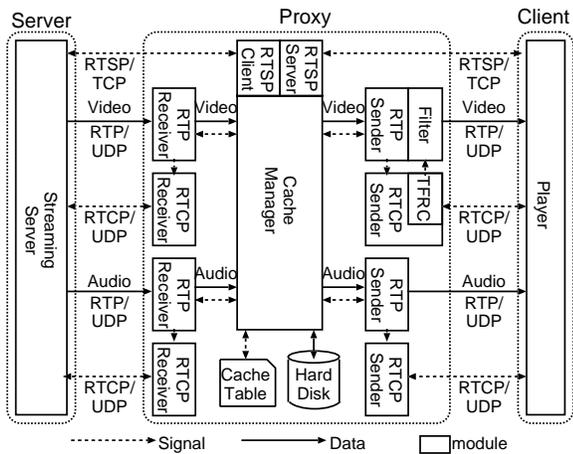


図 1 実装システムの構成
Fig. 1 Modules Constituting System

本稿では、適切な品質の動画を途切れなくクライアントに提供することのできる MPEG-4 動画配信のための品質調整機能を組み込んだプロキシキャッシングシステムの設計、実装を行う。本システムを用いることにより、利用者数や利用者環境に対する適応性の高い動画配信が可能となる。本システムは、文献 [5] の提案に基づいており、動画ストリームは、キャッシュバッファや伝送帯域の有効利用を図るため、一定時間ごとのブロックに分割される。プロキシは、ブロック単位で動画ストリームをサーバから取得、キャッシュバッファへ蓄積し、利用者環境を考慮した品質調整を施した後に、クライアントへ提供する。また、動画ブロックの先読み機構によりキャッシュミスによる遅延の増加を抑え、ブロック置き換え機構により有限容量のキャッシュバッファを効率良く利用する。

本稿では、トラフィック量の変化、キャッシュ内ブロック量、再生動画品質などの観点から評価を行い、本システムの実用性を検証する。実験の結果、本システムを用いることで、ネットワークの輻輳状態の変化にあわせて高品質な動画を途切れなく配信できることを示す。

以下、2 章では、実装したプロキシキャッシングシステムについて述べる。3 章で本システムの実用性、有効性を検証する。最後に、4 章で本稿のまとめと今後の課題について述べる。

2. 動画品質調整機能を組み込んだプロキシキャッシングシステム

本システムの構成を図 1 に示す。プロキシは四角で表された構成要素からなり、クライアントごとに生成される。また、実線の矢印は動画データの流れを、破線の矢印は制御情報の流れを表している。本システムでは、動画配信制御に RTSP/TCP を用いている。また、動画データ転送には映像データ、音声データ別に RTP/UDP およびフィードバック情報収集のための RTCP/UDP セッションが設定される。

2.1 基本動作

図 2 に本システムの動作例を示す。動画ストリームは、キャッシュバッファや伝送帯域の有効利用を図るため、一定時間毎のブロックに分割される。MPEG-4 動画は、フレームに相当する VOP (Video Object Plane) を単位として RTP パケット化されるため [6]、本稿における実装システムでは、1 ブロックの大きさを 300 VOP としている。例えば、動画ストリームのフレームレートが 30 フレーム毎秒の場合、先頭から

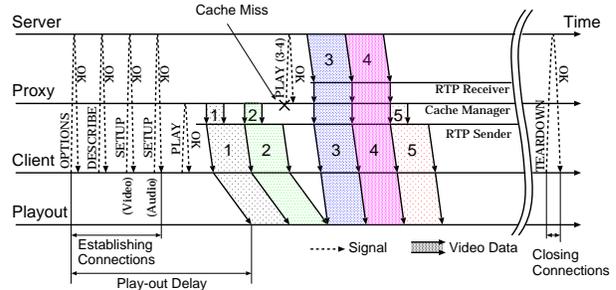


図 2 基本動作

Fig. 2 Basic Behavior of Our System

3 番目のブロックを取得するには RTSP PLAY メッセージを用いて Range 20-30 とすればよい。なお、映像ブロックは、映像ブロックと、対応する音声ブロックからなる。

まず、クライアントは RTSP OPTIONS, DESCRIBE, SETUP メッセージを順次プロキシに送ることで、映像および音声ストリーム配信のためのセッションをそれぞれ確立する。Cache Manager (図 1) は RTSP Server を介して受信した RTSP メッセージを動画サーバに中継することにより、動画サーバ/プロキシ間のセッションを確立する。動画サーバ/プロキシ/クライアント間それぞれのセッションの確立後、クライアントは RTSP PLAY メッセージをプロキシに送り、動画ストリームの転送を要求する。

Cache Manager は、Cache Table と呼ばれる表を用いてキャッシュした動画ブロックの情報を管理している。Cache Table は、ブロック番号、ブロックサイズ、および、ブロックを動画サーバから取得中かどうかを表すフラグからなる。なお、ブロックがキャッシュに蓄積されていない場合のブロックサイズは 0 とする。RTSP Server をとおして動画ストリームの転送要求を受けた Cache Manager は、キャッシュされたブロックを順次読みだし RTP Sender に渡す。動画ブロックのうち映像ブロックは TFRC モジュールの推定した利用可能な帯域にあわせて Filter によって品質調整された後、クライアントに転送される。また、Cache Manager は、サーバ/プロキシ間の帯域を利用してブロックの先読みを行う。

必要なブロックがキャッシュに無い場合には、プロキシは、RTSP PLAY メッセージの Range フィールドを用いて、動画サーバから動画ブロックを取得する。ただし、動画は先頭から順に再生、鑑賞されることを考慮し、伝送帯域の有効利用のため、連続したキャッシュされていないブロックはまとめて取得する (図 2 中、ブロック 3, 4)。サーバにブロック転送要求を行う際には、Cache Table 内のフラグがセットされる。

Cache Manager は、動画サーバから RTP Receiver をとおして取得した動画ブロックを、VOP 単位で RTP Sender に渡す。動画ブロックの受信が完了すると、Cache Manager は、Cache Table 内のフラグをリセットし、取得したブロックをキャッシュバッファに蓄積する。取得したブロックを蓄積するための十分な空きがキャッシュバッファにない場合、プロキシは取得したブロックとキャッシュ内ブロックの置き換えを行う。

プロキシは、RTSP TEARDOWN メッセージをクライアントから受信すると、動画サーバにメッセージを転送し、各セッションを切断する。

2.2 ブロックの先読み機構

キャッシュヒットの場合、Cache Manager は、サーバ/プロキシ間の帯域を利用してブロックの先読みを行う。続く P 個の

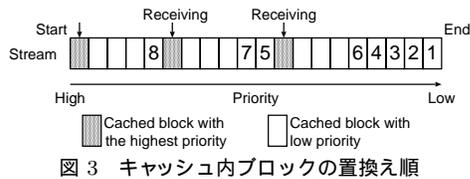


図 3 キャッシュ内ブロックの置換え順
Fig. 3 Priority of Cached Blocks

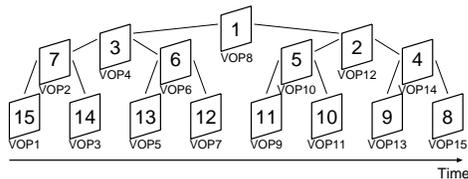


図 4 フレーム棄却順
Fig. 4 Frame Discarding Order

ブロックについて順に蓄積状態を調べ、キャッシュされていないブロックがあれば、動画サーバから取得し、将来のキャッシュミスを防ぐ。なお、キャッシュミスの場合と同様に、連続したキャッシュされていないブロックはまとめて取得する。

2.3 ブロックの置き換え機構

まず、*Cache Manager* は、蓄積されている動画ストリームの中から最も優先度の低い動画ストリームを LRU アルゴリズムにより選択する。次に、選択したストリーム内の各ブロックに優先度を与える。サーバから取得中のブロックおよび動画ストリームの先頭に位置するブロックは、優先度は最も高いものとする。その結果、図 3 に示されるように優先度の高いブロックに囲まれた領域ができる。これらのうち、より大きい領域の、より後ろに位置するブロックほど優先度が低いものとする。新たに取得したブロックが蓄積可能になるまで優先度の最も低いブロックから順に棄却する。図 3 にブロックの棄却順の例を数字で表す。

2.4 TFRC によるレート制御

TFRC は「同一ネットワークパス上における、非 TCP コネクションの得るスループットと TCP の得るスループットが等しい」という TCP-friendly の概念に基づき、帯域使用において TCP と公平なマルチメディア通信を実現するレート制御手法である [7]。TFRC では、RTT やロスイベント率といったネットワークの負荷状態をあらゆるフィードバック情報に基づき、同一パス上の TCP セッションの得るスループットを推定し、マルチメディアデータの送出レートを決定する。本システムでは、RTCP によりクライアントからのフィードバック情報を取得し、RTCP Receiver Report メッセージから得られる累積パケットロス数、受信した RTP パケットの最大のシーケンス番号からパケットロス率を、また、DLSR, LSR の値と RTCP Receiver Report メッセージが *RTCP Sender* に到着した時刻から RTT を、それぞれ計算する。

2.5 動画品質調整

本システムでは、動画品質調整手法としてフレーム棄却フィルタを用いる。フレーム棄却フィルタは適当な数のフレームを棄却することで、動画品質およびデータ量を調整する。再生動画の動きのなめらかさは失われるが、ローパスフィルタや再量子化フィルタに比べて、単純で処理負荷が小さい。

MPEG-4 において VOP を棄却する際は、VOP 間の依存関係を考慮する必要があるため、まず、*Filter* は 1 秒分の VOP を蓄積する。次に、バランスよく VOP を棄却して品質調整後

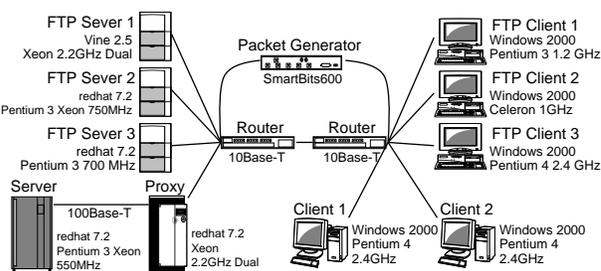


図 5 実験システム構成図
Fig. 5 Configuration of Experimental System

の再生をなめらかにするため、優先度を設定する。例えば、15 フレーム毎秒の動画の場合には、中央に位置する 8 番目の VOP を根とした二分木を作成し、根から順に幅優先で優先度を高くしていく。図 4 に、15 フレーム毎秒の動画ストリームにおける VOP の優先度順の例を示す。図中、それぞれの四角形は VOP を表し、四角形の中の数字は VOP の優先度を示す。なお、数字の小さいものほど優先度が低く、棄却されやすい。B-VOP のうち優先度の低い VOP から順に棄却することにより、利用可能な帯域で送信可能なよう動画データ量を調整する。全ての B-VOP を削除しても、動画レートが利用可能な帯域を上回る場合は、P-VOP を同様に順次棄却していく。ただし、再生動画の著しい品質劣化を防ぐため、I-VOP の棄却は行わない。

3. 実装システムの実験評価

実験システムの構成を図 5 に示す。2 台のクライアントが 2 台のルータを介してプロキシに繋がれており、3 対の FTP サーバ/クライアント、およびパケットジェネレータが、ルータ間のリンク容量を動画配信と競合する。実験には、平均 1 Mbps の符号化レートで MPEG-4 VBR 符号化された 320×240 画素、30 フレーム毎秒の映像ストリームと 96 kbps の音声ストリームが多重化された 600 秒、約 80 MByte の動画ストリーム 1 本を用いる。動画サーバには、動画ストリーム全体が蓄積されている。プロキシは、50 MByte のキャッシュバッファを有しており、はじめは何も蓄積されていない。なお、先読みブロック数 P は 5 とした。クライアント 1 が OPTIONS メッセージをプロキシに転送する時刻を 0 とし、時刻 150 秒にクライアント 2 が同じ動画に対する配信要求を行う。クライアントはいずれも一時停止や早送りなどの操作無しに 600 秒の動画を見続けた後、接続を終了する。時刻 300 秒から時刻 450 秒の間に 3 対の FTP サーバ/クライアント間で TCP セッションによるファイル転送が行われる。なお、パケットジェネレータは、常に 8 Mbps の UDP トラフィックを送出している。

比較の対象として、動画品質調整によるレート制御を行わない場合についてもあわせて評価した。

3.1 受信レートの変動の評価

クライアント 1、および 2 の動画データ受信レートの変化の様子を、動画品質調整によるレート制御を行った場合について図 6 に、なにも行わなかった場合について図 7 に示す。また、図 8 には、それぞれの実験における FTP セッションによるファイル転送の受信レートを示す。また、クライアント 1 の RTCP Receiver Report から計算したパケットロス率を動画品質調整を行った場合と行わなかった場合について、図 9 に示す。なお、実験に用いたクライアントアプリケーションは、RTCP Sender Report の受信とは無関係に 3~6 秒ごとに

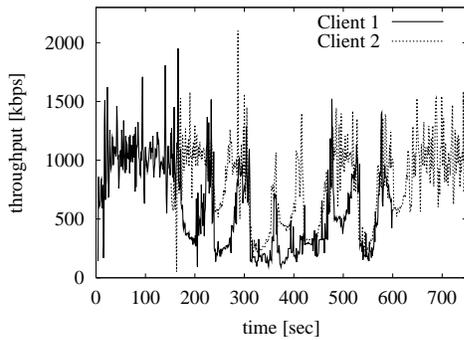


図 6 動画品質調整を行った場合の受信レート変動

Fig. 6 Reception Rate Variation with Quality Adjustment

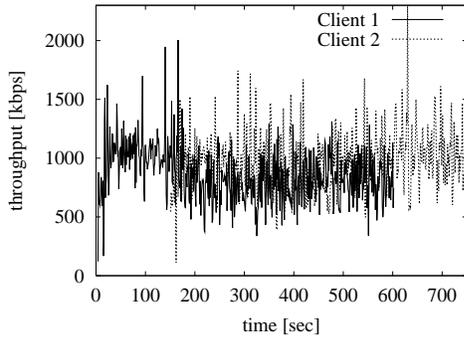


図 7 動画品質調整を行わなかった場合の受信レート変動

Fig. 7 Reception Rate Variation with Traditional Method

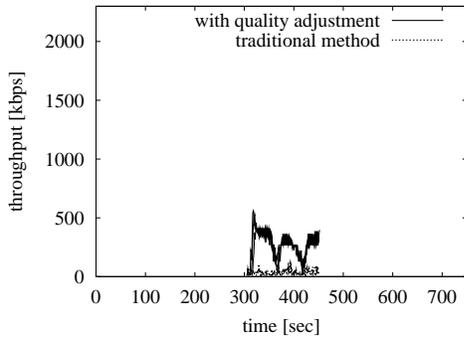


図 8 FTP スループットの変化

Fig. 8 Reception Rate Variation on FTP Sessions

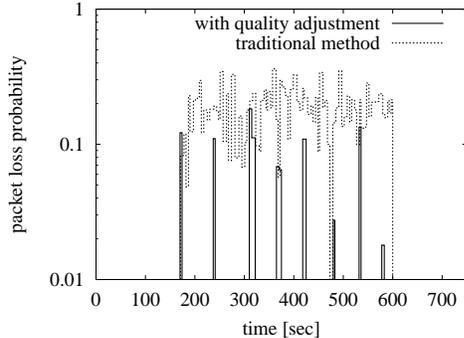


図 9 クライアント 1 におけるパケットロス率の変化

Fig. 9 Packet Loss Probability Variations at Client 1

RTCP Receiver Report を送出する。

図 6 および図 8 より、レート制御を行わない場合には、TCP のスループットが著しく低下するのに対して、TFRC に基づくレート制御にあわせて動画品質調整を行った場合には、クライアント間、および TCP セッションとの間で帯域共有が実現されていることがわかる。また、図 9 に示されるように、ネットワークの輻輳状態に応じたレート制御によりパケット棄却が

大きく抑えられていることがわかる。

3.2 キャッシュ内ブロック量の変動の評価

紙幅の制限のため、図には示さないが、キャッシュ内データ量がキャッシュ容量に達する時刻 370 秒以降で置き換えが発生した。しかしながら、クライアント 1 が動画配信を終了する時刻 600 秒以降は動画サーバからの新たな動画データ取得が発生せず、クライアント 2 に対しては、蓄積データを用いた配信を行っていることがわかる。したがって、本実験では、効果的なブロック置き換えが行われている。

3.3 動画品質変動の評価

本実験で用いたクライアントアプリケーションでは、ひとつでもパケットが失われると VOP 全体が棄却され、また、棄却された VOP を参照する VOP も復号、表示されない。このため、動画品質調整により意図的にフレームを間引いた場合に、クライアント 1 では 18000 フレーム中 9712 フレームが再生されたのに対し、品質調整によるレート制御を行わなかった場合に再生されたフレーム数は 9133 であった。したがって、ネットワークの輻輳状態に応じて動画品質調整によるレート制御を行うことにより、なめらかで高品質な動画再生が実現できる。

4. おわりに

本稿では、MPEG-4 動画配信のための品質調整機能を組み込んだプロキシキャッシングシステムの実装を行い、トラフィック量、キャッシュ内ブロック量、動画品質の観点からシステムの実用性について検証した。実験による評価の結果、ネットワークの輻輳状態の変化に応じて動的に動画品質調整を行うことにより、低遅延かつ高品質な動画配信が実現可能であることを示した。ただし、他の品質調整機構やサーバ、クライアントアプリケーションが混在する大規模なネットワークでの実証実験、利用者が停止、早送り、巻き戻しといった操作を行う場合の実験については、引き続き評価を行う必要がある。

文 献

- [1] R. Rejaie and J. Kangasharju, "Mocha: A quality adaptive multimedia proxy cache for internet streaming," in *Proceedings of NOSSDAV 2001*, June 2001.
- [2] K. Wu, P. S. Yu, and J. L. Wolf, "Segment-based proxy caching of multimedia streams," in *Proceedings of the 10th International WWW Conference*, pp. 36–44, May 2001.
- [3] B. Wang, S. Sen, M. Adler, and D. Towsley, "Optimal proxy cache allocation for efficient streaming media distribution," in *Proceedings of IEEE INFOCOM 2002*, Jun. 2002.
- [4] M. Reisslein, F. Hartanto, and K. W. Ross, "Interactive video streaming with proxy servers," *Information Sciences: An International Journal*, Dec. 2001.
- [5] Masahiro Sasabe, Yoshiaki Taniguchi, Naoki Wakamiya, Masayuki Murata, and Hideo Miyahara, "Proxy caching mechanisms with quality adjustment for video streaming services," to appear in *IEICE Transactions on Communications Special Issue on Content Delivery Networks*, Jan. 2003.
- [6] Y. Kikuchi, T. Nomura, S. Fukunaga, Y. Matsui, and H. Kimata, "RTP Payload Format for MPEG-4 Audio/Visual Streams," *Internet Request for Comments 3016*, Nov. 2000.
- [7] M. Handley, S. Floyd, J. Padhye, and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol specification," *Internet Request for Comments 3448*, Jan. 2003.