

RTP を利用した動画配信システムにおける QoS 制御方式

福田 和真, 奥村 誠司, 鷹取 功人, 大野 次彦, 下間 芳樹

三菱電機株式会社 情報技術総合研究所

本論文では, インターネット上で RTP を利用した動画配信システムにおいて, ネットワークの状態に応じ受信側で良い再生品質を実現するための QoS (Quality of Service) 制御方式を提案する. 時間制約のある連続メディア情報をベストエフォート型であるインターネット上で配信する場合, 常に再生品質が良いわけではなく, 安定した配信や良い再生品質を提供するため QoS 制御が必要となる. 提案する QoS 制御方式は, ネットワーク状態を的確に把握しその推移から転送レートを見積る輻輳予測・検出機能, その結果を踏まえネットワークの状態に適応したパケット送出を行う転送レート制御機能から構成される. これらの機能を配信サーバ上に実装し, インターネット上で測定を行った. 本論文では, その結果や評価, 提案した方式の有効性についても述べる.

A Method of QoS control on Media System by means of RTP

Kazuma Fukuda, Seiji Okumura, Norihito Takatori, Tsugihiko Ohno,
Yoshiki Shimotsuma

MITSUBISHI ELECTRIC CORPORATION
INFORMATION TECHNOLOGY R&D CENTER

In this paper, we propose a method of QoS (Quality of Service) control adapted network condition by means of RTP, which method enables Clients to playback streaming media better quality. Network condition, in general, wasn't always served best quality on streaming over the Internet. Therefore, it is necessary to consider QoS control in order to service better quality streaming. A method we proposed consists of the function to predict and detect congestion, and to control transport rate. We have implemented that method on Media System. We show results of the experiment and describe the effectiveness of that method.

1 はじめに

現在, インターネットの利用者数は増加し続けている. それに伴い, インターネット上で利用できるサービスも多種にわたり, インフラとしてのインターネットの重要性も増している. また最近では, インターネットを経由した音声や動画の配信サービスを利用する場面も多く見られる.

インターネット上におけるパケット通信は, 使用する帯域幅や遅延時間などが基本的に保証されない“ベストエフォート”型である. しかし, 音声や動画などのリアルタイム性を要求する連続メディア情報は帯域幅や遅延時間などの制約があるため, 経路上のさまざまな状況に適応し効率的に転送することで受信側で連続メディア情報の品質良い再生を行うことが重要となる.

そこで, 本論文ではそれらを実現する QoS 制御方式を提案する. 提案した方式を実現し, インターネット上での転送実験と評価を行い, その方式の有効性についても述べる. 提案する QoS 制御方式は, RTP (Real-time Transport Protocol, [1]) および RTSP (Real-Time Streaming Protocol, [2]) を使用して MPEG-4[3] コンテンツを配信するシステムを構築し, そのシステム上で実装した.

2 インターネット上における動画配信の課題

インターネット上での通信はベストエフォート型であり, 使用する帯域幅や遅延時間などは保証されない. しかし, 連続メディア情報は帯域幅や遅延時

間などに関して制約を持つ。

インターネット上の転送プロトコルとして、TCP などのコネクション型と UDP などのコネクションレス型がある。前者は信頼性のある転送を実現するため、欠損したパケットに対して再送などのエラー訂正機構やネットワーク状況に応じた配送を行う輻輳制御の機構を持つ。しかし、これらの機構による遅延の変動は大きくリアルタイム性を大きく損なう。後者はこれらの機能がなくシンプルで反応性は高いが、パケット欠損によるデータ欠落は連続メディア情報の再生品質の悪化となる。パケット欠損への対応として、エラー訂正や転送帯域の制御などの機構が必要となるが、それらの実現によりリアルタイム性が損なわれないようにする必要がある。

インターネット上で連続メディア情報を扱うためには、単にデータを転送するだけでなく遅延などの時間制約にも従う必要があり、リアルタイム性に関するさまざまな課題が存在することになる。

これらの解決のために多くの方式が提案されており、例えば、経路上の通信状態を把握しそれに依りて動的に転送レートや符号化レートを制御する方式、欠損したパケットを訂正するために再送や冗長な情報も同時に転送する方式、受信側のバッファ量やネットワーク上で転送中のデータ量を考慮して転送の制御を行う方式などがある。また、経路上の通信状態を把握するため、受信側から送信側への受信状況のフィードバックを行うことが多い。

3 動画配信システムの構成

まず動画配信システムの構成について述べる。その上に搭載する QoS 制御方式については後述する。

動画配信システムの基本構成は図 1 である。動画は MPEG-4、音声は MP3、G723.1 を使用し、それらを RTP ベイロードに格納して配信する。RTP ベイロードに MPEG-4 ストリームを格納するための標準化も行われており、本システムでは文献 [4] の仕様によるベイロードタイプである。なお、RTP は UDP 上に実装されている。ネットワーク再生時のサーバ/クライアント間でのセッション管理やストリーミング制御(再生・停止など)は、RTSP を使用¹している。

インターネット上を転送したデータを再生するときの処理の流れは次のようになる。

1. クライアントは、コンテンツの詳細情報や配信サーバ情報を SDP (Session Description Protocol, RFC2327) で記述した MVX (Movie Video eXtention) ファイル(独自)を配信サーバ/Web サーバ/ローカルホスト上から得る。

¹RTSP 標準のサーバやクライアントへの相互接続を可能としている。

2. クライアントは MVX に記述された情報を解析し、配信サーバへ RTSP セッションを確立し、コンテンツデータの受信準備 (RTP, RTCP セッションの確立) を行う。
3. 配信サーバでは、クライアントから要求されたコンテンツデータ (MP4 ファイル) を読み込み、RTP パケットに乗せて送信する。
4. クライアントでは、RTP パケットを受信 (バッファリング) し、コンテンツデータの再生 (decode) を行う。

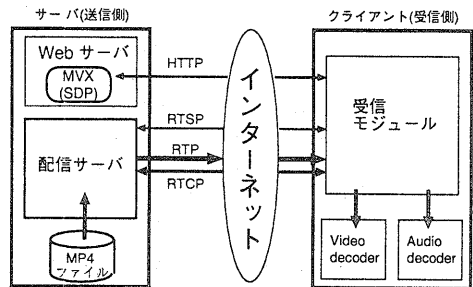


図 1: 動画配信システムの基本構成

4 提案する QoS 制御方式

ネットワーク状態の早期把握と適応した配信によりクライアント上で良い再生品質を実現するため、提案する QoS 制御方式として配信サーバ上に 2 つの機能(図 2)を用意した。1 つは輻輳予測・検出機能であり、クライアントからフィードバックされる RTCP の情報に基づき、ネットワークの輻輳状態の予測・検出を行い、その状態に適応する転送レートの計算を行う。もう 1 つは転送レート制御機能であり、輻輳予測・検出機能で求められた転送レートにしたがってパケットの送出時間(間隔)を計算する。これらは動画・音声それぞれのメディアについて行う。

また、動画再生においては、(MPEG-4 における)差分フレームが欠損したとき、次の I フレームまでデコードを行ってもブロックノイズが発生し画質の悪化となる。そのため、差分フレームが欠損した場合には、クライアント上では次の I フレームが届くまでデコードを行わないスキップ機能も用意した。なお、スキップ機能が実行されている間はフレームが欠損する直前の画面の静止画²となる。

次に、配信サーバ上における QoS 制御の各機能について説明する。

²この現象はフリーズと呼ばれ、その時間をフリーズ時間と呼ぶ。

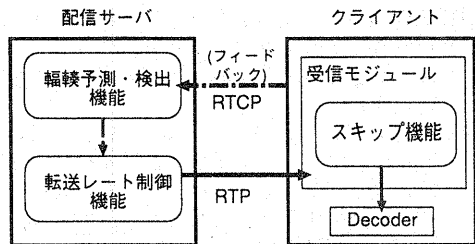


図 2: QoS 制御機能の構成

4.1 輻輳予測・検出機能

クライアントからの RTCP フィードバック情報のパケットロス率とジッタの値を使用してネットワーク状態を判定し輻輳の予測と検出を行い、その結果から理想的な転送レートを計算する。ここで、ジッタの値を J 、パケットロス率を P_r とする。

まず、ジッタとパケットロス率それぞれについて事前に設定した閾値(それぞれ α , β)と比較し、クライアントのフィードバックがあったときのネットワーク状態を分類する。それぞれのフィードバック値と閾値との比較の結果が図3で4つのどの領域(A, B, C, D)に属するのか判定する。ただし、ネットワークの状況によっては RTCP パケット自体が欠損する場合もあるのでその状態を N とし、ネットワーク状態を計5つのいずれかに分類する。なお、分類したそれぞれの状態は、RTP のジッタの計算式の特徴から表1のように説明できる。

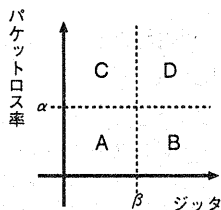


図 3: ネットワーク状態の分類

判定時に使用する閾値については次のように設定する。なお、 r_{media} はメディアの(符号化)ビットレート、 aps は平均パケットサイズとする。

パケットロス率閾値 α : メディアとフレームレートに応じて設定する。

動画: RTCP フィードバック間隔において I フレームの平均サイズ分のデータが欠損する割合 (r_{media} の 10 ~ 20% に相当)

音声: r_{media} の 5% のロス率

表 1: ネットワークの状態

状態	説明
A	J も P_r も小さく、良好な状態。
B	P_r は小さいので輻輳とはまだ断定できないが、 J が大きいので輻輳になる可能性が大きい。
C	P_r は大きい J は小さいため、一時的にネットワークが混雑した、または、輻輳状態から安定してきた。
D	輻輳状態である。
N	前の状態が B,C,D,N であれば、輻輳状態が悪化している。

ジッタ閾値 β : 転送レートに応じ動的に変更する。

$$\beta = aps \cdot \frac{1 - \alpha}{16 \cdot r_{media}}$$

次に、これらの状態を各ストリーム毎に過去数回分記憶しておき、表2のように1つ前の状態から現在の状態への遷移を調べネットワークの状態や輻輳を予測し、転送レートの変更方針を定め、理想的な転送レートを求める。変更方針に基づく転送レートの計算方法は次のようになる。なお、変更前の転送レートを r_{pre} 、求める転送レートを r_{calc} とする。

表 2: 転送レート変更方針

		1つ前の状態				
		A	B	C	D	N
現在の状態	A	↑	—	—	—	—
	B	↓	↓	↓	↓	↓
	C	↓	↓	↓	—	↓
	D	↘	↘	↘	↘	↘
	N	↓	↓	↓	↘	↘

—: 変更なし, ↑: レート上,
↓: レート下(小), ↓: レート下(中),
↘: レート下(大)

[レート下(中, 大)]

まず、転送レートを下げる割合 Δ_R を J , P_r から計算する。なお、 k, m, q は正の定数である。

$$\Delta_R = (1 - k\Delta_{P_r}) \left(1 - \frac{\Delta_J}{q}\right) \quad (1)$$

ただし、

$$\Delta_J = \begin{cases} \frac{J}{k\beta} & 0 \leq J < k\beta \\ 1 & k\beta \leq J \end{cases}$$

$$\Delta P_r = \begin{cases} P_r & 0 \leq P_r < m\alpha \\ m\alpha & m\alpha \leq P_r \end{cases} \quad (2)$$

次に, r_{calc} を求める. $r_{calc} = r_{pre} \cdot \Delta R$
[レート下(小)]

式(1)と同様に ΔR を求め, その大きさをより小さくして転送レートを計算する.

$$r_{calc} = r_{pre} \cdot (1 - (1 - \Delta R)/2)$$

[レート上]

転送レートを増加させる割合を ΔR として次のように計算する. なお, n は正の定数である.

$$\Delta R = \max \left\{ \alpha \left(1 - \frac{J}{n\beta} \right), \alpha - P_r \right\}$$

$$r_{calc} = r_{pre} \cdot (1 + \Delta R)$$

$$\text{ただし, } r_{calc} \leq \frac{r_{media}}{1 - k\alpha}$$

ここで上限を設定しているが, これは r_{media} よりも r_{calc} が大きくなりすぎパケットロスが発生することも考えられるためである. また, もしその状況でパケットロスが発生したとき, r_{calc} を r_{media} より下げないようにする必要がある. そのとき, $\Delta P_r = \alpha$ から (ジッタは無視して) $\Delta R = 1 - k\alpha$ となり $r_{calc} \cdot \Delta R = r_{media}$ で求めることができる.

なお, 音声の品質悪化は動画の品質悪化より知覚的に敏感であるため, 音声の r_{calc} は r_{media} より下げないようにし, 下げる必要がある場合にはその減少分を動画の r_{calc} に反映する.

4.2 転送レート制御機能

輻撃予測・検出機能で求めた転送レートに基づき, 送出するパケットの時間間隔やフレームレートの変更などの制御を行う. これらの制御により, ネットワークの輻撃の悪化の防止や転送効率の向上を可能とする. また, ネットワークの状態が良好な期間に可能な限りデータをクライアントへ転送しバッファに貯えておくことで, 輻撃時でも動画・音声再生の中断を抑える効果を与える.

次に送出時間間隔の計算方法やフレームレートの変更方法について述べる. なお, ある時間間隔内の i 番目のパケットサイズを S_i とし, メディアのフレームレートを f_{media} とする.

[輻撃状態ではないとき ($r_{calc} \geq r_{media}$)]

- 1) 単位時間内に r_{calc} でパケットを送出する. i 番目のパケットの送出時刻 T_i は次となる.

$$T_i = T_{i-1} + \frac{S_{i-1}}{r_{calc}} \quad (3)$$

- 2) 総クライアントバッファ量³ buf と r_{media} からバッファ可能な時間 $T_b = \frac{buf}{r_{media}}$ を求める.

³事前に別の手段で通知されているものとする.

- 3) 配列 Arr_n と Arr_s を用意し, それぞれに各 1 秒内に送出したパケット数と総パケットサイズを記録する. T_b 秒より前の配列の記録は削除する.

- 4) $buf \geq S_i + \sum Arr_s$ または $f_{media} \cdot T_b \geq \sum Arr_n$ のとき, つまり i 番目のパケット送出によりクライアントバッファのオーバーフローが発生すると判断したとき, $\sum Arr_n$ や $\sum Arr_s$ が減少するのを 1 秒単位, または, 再生時間単位で待ち, パケットの送出時間もそれに合わせる.

- 5) RTCP フィードバック情報の最大受信シーケンス番号と送出パケットの最大シーケンス番号を比較し, 差が $f_{media} \cdot T_b$ より大きい場合は同様にパケット送出時間を合わせる.

[輻撃状態であるとき ($r_{calc} < r_{media}$)]

- $\sum Arr_n$ や $\sum Arr_s$ が 1 ~ 2 秒分以上あれば送出時刻を式(3)で求める.
- クライアントバッファ量が少ないと判断したときは, 転送レートが r_{calc} 以下となるように (動画) フレームを間引く. 間引くフレーム数は 1 秒平均 $\left\lceil \frac{r_{media} - r_{calc}}{aps} \right\rceil$ 個である. なお, 音声の r_{calc} は r_{media} より小さくならないため音声パケットの間引きは行われない.
- 動画フレームは I フレームは間引かず必ず送出する. 間引くフレームは次の I フレームの直前の P, B フレームとする. これは, 間引きによる動画のブロックノイズの発生やフリーズ時間の増大を抑止するために行う.

4.3 輻撃時の動画フレームの理想的な間引き数について

説明を容易にするために, I フレームは t 秒に 1 回, フレームのサイズは各々のフレームタイプごとに平均の一定値 (P フレームであれば s_p) であるとする. また, ネットワーク上の利用可能な帯域幅に対し負荷が一定に r_{load} bps ある場合で考える.

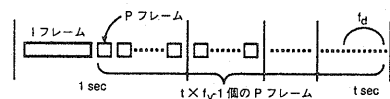


図 4: 動画フレームの間引き

輻撃状態になり音声のパケットは落さず次の I フレームの直前の P フレームだけを必要な数だけ間引くとき, t 秒間で間引くフレーム数 f_d は次で求められる. (図 4)

$$f_d = \left\lceil \frac{t \cdot r_{load}}{s_p} \right\rceil$$

負荷のある期間の動画総フレーム数を F とすると、ネットワーク輻輳時にパケットロスが発生しない理想的な総フレーム数 x は次で求められる。

$$x = F \cdot \frac{t \cdot f_v - f_d}{t \cdot f_v} \quad (4)$$

5 実験と考察

提案した QoS 制御方式の効果を検証するため、インターネットを経由した測定を行い、評価する。

5.1 実験環境

ネットワーク環境は、サーバが OCN 128kbps 常時接続回線、クライアントが ISDN 64kbps ダイアルアップ回線、両者間の hop 数は 15 である。また、コンテンツとして、音楽 (100 秒)、アニメ (80 秒)、ニュース (120 秒)、スポーツ (120 秒) を用意した。他の設定については表 3 である。

表 3: 実験環境

項目	設定
計算機	
サーバ	PentiumPro 200MHz, 192MB WindowsNT 4.0, C
クライアント	Pentium 166MHz, 64MB Windows98, C++
コンテンツ	
動画 codec	MPEG-4 Simple Profile
音声 codec	MP3
Bit Rate	50kbps (動画: 32kbps, 音声: 18kbps)
Frame Rate	動画: 10 fps, 音声: 5.2 fps
I フレーム	約 2 秒に 1 回
その他の設定	
クライアントバッファ	初期バッファ: 4 秒 総サイズ: 10 秒
RTCP フィードバック	4 秒 (固定)

ネットワーク上に負荷を与えた実験を行うため、コンテンツの時間の半分の間だけ負荷を与えた。なお、サーバからクライアントに向けて 16kbps の負荷である。

5.2 実験結果

実験結果を図 5 から図 6 に示す。これらはそれぞれのコンテンツにおいて集計した平均値である。図 5 では棒グラフの下に数値があるが、これは配信サーバからの送出フレーム数に対してクライア

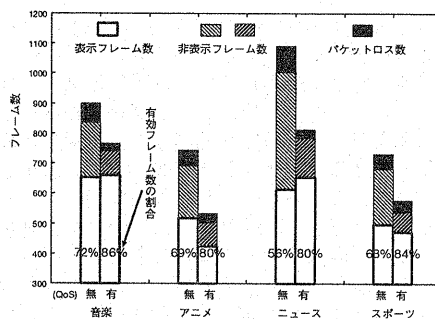


図 5: 動画の受信フレーム数と有効フレーム数

ント上で表示されたフレーム数の割合である。この値が大きいと送出されたパケットが有効に利用されたことになり、配信の効率性の目安となる。

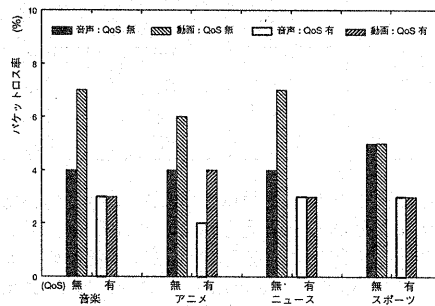


図 6: パケットロス率

5.3 考察

図 5 において、QoS 制御機能がある場合にはネットワークの負荷に応じ動画フレームは間引かれるため、すべての送出フレーム数が減少していることがわかる。しかし、表示フレーム数の割合はすべての場合において大きく増加し、表示フレーム数自体も増加しているものもあるため、無駄なパケットの送出を抑制しネットワークの利用効率を向上させていると考えられる。

効果が最も高いものはニュースであるが、これは動き自体が比較的少なく I フレームの間隔が実際には相対的に長めであり I, P フレームサイズも各々平均的であるためと考えられる。逆に今回動きの大きいコンテンツであるアニメやスポーツでは割合は増加しているが、表示フレーム数自体は若干減少している。動きが大きいコンテンツの場合、フレームサイズの大きいフレーム (特に I フレーム) が存在し、一時的に帯域幅を圧迫しそれらのフレームが欠

損しやすくなるためと考えることができる。

また、図6から、パケットロス率はQoS制御機能がある場合にはすべてにおいて大きく減少しており、制御の効果が得られている。特に動画ではほぼ半減している。そのため、ネットワークの状態に適応し無駄なパケット送出を抑制している効果が表れていると言える。パケットロス率をさらに減少させるためには、より多くのフレームを間引くことが考えられる。ただし、表示フレーム数の割合は増加するがフリーズ時間が大きくなることも考えられるため、エラー訂正技術を活用する、コンテンツ作成時にフレームサイズを可能な限り平滑化することなども必要である。

負荷を与えた時間と実際のコンテンツにおける値、式(4)から求めた各々のコンテンツでの理想的送出パケット数は表4である。間引くフレーム数は動きが少ないコンテンツほど多くなっている。

表4: 理想的送信パケット数との比較

コンテンツ	送出フレーム数	理想的送出フレーム数	全フレーム数
音楽	766	780	900
アニメ	534	632	743
ニュース	813	880	1091
スポーツ	577	636	731

実際の送出フレーム数は、理想的な値との比較から妥当な量であると考えられる。式(4)は、与えた負荷以外の負荷やコンテンツにおける各々のフレームサイズの変動までは対応していないが、送出フレーム数の上界としての意味も十分与えていると考えることができる。

コンテンツ再生時の実際の知覚的な印象として、QoS制御機能が無いときに負荷がある場合、動画はクライアント上での制御であるスキップはあるものの音声の途切れが多く非常に聞き取りが難しかった。これは、ネットワークの状態に適応せず無駄に送出されたパケットによる音声の受信状態の悪化であるとも言える。QoS制御機能がある場合には、クライアント上でスキップは大きく変わらないが音声の途切れは少なく不快感は大きく減少した。そのため、測定した数値だけでなく知覚的な品質も向上していると言える。

6 今後の課題

以上のように、提案したQoS制御によりクライアント上での良い再生品質を実現し効率的な配信を行えることがわかった。しかし、さらに向上させるために今後取り組むべき課題や検討すべき課題は残っている。

パケットロスは減少したが無くなったわけではな

く、表示フレーム数の割合が80%程度であり100%までは少し開きがあるので、その割合を向上させより効率的な配信を可能とする必要がある。

そのためには、一時的な輻輳やコンテンツのフレームサイズの変動にも追従するなど、輻輳予測・検出機能の精度向上を検討する必要がある。また、転送レートの制御やフレームの間引きだけでなく、コンテンツの符号化レートの変更やさまざまなエラー訂正も含め、複合的に対応する必要がある。なお、我々は文献[5]において再送によるエラー訂正の効果については報告している。

7 終わりに

本論文では、インターネット上でRTPを利用した動画配信システムにおいて、ネットワークの状態に応じ受信側で良い再生品質を実現するためのQoS制御方式を提案した。その方式は、ネットワーク状態を把握しその推移から転送レートを見積る輻輳予測・検出機能、その結果を踏まえネットワークの状態に適応したパケット送出を行う転送レート制御機能で構成される。これらの機能を実現し、インターネット上で測定を行った結果を述べ、提案した方式の評価を行った。

結果として、提案した方式はさまざまなコンテンツに対して有効であり、パケット配信の効率性向上やパケットロス率の低下を実現した。さらなる性能向上のため、精度の良い輻輳予測やレート制御技術、エラー訂正技術などについて検討していく必要がある。

参考文献

- [1] H. Shulzrinne, S. Casner, R. Frederick, and V. Jacobson. "RTP: A Transport Protocol for Real-Time Application". RFC1889, 1996.
- [2] H. Shulzrinne, A. Rao, and R. Lanphier. "Real Time Streaming Protocol". RFC2326, 1998.
- [3] ISO/IEC 14496-1. "ISO/IEC 14496-1 Information Technology - Coding of audio-visual objects - part1: Systems", 1999.
- [4] Civanlar, Balabanian, Basso, Casner, Herpel, and Perkins. "RTP Payload Format for MPEG-4 Streams". Internet Draft(draft-ietf-avt-avt-rtp-mpeg4*), 1999.
- [5] 奥村誠司, 福田和真, 鷹取功人, 大野次彦, 下間芳樹. "MPEG-4 over RTP 配信システムとQoS制御方式". 情報処理学会 マルチメディア・分散・協調とモバイル(DICOMO2000)シンポジウム, Vol. 2000, No. 7, pp. 433-438, 2000.