

TCP との公平性を考慮した動画像転送

若宮 直紀 村田 正幸 宮原 秀夫

大阪大学 大学院基礎工学研究科 情報数理系専攻

〒 560-8531 大阪府豊中市待兼山町 1-3

Tel: 06-6850-6586 Fax: 06-6850-6589

E-mail: wakamiya@ics.es.osaka-u.ac.jp

あらまし 通信品質保証のないインターネットにおいては、高品質な通信を求めるデータ系アプリケーションは TCP を、高速な通信を求める実時間型アプリケーションは UDP をそれぞれ利用するため、ネットワークには TCP、UDP の異なるプロトコルが存在している。そのようなネットワークにおいては、動画像通信に代表されるマルチメディアアプリケーションの生成する大量のトラフィックが UDP によりなんの制御も行なわれなまま転送されるため、輻輳制御を行なう TCP の使用帯域を圧迫し極端な品質劣化を招く。

そこで、本稿では TCP と公平な動画像転送を実現するため、TCP-friendly の概念を導入したレート制御を対象に、動画像転送への適用可能性や効果的な制御手法について検討した。動画像の品質を保ちつつ、TCP と公平な通信を実現するレート制御を行なうためには、(1) 制御間隔を適切に設定し、(2) ネットワークの状態を推測し、(3) TCP のスループットを推定して (4) 動画像データの生成レートを調整しなければならない。シミュレーションによる評価により、動画像を RTT の約 32 倍の制御間隔で TCP の推定レートにあわせて MPEG-2 インTRASライス符号化して送出すれば、TCP との公平性を実現しつつ、高品質な動画像転送が可能であることを明らかにした。

キーワード TCP-friendly, 動画像転送, QoS, レート制御

On TCP-friendly Video Transfer

Naoki WAKAMIYA Masayuki MURATA Hideo MIYAHARA

Department of Informatics and Mathematical Science,
Graduate School of Engineering Science, Osaka University
1-3 Machikaneyama, Toyonaka, Osaka 560-8531, Japan
Tel: 06-6850-6586 Fax: 06-6850-6589
E-mail: wakamiya@ics.es.osaka-u.ac.jp

Abstract

In the current Internet, both TCP and UDP sessions co-exist. Owing to the rapidly growing computer technology, the distributed multimedia applications are widely used which inject considerable amount of traffic with “greedy” UDP. Under such condition, the performance of TCP sessions easily deteriorate because of congestion incurred by UDP sessions.

In this paper, we introduce the TCP-friendly rate control protocol which enables non-TCP connections to fairly share the link with TCP connections. To achieve fairness among TCP and UDP sessions while performing high quality video transmission, (1) the interval of rate control must be appropriately determined, (2) the network condition must be accurately predicted, (3) the TCP throughput must be precisely estimated and (4) the video rate must be effectively adjusted. Through simulation experiments, we show that the MPEG-2 intra-slice coding algorithm is appropriate and the control interval should be 32 times as long as RTT.

key words TCP-friendly, video transfer, QoS, rate control

1 はじめに

通信品質に関する保証のないインターネットでは、これまでエンドシステムがアプリケーションごとに異なる通信プロトコルを使い分けることにより必要とする通信品質を得る努力がなされてきた。データの棄却に厳しい一方で遅延には比較的寛容なデータ型アプリケーション (http, ftp, telnet など) に対しては、ネットワークの状態に応じた輻輳制御を行なうことによりデータ損失の少ない高品質な通信を実現することのできる TCP が用いられている。一方、多少のデータ損失に耐えることのできる実時間型アプリケーション (TV 会議など) では、高速なデータ通信を行なうために、再送制御やレート制御を行わない UDP が使用されている。今後ますます利用が拡大すると思われるマルチメディア通信は従来のデータ通信と比較してその送出トラフィック量が極端に多いため、過度の UDP トラフィックによる帯域占有や、パケット棄却率の増大などの問題を引き起こすことが予想される。その結果、信頼性の高い通信を目指した TCP コネクションの極端な性能劣化を招くだけでなく、中継ノードに RED など適切なパケットスケジューリング機構がない場合には、同じ経路を通る UDP セッション間でも通信品質劣化、性能格差が生じると考えられる。

データ通信における TCP との公平性を表す概念として TCP-friendliness を導入した研究が近年活発に行なわれている [1-6]。TCP-friendliness は「non-TCP コネクションの得るスループットが同一パス上の TCP コネクションと同じであること」と定義され、ネットワークの状態に応じて non-TCP コネクションのデータ送出レートを調節することにより TCP と公平な通信を実現する。いずれの手法においても適切なパラメータ設定を行えばある程度公平性を達成することができるが、そのようなパラメータ設定が必ずしも動画通信に代表されるマルチメディアアプリケーションにとって適当であるとは限らない。例えば、公平性を高めるためには頻繁なレート制御を行なうことが有効であるが、受信側エンドシステムで再生される動画は時間による品質のばらつきが大きなものとなってしまう。逆にレート制御頻度が小さすぎる場合には、ネットワークの状態にそぐわないレートでの動画転送を長期間行なうことになり、不当に低い品質の動画が配送されたり、ネットワークに負荷を与え続ける可能性がある。

そこで、本稿では、TCP の通信品質を劣化させることなく高速かつ高品質な動画通信を行なうと同時に、ネットワーク全体の通信効率を高める TCP との公平性を考慮した動画レート制御手法について検討を行なう。なお、以降では [4] にない、TCP との公平な通信を実現する TCP-friendly な通信プロトコルを TFRCP (Tcp-Friendly Rate Control Protocol) と表記する。

2 動画転送に適した TFRCP

TFRCP では、(1) 適切な制御間隔ごとに、(2) 観測結果やフィードバック情報に基づいてネットワークの状態を推測し、(3) TCP のスループットを推定する。

2.1 TFRCP におけるレート推定

「non-TCP コネクションの得るスループットが同一パス上の TCP コネクションと同じであること」[4] という TCP-friendliness を目指すレート設定手法には大別して、ネットワークの状態から TCP コネクションのレートを推定するものと、似非 TCP なレート制御を行なうものがあるが、ここではネットワークの状態、すなわちパケット棄却率や RTT から TCP コネクションのスループットを推定する手法について検討する。なお、UDP は厳密にはコネクションレス型の通信プロトコルであり、コネクションという概念は存在しないが、公平性の対象となる TCP コネクションと同一経路を通る UDP による通信を便宜上 UDP コネクションと呼ぶ。

手法 1 [1, 2]: 本手法では、RTT やパケット棄却率によって表されるネットワークの状態から、次式を用いて TCP コネクションの得るスループット λ_{equ} を推定する。

$$\lambda_{equ} = C \times \frac{MTU}{RTT \sqrt{Loss}} \quad (1)$$

ただし、係数 C には受信側エンドシステムが Delayed ACK を使用している場合には 0.87、各受信パケットごとに ACK を返送する場合には 1.22 を与える。

手法 2 [3, 4]: 手法 2 は、手法 1 では考慮されていなかった受信側エンドシステムのウィンドウサイズ W_{max} やタイムアウト発生時のスループットの変化も含めた定式化を行なうことにより推定精度を高めている。

$$B(p) \approx \min\left(\frac{W_{max}}{RTT}, \frac{1}{RTT \sqrt{\frac{2bp}{3} + T_0 \min(1, 3 \sqrt{\frac{2bp}{3}}) p(1+32p^2)}}\right) \quad (2)$$

ただし、 b には Delayed ACK の使用、不使用に応じて 2 あるいは 1 を与える。また、 p はパケット棄却の頻度、 T_0 はバックオフの基本時間 (RTO) である。

これらの手法の推定精度を図 1 に示すネットワークモデルにおけるシミュレーションにより評価した。シミュレーションには ns version 2 [7] を用いた。ネットワークはタンデムに接続されたリンクから構成されており、各ノードにはドロップテイルのバッファが設けられている。それぞれ N , M 本の TCP (TCP-Reno)、UDP コネクションは全てのリンクを経由して受信側エンドシステムに対してデータを送出する。また、外部からの流入トラフィックとして、それぞれ 4 対の TCP, UDP コネクションを設定した。また、TCP コネクションの受信側エンドシステムは Delayed ACK を用いず、受信バッファサイズは、送信側エンドシステムのレート制御に影響を与えない程度に大きく設定した。なお、TCP のタイマ

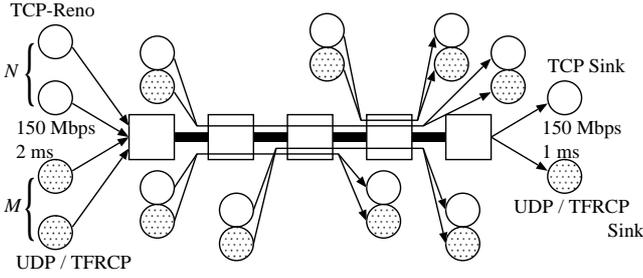


図 1: ネットワークモデル

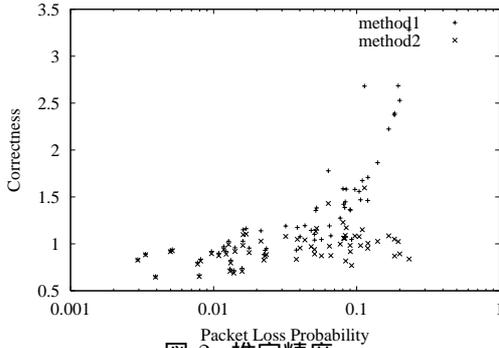


図 2: 推定精度

粒度は 10 msec とする。

中継ノードのバッファサイズを 5~30 パケット, リンク容量を 1~100 Mbps, 各リンクの伝搬遅延を 10 msec とし, TCP, UDP のコネクション数を $N = 1, M = 0$ とした場合のパケット棄却率と推定精度の関係を図 2 に示す。ただし, 推定精度 CR は次式で与えられる。

$$CR = \frac{\text{Estimated Rate}}{\text{Average Throughput}} \quad (3)$$

なお, 推定式に与える RTT, RTO として TCP-Reno による RTT 推測値 ($srtt$) および RTO のシミュレーションを通した平均値, MTU として TCP のパケットサイズ 1000 Byte, $Loss, p$ にはシミュレーション中の総送出パケット数に対する棄却パケット数の割合を与えた。なお, 本稿では受信側エンドシステムのバッファサイズを非常に大きく設定しているため, 式 (2) はパケットサイズを考慮して以下のように書き改められる。

$$B(p) \approx \frac{MTU}{RTT \sqrt{\frac{2p}{3}} + T_0 \min(1, 3\sqrt{\frac{3p}{8}}) p (1 + 32p^2)} \quad (4)$$

図に示される通り, パケット棄却率の大きい領域では, タイムアウトを考慮した手法 2 の推定精度の方が手法 1 よりも高い。また, 全体的に推定レートが実レートを下回る傾向があるが, これは手法 1, 2 のモデルがランダムなパケット棄却が発生する環境を前提としているのに対し, ドロップテイルを用いた本シミュレーションではバースト的な棄却が発生することなどによる。

2.2 RTT , パケット棄却率の推定法

TFRC が RTT やパケット棄却率を推定するためには, 適当な頻度でネットワークや受信側エンドシステムからのフィードバックを得る必要がある。例えば, [1-4, 6] ではパケットにタイムスタンプを付加し ACK の受信時刻

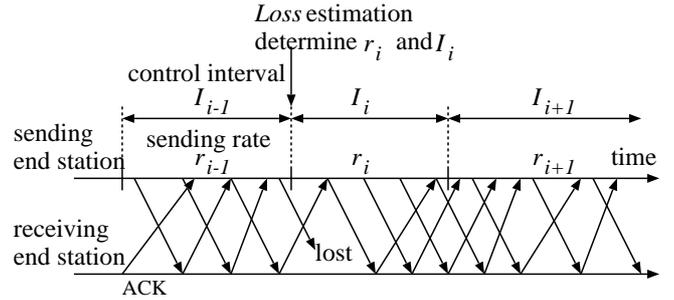


図 3: TFRC による動画転送

を観測することで, [5] では RTCP パケットを送受信することにより RTT の観測値を得, 複数の観測値に対して適当なフィルタを適用することで RTT を推定する。

パケット棄却率に関しては, パケットのシーケンス番号の抜けから棄却数を算出し, 総送出パケット数で除算する手法 [5, 6] や, TCP と同様に 1 RTT 内のパケット棄却は 1 回と数え, 観測期間のパケット棄却回数を総パケット数で除算する手法 [2-4] などがとられる。

前節であげたレート推定手法の精度は, ネットワークの状態推定の影響を大きく受ける。高い制御効果, すなわち公平性を得るためにはできる限り正確に RTT やパケット棄却率を推定する必要があるが, いずれにおいても特に定められた手法があるわけではない。また, その精度についても十分な議論がなされていないが, いずれのレート推定手法も TCP の輻輳制御をモデルとしていることから, TCP で用いられる RTT 算出アルゴリズムやパケット棄却の検出メカニズムと同程度の精度があれば, 制御効果が得られると考えられる。

2.3 TFRC におけるレート設定手法

本節では, 図 3 に示すように時間 I_i の制御期間 i で測定されたパケット棄却率や RTT に基づき次の期間 $i+1$ の制御間隔 I_{i+1} やデータ送出レート r_{i+1} を決定する TFRC アルゴリズムを提案する。

TFRC におけるレート推定手法としては 2.1 節での評価において推定精度の高かった手法 2 を用いる。ただし, 手法 2 では観測期間中パケット棄却が発生しなかった場合に次の送出レートを決定できないため, [4] にならない, 期間 $i+1$ のデータ送出レート r_{i+1} は, 期間 i におけるパケット棄却の有無に応じて次のように与えられる。

$$\text{棄却あり} \quad r_{i+1} = \frac{MTU}{RTT \sqrt{\frac{2p}{3}} + T_0 \min(1, 3\sqrt{\frac{3p}{8}}) p (1 + 32p^2)} \quad (5)$$

$$\text{棄却なし} \quad r_{i+1} = 2 \times r_i \quad (6)$$

ただし, 初期レートは特に定めず, 符号化された動画データがレート制御の適用なく送出されるものとする。

RTT の観測値はパケットヘッダにタイムスタンプを付加することにより得る。ただし, 受信側エンドシステムは全ての受信パケットに対して, そのシーケンス番号が書き込まれた ACK を返送するものとする。従って, 送信側エンドシステムは受信した ACK に対して簡単な算術演算 (ACK 受信時刻 - T_{Secr}) を施すことにより RTT

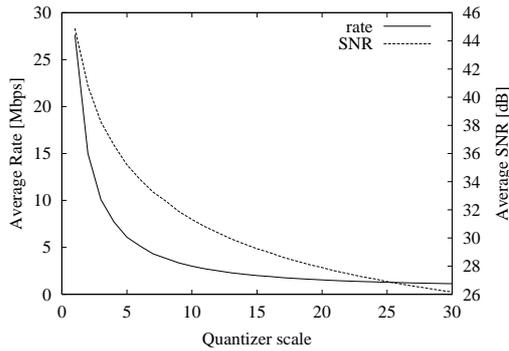


図 4: 量子化スケールとレート、動画像品質の関係

の観測値 rtt を得ることができる。得られた観測値 rtt から推定値 RTT および T_0 (RTO) を算出する際には、広く一般に用いられている Jacobson のアルゴリズム [8] を用いる。期間 i のパケット棄却率 p は、送信側エンドシステムにおいて次式によって算出される。

$$p = \frac{\text{総パケット数} - \text{受信パケット数}}{\text{総パケット数}} \quad (7)$$

ただし、総パケット数は、期間 $i-1$ の最終受信パケット番号 $+1$ - 期間 i の最終受信パケット番号で与えられる。

送信側エンドシステムは、設定されたデータ送出レート r_i に応じて動画像トラヒックの生成レートを調整しなければならない。本稿では動画像の符号化方式として広く用いられている MPEG-2 [9] を適用する。MPEG-2 では、入力画像に対して離散コサイン変換 (DCT: Discrete Cosine Transform) を施し、得られた DCT 係数を量子化することにより高い圧縮率を得る。したがって、生成される動画像トラヒックのレートや得られる動画像品質は量子化の度合、すなわち量子化スケールの値によって調整することが可能である。図 4 に、評価に用いた動画像における量子化スケールと生成される動画像の平均レートおよび SNR (Signal to Noise Ratio) で評価した平均動画像品質の関係を示す。ただし、MPEG-2 符号化アルゴリズムとしては、動画像シーケンスを通じて同じ量子化スケールを使用する VBR (Variable Bit Rate) 符号化手法を用いている。文献 [10] では、さまざまな動画像データに対して量子化スケール、解像度、フレームレートといった符号化パラメータから動画像転送に必要な帯域が得られる関係式を導出しているが、図 4 に対しても同様の手法を適用することにより量子化スケールと平均レートの関係を得ることができる。

本稿で用いる TFRCP アルゴリズムでは、図 4 から得られる関係に基づいて、式 (5) および式 (6) によって定められた送出レート r_{i+1} の範囲内で最も高い動画像品質を得ることのできる量子化スケールを選択する。期間 $i+1$ 中は決定された量子化スケールを用いて動画像符号化を行ない、生成された動画像データは送信側エンドシステムから順に送出される。ただし、期間内の総送出データ量が設定レート r_{i+1} と制御期間 I_{i+1} の積によっ

て与えられる期間内送出可能データ量を超過する場合には、残りの動画像データは送出されない。

2.4 動画像転送に適した制御頻度

TFRCP の制御間隔を決定する際には、レート制御による動画像転送アプリケーションの品質変動や、レート制御が TCP コネクションの通信品質に与える影響を考慮する必要がある。TCP との公平性を高めるためには 1 RTT に一度程度まで制御を行なう必要があるが [6]、頻繁なレート制御はネットワークの不安定性を引き起こすと同時に過度な画質変動を招くことが予想される。また、制御間隔が小さすぎると精度の高い状態推定が行えない。一方、レート制御の間隔が大きすぎる場合にはネットワークの状態にそぐわないレートでのデータ送出が長期間続くことになり、TCP コネクションにとって不利になると考えられる。

制御頻度と得られる効果について評価するため、図 1 のネットワークを対象に、リンク容量を 150 Mbps、中継ノード内バッファサイズを 100 パケットとし、TCP、TFRCP のコネクション数をそれぞれ 0~30 の範囲で変化させて 60 sec のシミュレーションを行なった。外部からの TCP、UDP コネクションには、図 4 において量子化スケールを 6 とした動画像を入力トラヒックとして与える。ただし、コネクションごとにランダムに決定されたフレームよりデータ送出を開始する。一方、 N 本の TCP コネクションには、ピークレートが約 28 Mbps の CBR トラヒックを入力として与える。これは、TFRCP への入力動画像において最も画質の高い場合の平均レートと等しい (図 4)。全てのコネクションのデータ送出開始時刻は 0~1/29.97 sec の間でランダムに決定される。

同じ条件で TFRCP のかわりに UDP を用いた場合の TCP、UDP の公平性を比較した結果を図 5 に示す。公平性の指標 FR は次式により与える。

$$FR = \frac{\text{average throughput of UDP connections}}{\text{average throughput of TCP connections}} \quad (8)$$

ただし、TCP におけるスループットは受信側エンドシステムへの到達パケット数から算出されており、グッドプットとは異なる。図 5 に示す通り、UDP のコネクション数が多くなるに従い公平性は劣化し、本シミュレーションでは最大約 36 倍の差が生じている。

まず、公平性を高めるために RTT (およそ 106 msec) ごと [6] にレート制御を行なった結果を図 6 に示す。UDP の場合 (図 5) と比較して公平性は高まっているが、最大約 10 倍の差がある。TCP、TFRCP のコネクション数がいずれも 10 本の場合の動画像品質、TFRCP の設定レートの変動の様子を図 7 に示す。ただし、見やすさのためシミュレーション時間 10 sec までを拡大した。また、図には受信側エンドシステムで再生される動画像の品質ではなく、送信側エンドシステムで生成し、送り出された動画像の品質を示している。図 7 より、レート制御頻

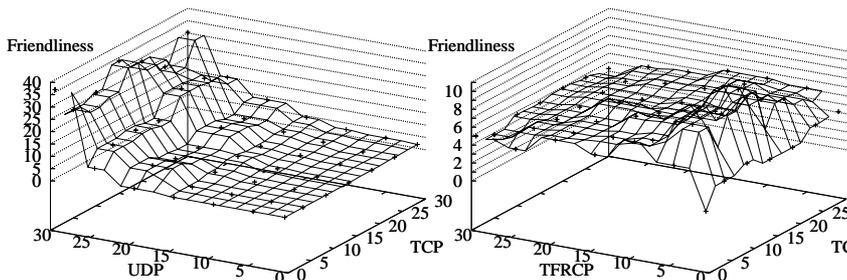


図 5: Friendliness (UDP)

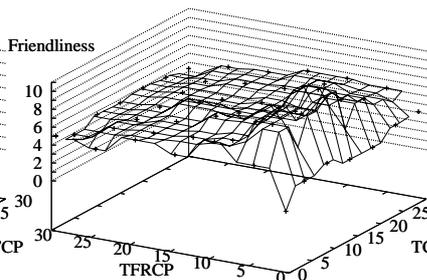


図 6: Friendliness ($I = RTT$)

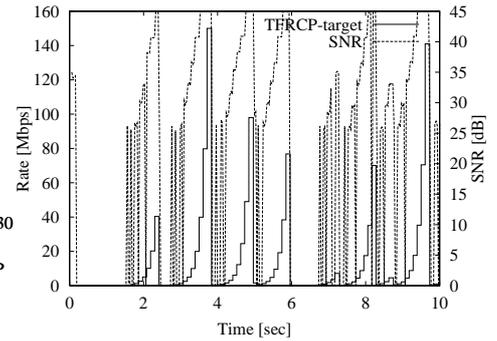


図 7: 設定レート, 動画品質の変動 ($I = RTT$)

度が高すぎるため画質の変動が大きく、動画が全く送られていない期間も存在することがわかる。

公平性を高め、画質変動の度合いを抑えるためには制御期間を長くすることが有効であると考えられる。また、MPEG ではそれぞれ符号化アルゴリズムの異なる I, P, B ピクチャを組み合わせることにより圧縮効率を高めるといふ、GoP (Group of Pictures) を単位とした動画符号化を行なっているが、制御期間の後半にサイズの大きい I ピクチャの送出タイミングが該当した場合には、総送出データ量が送出可能データ量を超過する可能性が高くなり、残りの動画データが送れなくなるという問題が発生する。受信側エンドシステムで観測した動画品質は、ネットワーク内でのパケット棄却などの影響を大きく受ける。特に、送出データ量超過により I ピクチャが送出されなかった場合には、GoP 内の残りの全てのピクチャの復号化に大きな影響を与えるため、実際に受信側エンドシステムで再生される動画の品質は図 7 に示したものよりも大きく劣化することが予想される。

動画符号化装置は制御期間ごとに動画の生成レートを変更しなければならないため、高品質な動画転送を実現するためには、制御間隔は GoP 時間の倍数であるのが望ましい。そこで、制御間隔として、RTT の 32 倍 [2] (およそ $106 \text{ msec} \times 32 = 3392 \text{ msec}$) を GoP 時間 ($30/29.97 \text{ sec}$) の倍数で近似したものをを用いた場合のシミュレーション結果を図 8 および図 9(a) に示す。図より、制御間隔を大きく設定することにより、最大 1.4 倍と高い公平性が得られていることがわかる。ただし、TCP のコネクション数が増加すると TFRC のスループットが TCP を下回るが、これは 2.1 節で示した通り、TFRC アルゴリズムで用いた推定式による推定スループットは共存する TCP コネクションの実スループットよりも低い傾向があることによる。また、TCP コネクションはネットワークの輻輳状態に応じて、与えられた CBR トラヒックのピークレートの範囲内で自由なレート設定が可能であるのに対し、量子化スケールによって調整される TFRC コネクションのデータ送出レートは離散値であるため、柔軟なレート設定が行えないことも一因と考えられる。図 9(a) より、送出失敗のない高品質な動画転送が行なえていることがわかるが、パケット棄却発

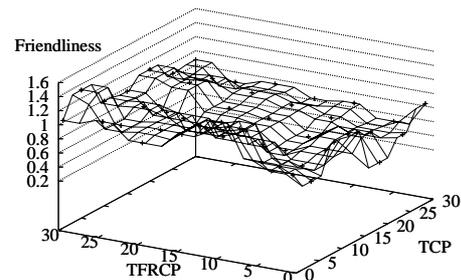


図 8: Friendliness ($I = 32 \times RTT$)

生によるレート変更の結果、およそ 12~15 sec ごとに急激な動画品質劣化が発生している。

3 動画トラヒックのレート制御

本章では、TFRC アルゴリズムにより決定された動画トラヒック送出レートにしたがって動画データの生成レートを調整するための制御手法について検討する。動画の送信側エンドシステムでのバッファリング (スムージング) を前提にすれば動画データの生成レートと送出レートは必ずしも同じである必要はないが、実時間性を損なうため本稿ではバッファを積極的に活用する手法はとらない。したがって、動画符号化装置は TFRC によって設定された送出レートに基づいて、動画符号化パラメータを変更するなどして動画データの生成レートを調節する。ただし、本稿では動画の符号化パラメータの変更は瞬時に生成データに反映されるものとする。動画の生成レート制御は符号化手法により異なるが、ここでは MPEG-2 のさまざまな符号化アルゴリズム [11] で適用可能な手法について検討する。

固定の量子化スケールを用いて符号化することでシーケンスを通じて動画品質が比較的安定する VBR (Variable Bit Rate) 符号化手法に対しては、2.4 節で評価に用いた手法と同様に、TCP の推定レートから適切な量子化スケールを算出し、制御期間内の全てのピクチャを同じ割合で量子化することにより、推定レートに応じた動画符号化が可能である。ただし、算出された量子化スケールを用いて動画を符号化しても、動画やシーンの内容によっては動画トラヒックの平均レートが必ずしも算出式と一致するとは限らないため、短期的な輻輳を引き起こす可能性がある。

一方、設定レートに応じてマクロブロック単位で量子

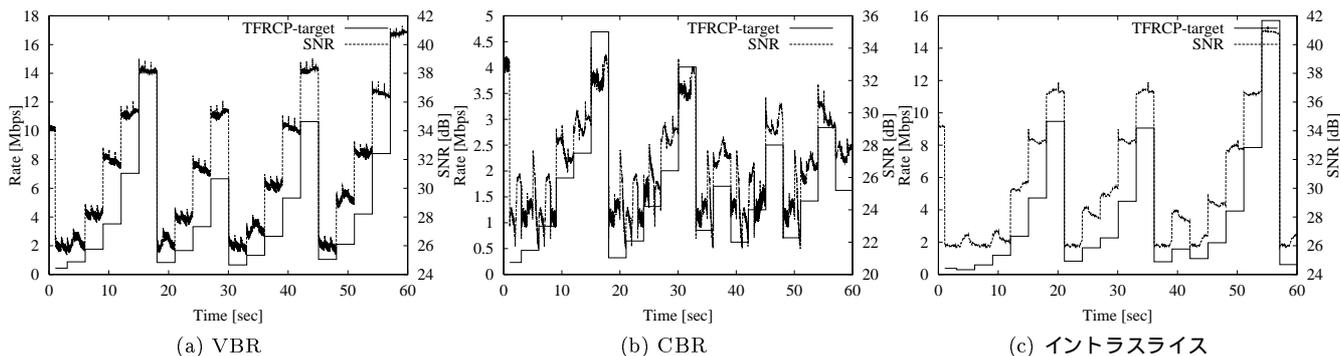


図 9: 設定レート, 動画品質の変動

化スケールを調整することによりレート制御を実現する CBR (Constant Bit Rate) 符号化手法においては, 得られた推定レートを符号化の目標レートに設定することにより, 容易に推定レートに応じた動画データ生成が行なえる [12]. ただし, 図 9(b) に示す適用結果から明らかとなり, マクロブロック単位の量子化スケール調整の結果, 動画品質の変動が大きくなる.

イントラスライス符号化手法では, VBR, CBR 符号化手法と異なりピクチャタイプに依存した符号化は行なわない. そのため, VBR, CBR 符号化手法と比較してレートや動画品質の変動が小さい (図 9(c)). ただし, VBR 符号化手法と同様に量子化スケールと平均レートの関係式を用い, TCP の推定レートに応じた動画データ生成を行なった. あるいは, CBR 符号化手法のように各ピクチャごとの割当ビット数と仮想バッファに基づいて量子化スケールを調整するなどの手法も有効である. いずれの場合にも, GoP 構成に依存しないため, 制御期間を柔軟に設定することができる.

4 おわりに

本稿では, ベストエフォート型のインターネットにおいて, TCP と公平な動画転送を行なうためのレート制御について検討した. シミュレーションによる評価の結果, 動画符号化手法として MPEG-2 イントラスライス符号化手法を用い, RTT のおよそ 32 倍の制御間隔でネットワークの状態推測に基づいてデータ送出レートを設定するアルゴリズムが有効であることが明らかにした.

ただし, 本稿で用いたレート設定手法ではパケット棄却発生時の動画品質劣化の度合いが大きいため, 動画データ送出レートのより安定する手法について検討しなければならない. また, パケット棄却が発生した場合の画質劣化を防ぐためには, 時間的, 空間的なエラー回復制御や階層符号化を利用したエラー制御などをあわせて用いる必要がある.

参考文献

- [1] M. Mathis, J. Semke, J. Mahdavi, and T. Ott, "The macroscopic behavior of the TCP congestion avoidance algorithm," *Computer Communication Review*, vol. 27, pp. 67–82, July 1997.
- [2] W. tian Tan and A. Zakhor, "Real-time internet video using error resilient scalable compression and TCP-friendly transport protocol," *IEEE Transactions on Multimedia*, vol. 1, pp. 172–186, June 1999.
- [3] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A simple model and its empirical validation," *Proceedings of ACM SIGCOMM'98*, September 1998.
- [4] J. Padhye, J. Kurose, D. Towsley, and R. Koodli, "A model based TCP-friendly rate control protocol," *UMASS-CMPSCI Technical Report*, October 1998.
- [5] J.-C. Bolot and T. Turletti, "Experience with control mechanisms for packet video in the Internet," *ACM SIGCOMM Computer Communication Review*, vol. 28, pp. 4–15, January 1998.
- [6] R. Rejaie, M. Handley, and D. Estrin, "RAP: An end-to-end rate-based congestion control mechanism for realtime streams in the Internet," *Proceedings of IEEE INFOCOM'99*, March 1999.
- [7] The VINT Project, "UCB/LBNL/VINT network simulator - ns (version 2)," <http://www-mash.cs.berkeley.edu/ns/>, 1996.
- [8] W. Stevens, *TCP/IP Illustrated*. Addison-Wesley, 1994.
- [9] ISO/IEC DIS 13818-2, "MPEG-2 video," *ISO standard*, 1994.
- [10] 福田健太郎, 若宮直紀, 村田正幸, 宮原秀夫, "帯域予約型ネットワークに適した MPEG-2 動画における品質と帯域の関係," *電子情報通信学会論文誌 B*, vol. J82-B, pp. 358–367, March 1999.
- [11] 若宮 直紀, 長岡 利幸, 村田 正幸, 宮原 秀夫, 大山 公一, "アプリケーションレベルの要求品質を考慮した動画符号化転送方式," *電子情報通信学会技術研究報告 (IN98-202)*, pp. 111–116, March 1999.
- [12] ISO/IEC/JTC1/SC29/WG11, "Test model 5," April 1994.