

QoSを保証しないネットワークのための 輻輳制御機能を有する連続メディア情報転送プロトコル

加藤 聰彦† 木村 昭‡ 鈴木 健二†

† 国際電信電話(株) 研究所
〒356 埼玉県上福岡市大原 2-1-15

‡ 電気通信大学(現在 国際電信電話(株))
〒182 東京都調布市調布ヶ丘 1-5-1

近年、Ethernetなどの共有メディア型LANやインターネットなどのQoSを保証しないネットワークを介して、映像や音声等の連続メディア情報を転送する要求が高まっている。しかし、このようなネットワークでは、通信を要求するトラヒックがネットワークの帯域を越えると、ネットワークの輻輳により遅延やパケット損失が発生する。本稿では、ネットワークの輻輳状況に応じて、符号化レートを変えながら連続メディア情報を転送する通信プロトコルを提案するとともに、シミュレーションにより、その性能評価を行った結果を示す。本プロトコルは、(1) 往復伝送遅延(RTT)の増加によりネットワークの輻輳を検知する、(2) 輻輳が検知された場合は、一旦データパケットの送出速度を減少させ、その後、輻輳が検知されなければ、パケット送出速度を徐々に増加させる、(3) データパケットの送出速度と、符号化速度を独立に扱い、データパケット送出速度が頻繁に変動しても、符号化速度は緩やかに変化させることを可能とするという機能を有する。

A Continuous Media Transfer Protocol with Congestion Control Mechanism for Network without QoS Guarantee

Toshihiko Kato† Akira Kimura‡ Kenji Suzuki†

† KDD R & D Laboratories
2-1-15 Ohara, Kamifukuoka-shi, Saitama 356, JAPAN

‡ University of Electoro-Communications
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182, JAPAN

Recently, it is required to transfer continuous media over networks with out QoS guarantee. In these network, congestion will cause delay and packet loss which will degrade the quality of continuous media transfer. This paper proposes a new protocol which provides congestion control adjusting encoding rate to the network situation, and describes the results of performance evaluation using simulation. This protocol includes the features such that (1) it detect congestion through increase of RTT, (2) if it detects congestion, it decreases data packet transfer rate and, after that, if it does not detect any congestion, it increases the rate gradually, and (3) it handles the data packet transfer rate and the encoding rate independently.

1 はじめに

近年、コンピュータネットワークを介した、映像や音声等の連続メディア情報の転送に対する要求が高まっている。連続メディア情報の転送においては、ファイル転送などの通常のデータ転送と異なり、再生する時刻までにメディア情報を到着させることが必要となる。これを完全に満足するためには、ネットワークが、個別の連続メディアトラフィックに対して、必要な帯域を割り当て、遅延、遅延変動、パケット損失率等のサービス品質 (QoS) を保証することが要求される。

しかし、広く使用されている Ethernet などの共有メディア型 LAN や、インターネットでは、利用できる帯域をすべてのトラフィックが奪い合うため、QoS の保証は一般には困難である。すなわち、通信を要求するトラフィックがネットワークの帯域を越えると、送信側端末やルータにおいて遅延やパケット損失が発生する。この状況はネットワークの輻輳と呼ばれている。このようなネットワークを介して、連続メディア情報を転送する際には、次のような輻輳制御を行う必要がある。

- 連続メディア情報の符号化に対して、複数化レートをサポートする。
- ネットワークの輻輳による QoS の劣化を検出して、連続メディア情報の転送が、その輻輳に影響されないように、符号化レートを減少させる。
- ネットワークの輻輳が終了し、QoS が向上したことを、検出して、その状況において転送可能なレートまで、符号化レートを増加させる。

これまでにも、連続メディア情報の輻輳制御手順の提案は行われている [1-3]。これらの方法では、輻輳の制御と連続メディア情報の符号化レートの変更を直接対応させている。例えば、文献 [1] の方法では、連続メディア情報のパケットの往復遅延時間の増加により輻輳を検出すると、連続メディア情報の符号化レートを減少させ、その後、一定時間輻輳が検出されないと、輻輳が終了したと判断し、符号化レートを増加させる。しかし、このような方式では、以下のような問題が考えられる。

- 符号化レートの変更を頻繁に行うことができないため、輻輳への応答が遅れる場合がある。
- 連続メディア情報を多重化した場合には、互いに輻輳制御手順が起動され、符号化レートの減少と増加を繰り返し、一定の符号化レートを保つことができない。

そこで筆者らは、データ転送制御における輻輳の発生と終了の検出と、連続メディア情報の符号化レートの変更という 2 つのレベルに対して、独立なレート制御を導入することにより、前述の問題点を解決する通信プロトコルの検討を行っている [4]。本プロトコルでは、データ転送制御におけるレート制御は、TCP における輻輳制御手順で

あるスロースタートと輻輳回避アルゴリズム [5] の手順を参考にして決定され、輻輳の発生と終了を素早く検出し、一方、符号化レートの変更は、連続メディア情報の品質を劣化させない時間間隔で行うことを可能としている。本稿では、筆者らの提案するプロトコルの概要と、シミュレーションによる性能評価の結果について示す。

2 TCP の輻輳制御手順

TCP 通信においては、フロー制御がエンドツーエンドで行われるため、ネットワークに伝送速度が小さい通信路が存在する場合や、特定のネットワークにトラフィックが集中した場合などには、ルータなどのネットワーク内の通信機器においてバッファリングによる遅延や、バッファオーバフローによるパケット損失が発生する。そこで、TCP ではパケット損失を契機に、スロースタートと輻輳回避アルゴリズムを起動する。その手順は以下の通りである [5]。

- 送信側が内部的に使用するウィンドウサイズ (congestion window: *cwnd*) と、スロースタートのための閾値 (slow start threshold: *ssthresh* という内部変数を用いる。
- 送信側は *cwnd* と受信側から通知されたウィンドウサイズの小さい値に従ってデータを送信する。
- タイムアウト再送または重複した応答パケットによる再送を契機に、輻輳が検出されると現在のウィンドウサイズの $1/2$ を *ssthresh* に代入する。タイムアウト再送の場合は、*cwnd* の値をパケットサイズ分にする。
- 受信側から新たなデータの応答を受信すると、*cwnd* を次のようにして増加させる。*cwnd* が *ssthresh* 以下である場合は、TCP をスロースタートを行い、*cwnd* は応答を受信するごとにパケットサイズ分増加させる。これはウィンドウを、往復遅延時間 (RTT: Round Trip Time) ごとに 2 倍に、指数関数的に増加させることになる。一方 *cwnd* が *ssthresh* より大きい場合は、輻輳回避を行い、RTT ごとにパケットサイズ分ずつ線形に増加させる。

このような手順により、ウィンドウサイズを一旦減少させた後、*ssthresh* までは急速に、その後は緩やかに増加させ、スループットの低下と輻輳の再発を抑えることができる。

3 連続メディア情報転送プロトコルの基本方針

(1) 本プロトコルは、図 1 に示すように、連続メディア情報の符号化制御部とデータ転送制御部の 2 つのレベルにより実現される。符号化制御部は、データ転送制御部から

指定された符号化レートで連続メディア情報の符号化、復号を行う。データ転送制御部は、データ転送、ネットワークのQoSの変動の検出、変動したQoSに対応する符号化速度の符号化部への通知の機能を提供する。

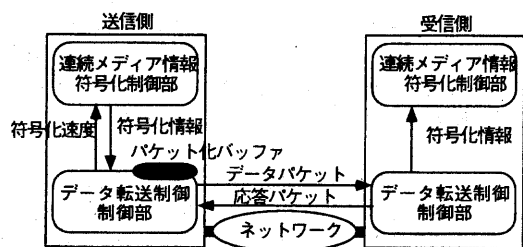


図1: 連続メディア情報転送プロトコルの構成

(2) 輻輳によるQoSの劣化および輻輳の終了によるQoSの向上の双方を素早く検出するように、データ転送制御部に以下のような機能を用いる。

- 連続メディア情報を運ぶデータパケットと、それに対する応答パケットを導入する。さらに、両者を対応付けるために順序番号を使用し、またデータパケット長は固定長とし、その送信レートの変更によりデータ転送のレートを制御可能とする。
- QoSの劣化を検出するために、データパケットと対応する応答パケットにより、データパケットに対するRTTと損失を測定する。RTTが増加した場合または損失率が一定の値を越えた場合は、ネットワークに輻輳等が生じ、QoSが劣化したと判断する。
- QoSの劣化を検出すると、送信レートを減少するために、データパケットの送信間隔を増加する。その方式は、輻輳の程度が大きい場合には、大きく増加させ(TCPのスロースタートに対応)、軽微な場合には、緩やかに増加させる(TCPの輻輳回避に対応)。
- 新たに採用した送信レートにおいて、QoSの劣化が観測されない場合は、徐々に送信間隔を小さくし、送信レートを上げる。その際、スロースタートと輻輳回避アルゴリズムと同様に、一定の閾値までは指数関数的に増加させ、その後は線形的に増加させる。

(3) 送信レートの変動とは独立に、一定間隔ごとに、データパケットの送信レートを測定し、それに対応する符号化レート(ただし符号化部がサポートする符号化レートから選択する)を決定し、符号化部へ通知する。符号化部は符号化レートが通知されると、次の符号化情報からそのレートをを用いて生成する。

4 詳細手順

4.1 使用する内部パラメータ

本プロトコルでは、次のような内部パラメータを使用する。

- int : データパケットの送信間隔。本プロトコルではデータパケットの送信レートをその送信の時間間隔で制御する。
- int_cong : 輻輳を検出した時点でのデータパケットの送信間隔の値。
- α, β : 輻輳の程度を判定するRTTの増加率。RTTが正常な値(RTTの最小値)の α 倍となると輻輳が発生したと判断する。RTTが α から β 倍となった範囲では、軽微なRTT劣化、RTTが正常値の β 倍を超えると大きなRTT劣化と判断する。
- e : 輻輳と判断するパケット損失率。
- A, B : 輻輳を検出した時点でのデータパケットの送信間隔の増加率。
- a, b : 輻輳が検出されない場合のパケット送信間隔の減少率。
- Δ, δ : 符号化レートの決定するためのパラメータ。

4.2 データパケットの送信レートの変更手順

本プロトコルでは、データパケットごとにRTTと損失率を測定し、RTTが、正常なRTT値(要求するQoSが満足されている場合のRTTの値)よりも増加したか、損失率が一定値を越えると、次の2つの手順のいずれかの手順に従って、送信レートを変更する。

(1) 軽微なRTT劣化に対する処理

RTTが正常値の α 倍から β 倍の範囲で増加したことを検出すると、その時点の int の値を int_cong に代入し、 int の値を新たに、 int_cong の A 倍とする。その後は、 int の変更直後に送出されたパケットに対応する応答パケットの受信ごとに、輻輳が検出されなければ、すなわち、RTTが正常値の α 倍以内であれば、 int を $int_cong \times a$ だけ減少させる、すなわち、

$$int = int - int_cong \times a$$

の処理を行う(線形的な減少)。

(2) 大きなRTT劣化またはパケット損失に対する処理

RTTが正常値の β 倍を超えた場合、または、データパケットの損失率が e を超えた場合は、その時点の int の値を int_cong に代入し、 int の値を新たに、 int_cong の B 倍とする。その後は、 int の変更直後に送出されたパケットに対応する応答パケットの受信ごとに、輻輳が検出されなければ、 int をそれまでの int の b 倍に減少させる(指数的な減少)。減少させた送信間隔が int_cong の A 倍よりも小さくなった場合は、(1)の手順に従う。

送信間隔を変更させる間も、引続き RTT および損失率の測定を行い、QoS の劣化を検出すると、状況に応じて (1) または (2) の手順を行う。

4.3 符号化レートの決定方法

連続メディア情報の符号化レートは、データパケットの送信レートと独立に以下のように決定する。すなわち、一定時間 Δ (RTT に比較して大きな値、例えば数 sec 程度) ごとに、データパケットの送信間隔 int の分布を求め、送出されたデータパケットのうち、割合 δ のデータパケットで転送可能な符号化情報を発生するような符号化レートを、次の一定時間 Δ のレートであるとして、符号化部へ通知する。

4.4 データパケットの送出方法

本プロトコルでは、連続メディア情報符号化制御部の発生する符号化情報を以下のような手順で、データパケットとして送出する。

- 符号化制御部が連続的に発生する符号化情報を、パケット化バッファに蓄積する。
- パケット送出間隔 int ごとに、その時点でパケット化バッファに蓄積された符号化情報を用いてデータパケットを作成する。
- その際、データパケットのサイズよりも、多くの符号化情報がパケット化バッファに蓄積されていた場合は、必要な符号化情報のみをバッファから取り出す。また、パケット化バッファに蓄積された符号化情報の総量が、データパケットのサイズよりも小さい場合は、不足分をパディング情報により補うことにより、固定長のデータパケットを作成する。

5 性能評価方法

5.1 性能評価用プログラム

今回提案したプロトコルの性能評価を行うために、プロトコルシミュレータ REAL [6] を用いたシミュレーション実験を行った。REAL はカリフォルニア大学バークレー校で開発されたパケット網用プロトコルシミュレータであり、パケットの生成/終端ノード (ソース/シンク)、パケット交換ノード (ルータ、ゲートウェイ) を並列に実行し、各ノードの動作をシミュレートする。このシミュレーションでは、物理回線へのパケットの送出時間と、ノード間の伝送遅延が考慮され、プロトコル処理時間は考慮されない。

REAL を使用するためには、シミュレートするネットワーク毎に、必要となるノードのプログラムを作成し、ノ

ド間の接続状況やその間の物理回線情報をノード接続テーブルに指定する必要がある。各ノードのプログラムは、他のノードからのパケットや、タイムアウトを通知するパケットに対する動作を規定する形で実装される。ただし、TCP をシミュレートするソースとシンクのプログラム、および、ルータのプログラムは予め用意してある。

そこで、本プロトコルを実装するために、送信側符号化制御部、送信側データ転送制御部、受信部の 3 つのノードを作成し、さらに次のようなパラメータ値を採用することとした。

- 送信側符号化制御部は、1Mbps から 0.1Mbps 刻みで 0.1Mbps まで、および、50Kbps と 10Kbps の 12 段階の符号化レートをサポートする。また、初期の符号化レートは 1Mbps である。
- 符号化情報の大きさは 125 バイト、データパケットの大きさは 1250 バイトとする。
- プロトコルの内部パラメータの値としては以下を採用した。
 - 輻輳を検出するパラメータについては、RTT の増加率に関して $\alpha = 1.1$ 、 $\beta = 2$ 、パケット損失率について $e =$ 最近の 10 パケットに対して 0.2 とした。
 - 軽微な輻輳検出時においては、パケットの送信間隔を細かく制御し、逆に、大きな輻輳の検出時には大きく増加させることとして、 $A = 1.2$ 、 $B = 16$ とした。
 - パケットの送信間隔の減少率については、軽微な輻輳時には、TCP の輻輳回避と同様に、ゆっくり制御するために $a = 1/16$ とした。また、大きな輻輳時には、スロースタートと同様に、RTT ごとにパケット送信間隔を $1/2$ とするよう、 $b = 1/2$ とした。
 - 符号化レートの決定パラメータについては、 $\Delta = 4$ 秒、 $\delta = 0.7$ とした。

以下に各部の処理内容を示す。

(1) 送信側符号化制御部

入力と、それに対する動作は次のとおりである。

符号化情報送信タイムアウト 符号化情報を送信側データ転送制御部へ送出するとともに、次の符号化情報の送出する時刻に、タイムアウトを通知するよう要求する。

符号化間隔指示 送信側データ転送制御部からの入力で、次の Δ 時間の符号化レートに対する送出時間間隔の値を変更する。

(2) 送信側データ転送制御部

入力と、それに対する動作は以下のとおりである。

符号化情報 パケット化バッファに蓄積する。

パケット送信タイムアウト パケット化バッファから、符号化情報を取り出し、データパケットを作成し送出する。また、次の符号化レートを計算するために、今回のパケット送信のレート(前のパケットとの送信間隔)を、符号化レートに対応した時間間隔(10msecから1sec)に従って分類する。さらに、次の送信タイムアウトを要求する。

応答パケット 受信部からの入力であり、その順序番号により、データパケットの損失およびRTTを求め、4.2節に規定した方法により、データパケット送信間隔を決定する。

符号化速度変更間隔タイムアウト Δ に対応するタイムアウトで、パケット送信間隔の分類に従って、割合 δ のパケットで転送することのできる符号化レートを決定し、送信側符号化制御部へ通知する。さらに、次のタイムアウトを要求する。

(3) 受信部

受信側の符号化制御部とデータ転送制御部に対応する部分であり、以下のような、入力に対する動作を行なう。

データパケット 受信したパケットを破棄し、対応する応答パケットを送信する。

5.2 性能評価実験の条件

実験では、ルータを介してLANが相互接続されたネットワークを想定した。このため、図2に示すように、ルータノードを1Mbpsの回線で接続し、その間の伝送遅延を10msecまたは100msecとした。このようなネットワークで、前節に示した3つの部分を用いた連続メディア情報の転送、ならびに、REALに用意されたTCPソースとTCPシンクを用いたTCP通信を行なわせ、評価を行なった。回線速度や伝送遅延などのネットワーク条件は、図2に示すとおりである。また、シミュレーションにおいては、50秒の通信時間を想定し、ルータには250Kバイトのバッファを持たせ、ルータによるパケット損失は起こらない状況を想定した。

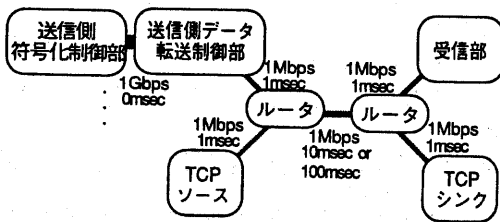


図2: 性能評価のためのネットワーク条件

6 性能評価結果

シミュレーションにより、次のような、性能評価結果を得た。各々の結果の図においては、送信側における符号化情報の送信間隔(ただし、パケットの送信に換算するために10倍したもの)と、送信側におけるデータパケットの送信間隔の時間変動を示している。また、符号化情報の送信間隔を図示するために \diamond を、データパケットの送信間隔を示すために $+$ を、それぞれ使用した。

(1) ルータ間の伝送遅延が10ミリ秒、または、100ミリ秒で、本プロトコルによる通信を2対多重化した場合の、1つの通信の結果を、それぞれ、図3と、図4に示す。

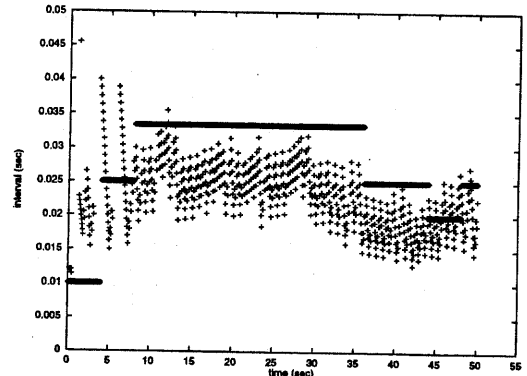


図3: 2対多重化の場合の送信状態(伝送遅延 10msec)

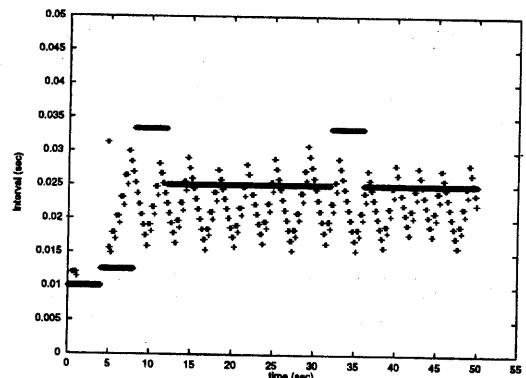


図4: 2対多重化の場合の送信状態(伝送遅延 10msec)

これらの結果においては、ルータ間の回線において輻輳が生じたため、ネットワークのQoSが保証されなくなり、これに対応するために、送信側がデータパケットの送信間隔と、符号化速度を変化させながら通信を行なっている。この時、プロトコル制御部のパケット送信間隔は頻繁に変動し、符号化部の送信間隔は緩やかに変動している。また、2対の連続メディア情報が転送される場合、0.5Mbpsの符号化速度(送信間隔:20msec)となるのが最適である。一方、結果から得られた符号化速度は、0.4Mbps

(送信間隔: 25msec) から 0.3Mbps (送信間隔: 33msec) であった。さらに、4 対の場合でも、符号化部は理論値の 0.25Mbps (送信間隔: 40msec) に対して、0.2Mbps (送信間隔: 50msec) から 0.1Mbps (送信間隔: 100msec) で符号化情報を転送している。

(2) ルータ間の伝送遅延が 10 ミリ秒で、本プロトコルによって、初期符号化速度 1Mbps (パケット送信間隔: 10msec) と 0.5Mbps (パケット送信間隔: 20msec) の連続メディア情報を転送した場合を評価した。多重化の結果、初期符号化速度 1Mbps の通信は 0.4Mbps から 0.5Mbps に、0.5Mbps の通信は 0.3Mbps に、それぞれ符号化速度が減少した。

(3) ルータ間の伝送遅延が 10 ミリ秒で、本プロトコルによる転送の際に、バックグラウンドトラフィックとして 1 対の ftp 通信 (TCP のウィンドウサイズ: 8K バイト) を行なった場合の結果を図 5 に示す。また、ルータ間の伝送遅延が 100 ミリ秒の場合の結果を図 6 に示す。

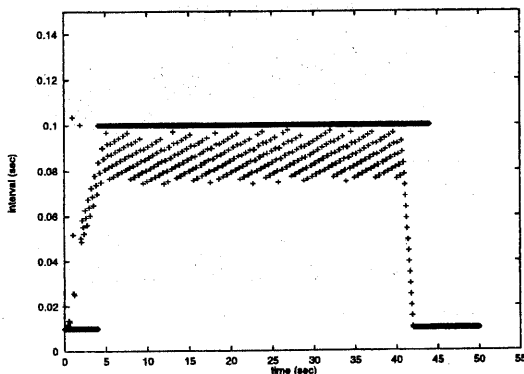


図 5: ftp 通信との多重化の場合の送信状態 (伝送遅延 10ms)

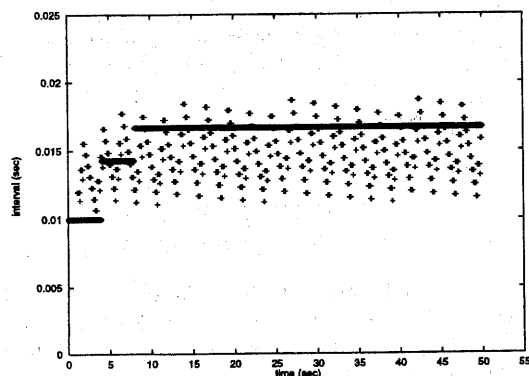


図 6: ftp 通信との多重化の場合の送信状態 (伝送遅延 100ms)

ルータ間の伝送遅延が 10 ミリ秒の場合は、ftp 通信による輻輳の程度が大きいため、符号化速度が最低の 10Kbps となっている。この場合は、さらに QoS の劣化を検出し

ても、これ以上は符号化速度を減少させない。また逆に、ルータ間の伝送遅延が 100 ミリ秒の場合は、ftp 通信を行なっても、符号化速度の減少は比較的小さい。

(4) ルータ間の伝送遅延が 10 ミリ秒または 100 ミリ秒で、本プロトコルによって通信を行なっている最中に、もう 1 対の本プロトコルによる通信を多重化した場合の評価を行った。この結果から、プロトコル制御部において、輻輳の開始と終了を素早く検知していることが確認できた。

7 むすび

本論文では、QoS を保証しないネットワークにおいて、動画などの連続メディア情報を、ネットワークの輻輳状況に応じて、符号化レートを変えながら転送する通信プロトコルを提案するとともに、シミュレーションにより、その性能評価を行った結果を示した。本プロトコルは以下のような特徴を持つ。

- データパケット毎に RTT (Round Trip Time) を検査し、その増加によりネットワークの輻輳を検知する。
- 輻輳が検知された場合は、一旦データパケットの送出速度を減少させ、その後も、RTT 毎に輻輳の有無を調べ、検知されなければ、パケット送出速度を徐々に増加させる。
- データパケットの送出速度と、符号化速度を独立に扱い、データパケット送出速度が頻繁に変動しても、符号化速度は緩やかに変化させることを可能とし、連続メディア情報の品質を一定に保つ。ここで、QoS の向上の検出を目的としてパケット転送速度を増加するためには、パディング情報を用いた無駄なデータ転送を用いている。しかし、階層型画像符号などを用いて、必須な符号化情報と、高品質のためのオプションの符号化情報とを区別できる場合は、後者を用いて転送速度の増加を行うことも考えられる。

参考文献

- [1] T. Sakatani, "Congestion Avoidance for Video over IP Networks" Proc. of International COST 237 Workshop: Multimedia Transport and Teleservices, Nov. 1994.
- [2] H. Kanakia, et al., "An adaptive Congestion Control Scheme for Real-Time Packet Video Transport," Proc. of ACM SIGCOMM '93, Sept. 1993.
- [3] J. Bolot and T. Turletti, "A Rate Control Mechanism for Packet Video in the Internet," Proc. of INFOCOM '94, June 1994.
- [4] 木村, 加藤, 鈴木, "QoS を保証しないネットワークにおける連続メディア情報転送プロトコルに関する一考察," 情処第 54 回全国大会, 4N-6, March 1997.
- [5] W. Stevens, "TCP/IP Illustrated, Volume 1 [The Protocol]," Addison-Wesley, Feb. 1994.
- [6] S. Keshav, "REAL: A Network Simulator," Computer Science Department Technical Report 88/472, UC Berkeley, Dec. 1988.