

October 21, 2003

RT0555
Multimedia; Network 28 pages

Research Report

Playtime-Oriented Retransmission Control Method for Multimedia Streaming

Hiroaki Hagino, Masaaki Taniguchi, Hirotaka Komaki, et al.

IBM Research, Tokyo Research Laboratory
IBM Japan, Ltd.
1623-14 Shimotsuruma, Yamato
Kanagawa 242-8502, Japan



Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

Limited Distribution Notice

This report has been submitted for publication outside of IBM and will be probably copyrighted if accepted. It has been issued as a Research Report for early dissemination of its contents. In view of the expected transfer of copyright to an outside publisher, its distribution outside IBM prior to publication should be limited to peer communications and specific requests. After outside publication, requests should be filled only by reprints or copies of the article legally obtained (for example, by payment of royalties).

論文

マルチメディアストリーミングにおける
再生時刻を考慮したパケット再送制御

Playtime'Oriented Retransmission Control Method
for Multimedia Streaming

萩野 浩明（株式会社 NTT ドコモネットワーク研究所）
[正会員 199512412]

宮崎雄一郎（株式会社 NTT ドコモネットワーク研究所）

尾上 裕子（株式会社 NTT ドコモネットワーク研究所）
[正会員 199205396]

渥美 幸雄（専修大学経営学部）
[正会員 197806118]

谷口 雅招（日本 IBM 株式会社）
[正会員 199508394]

駒木 寛隆（日本 IBM 株式会社）

山内 長承（東邦大学理学部）
[正会員 198262825]

連絡先

萩野浩明
〒239-08536 神奈川県横須賀市光の丘 3-5
NTT ドコモ R & D センター ネットワーク 研究所
TEL: 046-840-3870 :FAX: 046-840-3781
E-mail: hagino@netlab.nttdocomo.co.jp

概要

本論文では、マルチメディアストリーミングにおける再生時刻を考慮したパケット再送制御方式を提案する。マルチメディアストリーミングではクライアントはパケットを受信しながらコンテンツの再生を行う。そのため、コンテンツの再生が開始されるとそれぞれのパケットについて再生すべき時刻が決定し、それまでにパケットがクライアントに到着しなければそのパケットはクライアントアプリケーションによって破棄される。提案方式では、サーバがRTTとクライアントバッファの時間的蓄積量を比較し、損失したパケットを再送した際に再生時刻までにクライアントに到着するかどうかを予測する。そして再生時刻に間に合うと予測されるパケットのみが再送される。これによって、パケットが損失した際の無駄なパケット再送を回避することが可能となる。さらに本論文では提案方式の評価実験を行い、単純なNACKベースの再送制御との比較を行う。この実験結果から、提案方式が再送されるパケット数を大きく減少させるにも関わらず、再送が成功するパケット数がほとんど低下しないことを示す。

Abstract

In this paper, we propose playtime-oriented retransmission control method for multimedia streaming. In multimedia streaming, playtime of each packet is decided when a client starts playing contents. If a packet does not arrive at a client until its playtime, the packet is discarded by the client application. In our method, a client sends a server NACK packet when it detects packet losses. NACK packet includes not only identifiers on lost packets but also playtime of a top packet of client buffer to a server. The server subtracts playtime of the top packet of client buffer from that on lost packet, and compares result of the calculation to RTT between the server and the client. If former is larger, the server retransmits the lost packets to the client. If not, the server retransmits no packets. Therefore, the number of unnecessary retransmission of packets is decreased. Moreover, we estimate our method and compare it to the simple retransmission method using NACK packets. From the results, we show that our method hardly decreases the number of retransmitted packets which arrive at a client until their playtime while our method drastically decreases the number of retransmitted packets.

1 はじめに

近年、有線、無線を問わずネットワークの広帯域化が進んでいる¹⁾²⁾。これに伴い、より高品質のサービスに対する要求が高まっている。特に、これまでのネットワークではネットワーク上での交換が困難であった、動画などのマルチメディアコンテンツに対する期待が大きい。動画配信サービスは、ダウンロード型サービスとストリーミング型サービスとに大きく分類できる。ダウンロード型サービスに対する期待が大きい。動画配信サービスは、ダウンロード型サービスとストリーミング型サービスとに大きく分類できる。ダウンロード型サービスは、動画をファイルとしてサーバからダウンロードして、

ロー

ローカルディスクに保存し、保存したファイルを動画再生ソフトで再生する。このサービスでは、配信されるファイルが例えば動画コンテンツであるか、テキストファイルであるか、といったことは制御にほとんど影響を与えない。従来、狭帯域ネットワークで交換されていたファイルのサイズが大きくなったものが、広帯域ネットワークにおいてダウンロード型サービスでやりとりされるコンテンツであると考えられる。ネットワークの広帯域化によって、サイズの大きいファイルの交換が現実的なものとなったが、これはTCPなどの従来技術によって十分実現可能である。

一方、ストリーミング型サービスは、ユーザはクライアント端末からサーバ

にアクセスし、サーバから動画コンテンツや音楽/音声コンテンツの配信を受けながら、同時に再生を行うものである。ローカルディスクに保存する必要がないため、不正コピーが行われる可能性が低く、著作権のあるコンテンツ配信に有効である。また、ライブ放送など、リアルタイム性が必要とされるコンテンツの配信手段として重要であると考えられる。

ここで、ネットワークにおけるパケットの損失は、マルチメディアストリーミングのサービス品質を大きく劣化させる要因の一つである。特に、Mpegフォーマットでエンコーディングされたコンテンツなどのように、フレーム間の差

分情報を用いているコンテンツの場合、一つのパケット損失によって、さらに多くのフレームが乱れてしまうことがある。しかし、多くのマルチメディアストリーミングサービスでは、リアルタイム再生を実現するためにコンテンツ配信中に RTP3)/UDP を用いているため、パケットの損失をリカバリすることはでき

ない。この問題を解決するために、IETF では、RTP を用いてパケットを再送するためのプロトコル 4) が提案されている。現在、このインターネットドラフトは expired されているが、同様の内容のものが別のドラフトとして提案されている 5) 6)。しかしこれらの差分は主にパケットフォーマットの軽微な変更であり、方法論や有効性の議論にはほとんど影響がない。そこで以下の議論では文献[4]を IETF 提案として扱い、新しいフォーマットへの対応は今後の課題とする。文献[4]で提案されているプロトコルでは、クライアントは RTP の制御プロトコルである RTCP を用いてサーバに再送要求メッセージを送信し、サーバは再送要求メッセージに指定されているパケットのうち、重要度の高いものを再送する。しかし、ストリーミング型サービスでは損失したパケットを再送しても、そのパケットが構成するフレームの再生時刻までにクライアントに到着しなければそのパケットは無駄になってしまう上に、再送によってサーバがクライアントに送信する総トラフィックが増加し、メインのストリームとして送信してい

るパケットの到着時刻までが遅くなってしまう可能性もある。以上から マルチメディアストリーミングでは、再送によるさらなる通信状況の悪化を避けるために、再送するパケットを厳選し、本当に必要なパケットのみを再送しなければならない。

そこで本論文では、クライアントでのコンテンツ再生状況を考慮して、再送するパケットが再生時刻に間に合うかどうかを計算し、間に合うと考えられるパケットのみを再送する、リアルタイム再送方式を提案する。この方式によって、無駄な再送を極力抑えた上でのサービス品質改善が可能になり マルチメディアストリーミングでの再送制御が現実的なものとなる。さらに I E T F で提

案されている RTP 再送プロトコルを拡張してプロトコルの設計を行い、ストリーミングシステム上に実装して評価実験を行う。実験結果から、再送時に再生時刻を考慮した再生判定を行うことで、再送成功パケット数をほとんど変えることなく、再送パケット数を大幅に減らせることを示す。

以下では、まず 2 章においてリアルタイム再送方式について説明し、3 章では IETF 提案を拡張したプロトコルの設計について述べる。4 章では、プロトコル

の実装と検証結果について述べる。最後に、5章で本稿のまとめを行う

2.リアルタイム再送方式

1章で述べたように、マルチメディアストリーミングでは、損失したパケットを再送しても、そのパケットが構成するフレームの再生時刻までにクライアントに到達しなければそのパケットは無駄になってしまう。さらに、マルチメディアストリーミングでは、サーバが送出するビットレートよりも実際に利用可能な帯域幅が小さいとき、ネットワークに滞留するパケットはサーバが想定す

る時刻よりも遅れてクライアントに到着することになる。これが累積されると、いずれクライアントがあるフレームを再生しようとしたときにそのフレームが

クライアントバッファ内に存在しない、という状況が起こり得る。このような
ことを防ぐために、マルチメディアストリーミングシステムには
QOS 制御を行うこ
とでネットワーク帯域の変化に伴ってネットワークを通過す

るデータのビットレートを調整しているものもある 7)8)。しかし、QOS 制御によってメインのストリームにおけるビットレートが適性に保たれていても、そこへさらに再送パケットが加わることで帯域を圧迫し、その結果上記のような状況に陥る可能性がある。本章では、この問題を解決するために筆者らが提案す

る。リアルタイム再送方式について説明する。

再送制御の流れ

マルチメディアストリーミングでは、サーバおよびクライアントにおける処理遅延が大きいとコンテンツのリアルタイム性が損なわれる可能性がある。

リアルタイム再送方式では、サーバおよびクライアントの処理遅延を小さくす

るために、TCP などで用いられる ACK に基づいた再送ではなく、NACK に基づいて再送を行う。クライアントは、サーバからマルチメディアストリーミングを受信しているときにパケットの損失を検出すると、再送要求メッセージを作成しサーバへ送信する。再送要求メッセージには、損失したと考えられるパケットを示す識別子と、クライアントバッファの先頭にあるパケットが構成するフレームの再生時刻が含まれる。なお、各ストリーミングパケットが構成す

るフレームの再生時刻は、それぞれのパケットのヘッダに記載されているものとする。

サーバは、クライアントにストリーミングパケットを送信した後、そのパケットを再送バッファと呼ぶバッファにバッファリングしておく。サーバはクライアントから再送要求メッセージを受け取ると、その中で指定されているパケットが再送バッファ内に存在するかどうかを調べる。存在していない場合は、パケットの再送は行わない。存在していた場合、まず IETF 提案の RTP 再送と同じように、パケットの重要度を調べる。例えば、Mpeg フォーマットコンテンツでは、単体でフレームを表示できる I ピクチャを構成するパケットは重要度が高く、差分階報で構成される P ピクチャを構成するパケットは重要度が低いと考えることができる。再送要求メッセージによって指定されているパケットの重要度が低い場合、そのパケットは再送されない。重要度が高いパケットであった場合、このパケットを再送したときに再生時刻までにクライアントに到着可能かを計算で見積もる。その結果、間に合うものと判断されたパケットのみを再送する。

クライアントでは、再送パケットを受け取ると、そのパケットの再生時刻を調べ、再生時刻よりも先に到着している場合のみ、パケットの順序が正しくなるように、クライアントバッファの中へ挿入する。再生時刻より遅く到着したパケットは、バッファリングせずに廃棄する。

2.2.リアルタイム性を考慮した再送判定

本節では、2.1 節で述べた再送制御において、再送したパケットが再生時刻までにクライアントに到着するか判定する方法について述べる。

このときの計算には、以下に示す変数を用いる。

- ・ サーバークライアント間の RTT.

サーバはクライアントとの間で交換する制御パケットを用いて、RTT を定期的に測定しているものとする。

- ・ 再送要求メッセージ作成時にクライアントバッファの先頭にあったパケットの再生時刻.

2.1 節で述べたように、この情報は再送要求メッセージに含まれている

- ・ 再送対象であるパケットが構成するフレームの再生時刻.

この情報はストリーミングパケットのヘッダに含まれる.

これらの変数が次の条件を満たすとき、サーバはそのパケットを再送する.

$$n > n + RTT + a$$

ここで、 n は再生対象であるパケットが構成するフレームの再生時刻を、 n は再生要求メッセージ作成時にクライアントバッファの先頭にあったパケットの再生時刻を、 RTT はサーバクライアント間の RTT を示す。また、 a はサーバ、クライアントにおける処理遅延や誤差を取り扱うための変数である。副こは例えば、サーバおよびクライアントのバッファにおける処理遅延やネットワーク伝送による遅延などが含まれる。これらの値は状況によって変化するため、システム構築の際に試行を重ねて決定されるべきである。図 1 にパケット再送のシーケンスを示す。図 1 にもあるように、上記の条件式は再送要求メッセージが作成された時刻を起点として、再送パケットがクライアントに到着すると考えられる時刻が到着期限時刻よりも早いかどうかを調べるものである。この条件を満たすパケットのみを再送することで、再送しても無駄になってしまうパケットの再送を防ぎ、メインストリーム以外のトラヒックの増加を最低限に抑えることができる。

3.リアルタイム再送制御プロトコルの設計

筆者らは 2 章で述べたマルチメディアストリーミングのための再送制御を、IETF で提案されている RTP 再送プロトコルを拡張することで実現する。本章では、そのプロトコルの設計について述べる。

3.1. ペイロードパケット

IETF で提案されている RTP 再送プロトコル⁴⁾では、従来の RTP ペイロードパケットのヘッダに、パケットの重要度を示す PRI フィールド、以前の重要度の高いパケットからのシーケンス番号の差を示す DSN フィールド、以前の重要度の高いパケットのタイムスタンプとそのパケットのタイムスタンプの差を示す DT フィールドが追加されている。リアルタイム再送制御方式においてもパケットの重要度を考慮しているため、PRI フィールドの存在は有効であるのだが、その他の追加フィールドの必要性が低いと考えられるため、ペイロードパケットヘッダには従来の RTP ペイロードパケットヘッダを用いる。

3.2. 再送パケット

IETF で提案されている RTP 再送プロトコルでは、再送されるパケットをメインストリームのパケットと区別するために、再送のためのヘッダを改めて定義し、再送対象となる RTP パケットを再送パケットヘッダでカプセル化する方法がとられている。カプセル化を行うことで、再送されるパケットをメインのストリームに挿入し、一連のシーケンス番号を割り振ることができる。こうすることで、クライアントは再送されたパケットの損失も検出することができる。

しかし、筆者らはこの再送のための特殊なヘッダを用いず、TCP などと同じように、メインストリームにおいてクライアントへ送信したパケットをそのまま再送する方法を採用した。この方法では再送されたパケットの損失は検出できない。しかし、リアルタイム性の高いストリーミング通信の特性を考慮すると、2度目の再送を行ってもそのパケットが再生時刻までに到着する可能性が低いと考えられるため、サーバおよびクライアントにおける処理遅延を小さくするためにこのような設計を行った。

3.3. 再送要求パケット

RTCP の RR(Receiver Report)では、ロスパケット数を記述するフィールドは定義されているものの、損失したパケットを同定する情報は記述できない。RTP での再送を実現するために IETF で提案されている RTP 再送プロトコルで

は、RTCP の APP メッセージを用いて再送要求パケットを実現している。APP パケットはアプリケーションが自由に利用可能なパケットで、基本となるパケットヘッダのみが定義されており、それに続く部分は自由に定義可能である。本研究では、文献[4]で提案されている RTP 再送プロトコルにおける再送要求パケットのフォーマットに基づいて再送要求パケットを設計する。このパケットフォーマットを図 2 に示す。なお、RTP SSRC フィールド以下が RFC1889 で定義された APP パケットフォーマットに追加するフィールドである。

図中の各フィールドについて以下で説明する。

V フィールド: RTP バージョン。ここでは、V=2

P フィールド: パディングビット。

Subtype フィールド: サブタイプ。Subtype=1(再送要求)。

PT フィールド: RTCP パケットタイプ。PT=APP=204。

Length フィールド: パケット長。

Sender SSRC/CSRC フィールド: パケット送信元の SSRC もしくは CSRC。

Name フィールド: APP パケット名。Name="RETX"。

RTP SSRC フィールド: RTP パケットの SSRC。文献[4]における再送要求パケットに新たに追加するフィールド。セッションレベルでの宛先同定を行うために追加している。

PID フィールド: 損失したパケットのシーケンス番号。

BLP フィールド:PID で指定されたパケットの後続 16 パケットの損失を示すためのフラグとして用いる。最上位ビットが PID + 16, 最下位ビットが PID + 1 の RTP パケットに対応し, パケットが損失した場合, そのパケットに対応するビットに 1 をセットする。

TTP フィールド:クライアントバッファの先頭パケットの RTP タイムスタンプ。文献[4]における再送要求パケットに新たに追加するフィールド。

クライアントにおいて, パケットの損失は到着するパケットのシーケンス番号の不連続として検出される。クライアントはパケットの損失を検出すると, すぐに再送要求パケットを作成する。このとき, 連続して損失したパケットが複数であった場合に BLP フィールドを用いる。

3.4. サーバの動作

再送要求パケット受信時のサーバの動作フローを図 3 に示す。サーバでは再送要求パケットの PID フィールドおよび BLP フィールドで指定される全てのパケットに対して, 重要度と再送時刻の判定を行い, 2 つの条件を満たすパケットのみを再送バッファにキューイングする。

4. リアルタイム再送制御方式の実装と評価

本論文で, 提案するリアルタイム再送制御方式をストリーミングシステム上に実装する。ストリーミングシステムには, 筆者らが研究を進めているモバイルマルチメディア QOS 制御システム 9)10)11)を用いる。

4.1. モバイルマルチメディア QOS 制御システムへの実装

筆者らが研究を進めるモバイルマルチメディア QOS 制御システムでは, サーバとクライアントは 1 対 1 で通信を行う。クライアントはサーバの提供する HTML ページにアクセスし, ページ上で視聴したいコンテンツを選択する。サーバは RTSP[12]を用いてコンテンツ配信のためのセッション制御情報をクライアントと交換し, その後 RTP を用いてコンテンツ配信を開始する。クライアン

トはサーバからコンテンツを受信しながら、定期的に RTCP の RR を用いて、受信した最大のシーケンス番号とロスパケット数をサーバへ通知する。サーバではこれらの情報とは別に、RTCP の SR (Sender Report) と RR を用いて RTT を測定している。サーバは以上の情報に基づき、送出量をネットワーク帯域幅の変動に応じて変化させる [10][11]。ただし本論文で行う実験では、再送による効果を明らかにするために送出量制御を行わない。サーバがクライアントに配信する映像は Mpeg-4 ファイルを、音声/音楽は G.723 ファイル、MP3 ファイルおよび AAC ファイルを利用できる。なお、サーバ、クライアントともに Windows2000 をプラットフォームとしている。

リアルタイム再送方式はこのシステム上に、Visual C++ 6.0 を用いて実装した。

4.2. 実験結果

提案方式の有効性を検証するために、4.1 節で述べたシステムを用いて評価実験を行う。以下では、実験の条件と結果、および結果に対する考察を述べる。

4.2.1. 実験の条件

筆者らが行った実験では、ネットワークの遅延やロス率を自由に設定するためにネットワークエミュレータ NE300013)を用いる。再送制御を実装したモバイルマルチメディア QOS 制御システムのサーバとクライアントの間にネットワークエミュレータを設置し、平均遅延を 50ms~250ms、平均パケットロス率を 1%~10%まで変化させる。実験時にはモバイルマルチメディア QOS 制御システムのレート制御機能を OFF とした。サーバがストリーミング配信するコンテンツは 90 秒の長さで映像が 400Kbps の Mpeg4 フォーマット、音声は 8Kbps の G.723 フォーマットのものを用いた。映像は 15fi)s、GOP が IPPPP から成る。この結果、映像、音声を合わせてコンテンツ開始から終了までに 7202 個のパケットがサーバからクライアントへ送信される。クライアントはコンテンツ再生の安定化のために RTP パケットを受信してから 2 秒間バッファリングした後再生を行う。

以上のような条件のもとで、パケットの重要度判定のみを行う IETF 提案の方式と筆者らの提案するリアルタイム再送制御方式とを比較する。なお、リア

ルタイム再送制御方式で用いる計算式の α の値は0に設定する。

4.2.2.実験結果

[再送回数について]

提案方式によって再送されるパケット数が減少する数を調べるために、IETF提案と筆者らの提案方式を比較する。その結果を図4~6に示す。これらの図はあるパケットロス率のときにネットワークの平均遅延を50msから200msまで変化させたときの、再送されたパケット数とロスパケット数との比(以下、再送率)をとったものである。横軸にネットワークの平均遅延、縦軸に再送率をとったものである。それぞれ、図4はパケットロス率が1%のとき、図5はパケットロス率が5%のとき、図6はパケットロス率が10%のときのグラフである。これらのグラフから、IETF提案ではパケットの重要度のみを調べて再送を行うため、遅延が大きくなっても再送率がほとんど変化しない一方、提案方式ではネットワークの平均遅延が大きくなると再送されたパケットが再生時刻までに到着する可能性が小さくなるため、平均遅延が大きくなるとパケット再送率が小さくなることがわかる。ネットワークの平均遅延が小さいときはIETF提案と筆者らの提案方式とはほとんど同じ再送率であるが、特にネットワークの平均遅延が大きい場合に筆者らの提案方式は再送率を大きく低減できる。

[ロスリカバリ率について]

パケットの再送がどれだけ有効に機能しているかを調べるために、再送の結果再生時刻までに到着したパケット数(以下、再送成功数)とロスパケット数の比をロスリカバリ率とし、その値についてIETF提案と筆者らの提案する方式を比較した。その結果を図7~9に示す。それぞれのグラフでは、横軸はネットワークの平均遅延、縦軸はロスリカバリ率を示している。これらの3つのグラフは順に、パケットロス率が1%、5%、10%のときの実験結果である。IETF提案では、サーバはパケットの重要度判定のみを行ってパケットを再送する。図7~9のグラフからIETF提案ではネットワークの平均遅延が大きくなるとロスリカバリ率が小さくなることがわかる。図4~6の結果より、IETF提案ではネッ

トワークの平均遅延が犬きくなっても再送率がほとんど変化しないことが明らかになっている。これらのことから、IETF 提案ではネットワークの平均遅延が大きくなることによって、パケットの無駄な再送回数が多くなるといえる。一方、筆者らの提案方式では、IETF 提案と同様にネットワーク遅延が大きくなるに従ってロスリカバリ率が小さくなるものの、あらゆるパケットロス率の場合において IETF 提案とほとんど同じロスリカバリ率が得られることがわかる。以上より、筆者らの提案するリアルタイム再送制御方式では、ネットワークの遅延時間が長いとき、つまり再送されたパケットが再生時刻までにクライアント

トに到着する可能性の低いときに、再送機能の性能をほとんど劣化させることなく

再送されるパケット数を大きく低減できることがわかる

[再送成功率について]

再送されたパケットの有妨ト生を検証するために再生時刻までにクライアントに到着した再送パケット数とサーバが再送した全パケット数の比(以下、再送成功率)について IETF 提案とリアルタイム再送制御方式を比較した。その結果を図 10~12 に示す。これらの図も順にパケットロス率が 1%、5%、10% の場合を示す。横軸にはネットワークの平均遅延、縦軸には再送成功率をとる。これらのグラフから、まず IETF 提案ではネットワークの遅延時間が大きくなるに従って再送成功率が劣化することがわかる。これは図 4~6 において IETF 提案では再送率がネットワークの遅延に関わらずほぼ一定であること、および図 7~9 において IETF 提案ではネットワーク遅延が大きくなるに従ってロスリカバリ率が小さくなることから自明である。

平均遅延が 50ms の場合、図 10、11 ではリアルタイム再送制御方式と IETF 提案はほとんど同じ値をとっている。これは、2 方式で再送されるパケット数がほとんど同じであるためであるが、遅延が小さい上にロスパケット数もそこまで大きくないことも要因として考えられる。ちなみにリアルタイム再送制御方式が IETF 提案よりも悪い再送成功率を示しているが、これは再生時刻までに間に合わないと判断されたにも関わらず、実際は間に合ったパケットが存在し

たためであると考えられる。一方、図 12 では平均遅延が同じ 50ms の場合でも提案方式の方が明らかに良い結果を示している。これは IETF 提案ではパケットロス率が高いためにパケットの再送が大量に行われることでネットワーク遅延が大きくなり、その結果再生までに間に合わないパケットが多く発生したものと考えられる。リアルタイム再送制御方式では図 6 でもわかるように、わずかながら再送率が低くなっている。リアルタイム再送制御方式は RTT を用いて再送の判定を行うため遅延の変化に敏感であり、このような障害が起こることを回避できているため、図 12 のような結果が得られているものと考えられる。

平均遅延が 250ms の場合、図 10 では IETF 提案とリアルタイム再送制御方式の再送成功率がほとんど同じ値になっている。これはそもそもパケットロス率が 1% であるためにロスパケット数が少なく、その上遅延が大きいために再生に間に合うパケットがほとんど存在しないためである。なお、IETF 提案の方がわずかに優れた結果を示しているのは、平均遅延が 50ms の場合と同様の理由である。

図 11 と 12 では程度の大小に差はあるものの、明らかにリアルタイム再送制御方式の方が優れている。再送しても再生に間に

多いことは図 10 の場合と同様だが、ロスパケット数が多いため IETF 提案の再送数が増加するのに対して、リアルタイム再送制御方式での再送数があまり変化しないためであるものと考えられる。なお図 11 よりも 12 の方がリアルタイム再送制御方式の有効性が低くなっている。これはパケットロス率が高くなることでパケットロスが連続して発生する可能性が高まり、パケットロスの検出が遅れてしまうためである。検出が遅れると再送されたパケットが再生時刻に間に合う可能性が全体的に低下してしまう。つまり再送成功数自体があまりに少ないために平均遅延 250ms におけるリアルタイム再送制御方式の再送成功率が図 11 よりも図 12 の場合の方が低くなるものと考えられる。

また、リアルタイム再送制御方式はネットワーク遅延がある値のとき(今回の実験では 150ms)、再送成功率が最大となることがわかる。この理由を説明するために図 13 を示す。このグラフはパケットロス率が 5% のときにリアルタイム再送制御方式を用いた場合の、遅延の変化に伴うパケット再送数と再送成功数

の変化を示したものである。このグラフの横軸はネットワークの平均遅延、縦軸はパケット数を表している。このグラフから、平均遅延が 150ms 以下では再送成功数がほとんど変化していないことがわかる。これは平均遅延が 150ms の場合、損失したパケットのほとんどの再生が成功するためであるものと考えられる。一方、平均遅延が 150ms を超えると再送成功数は減少を始める。これらのことから、次のように考えられる。

150ms 以下のときは再送成功数に対して再送数が多いためにより再送成功率が得られない。
・150ms 以上では再送成功数が少なくなるためにより再送成功率が得られない

これらの理由によりリアルタイム再送制御方式の平均遅延が 150ms のときに再送成功率が最高になるものと考えられる。なお、最高の再送成功率が得られる遅延の値はクライアントのバッファ量によって変化する。例えば今回の実験ではクライアントがコンテンツの再生を開始する際のバッファリング時間を 2 秒と設定したが、この値が長くなれば再送成功率が最高となるネットワークの遅延時間も変化する。これは逆に、ネットワークの遅延時間が分かれば、リアルタイム再生制御方式を有効に活用できるバッファリング時間を決定できることを示している。またこのグラフから、遅延の値が小さい場合はリアルタイム再送制御方式は多数の再生に間に合わないパケットを再送してしまっていることがわかる。これはネットワーク遅延が小さいためにサーバ、クライアントにおける処理遅延が無視できなくなっているためであると考えられる。つまり本来なら副適正値を代入すべきところを 0 に設定しているために提案方式が正常に動作していないものと考えられる。実際に運営する場合は、テストによって α の適正値を発見する必要がある。

[再送数と再送成功数の比較について]

最後に、ネットワークの平均遅延を 200ms に固定し、パケットロス率の変化

に伴う再送数と再送成功数の変化を IETF 提案とリアルタイム再送制御方式とで比較する。実験結果を図 14, 15 に示す。図 14 は再送数の比較, 図 15 は再送成功数の比較である。横軸はパケットロス率, 縦軸はそれぞれ再送数と再送成功数を示している。この 2 つのグラフから, リアルタイム再送制御方式はパケットロス率の変化してもほとんど再送数に変化していないにも関わらず, 再送されるパケット数を大きく減少できることがわかる。特にパケットロス率が高い場合, 約 9 倍程度まで再送数を低減できる。これによってリアルタイム再送制御方式はストリーミング配信での再送によって発生するトラヒックを大幅に抑え, 再送によってメインのストリーミングの通信性能が低下することを防ぐことができる。ただし, 図 15 を見てもわかるようにパケットロス率が高い場合には, 判定の誤りにより再送されるべきパケットが再送されない場合がある。

先にも述べたように, このような判定誤りについては α の値の設定によって対処可能であるものと考えるが, その決定方法は今後の重要な課題の一つである。

5.まとめ

本稿では, マルチメディアストリーミングのためのリアルタイム再送方式を設計し, 筆者らが研究をすすめるモバイルマルチメディア QOS 制御システム上に実装した。さらに, 筆者らの構築した実験環境において検証実験を行い, 提案方式が従来方式と同等の再送性能を満たした上に再送パケット数を大きく低減できることを示した。今後は, 遅延が小さいネットワークにおける再送成功率の向上について検討をすすめる予定である。

参考文献

- 1) 渾田寛, 有馬秀平, "IMT-2000 ネットワークアーキテクチャ," 電子情報通信学会誌, VOI.82, NO.2, pp.145-152, 電子情報通信学会, 1999.
- 2) 高木雅裕, 加藤紀康, 渋谷尚久, 森谷修, 坂本岳文 / 'ハイブリッドMMAC 実験システムにおけるネットワーク機能," マルチメディア, 分散, 協調とモバイル (DICOM0 2001) シンポジウム論文集, June.2001.
- 3) H. Schulzrinne, S. Casner, R. Frederick, and v Jacobson, "RTP: A Transport Protocol for Real-Time Streaming Protocol," RFC1889, IETF, Jan. 1996.
- 4) A. Miyazaki, H. Fukushima, K. Hata, T. べ viebke, R. Hakenberg, C. Burmeister, N. Ijkatori, S. Okumura, T. Ohno / 'RTP Payload Formats to Enable Multiple Selective Retransmissions," draft-ietf-avt-rtp-selret-03.txt, IETF, May 2002.
- 5) J. Ott, S. べ Venger, N. Sato, C. Burmeister, and J Rey / 'Extended RTP Profile for RTCP-based Feedback (RTP/AVPF)," draft-ietf-avt-rtp-fbfeedback-07.txt, IETF, June 2003.
- 6) J. Rey, D. Leon, A. Miyazaki, V. varsa, and R. Hakenberg / 'RTP Retransmission Payload Format," draft-ietf-avt-rtp-retransmission-09.txt, IETF, Aug. 2003.
- 7) C. Aurrecochea, A. T. Campbell, L. Hauw, "A survey of QOS architectures," Multimedia Systems, Vol. 6, pp. 138-151, 1998.
- 8) D. Chalmers and M. Sloman, "A Survey of Quality of Service in Mobile Computing Environments," IEEE Communications Surveys, Second Quarter 1999, pp. 2-10, 1999.
- 9) 串田高幸, 富田アルベルト, 黒川雅人, 山内長承, 尾上裕子, 安木成比古, 渥美幸雄, 高橋修, "モバイルマルチメディア QOS の構成方法," マルチメディア, 分散, 協調とモバイル (DICOM0 2001) シンポジウム論文集, pp. 723-728, June. 2001.
- 10) 村尾高秋, 汗谷口雅昭, 串田高幸, 萩野浩明, 尾上裕子, 高橋修, "ワイヤレス区間を想定したビデオストリーミングシステム," 情報処理学会マルチメディア通信と分散処理 (DPS) 研究会研究報告, Sep. 2001.
- 11) 萩野浩明, 尾上裕子, 安木成比古, 渥美幸雄, 駒木寛隆, 村尾高秋, 串田高幸, 山内長承, "モバイルストリーミング QOS サーバにおけるファイル切り替え

方式,” 電子情報通信学会技術研究報告 SST2001-141, pp.91-98, Mar.2002.

12) H. Schulzrinne, A.Rao, and R. Lanphier, “Real Time Streaming Protocol,” RFC2326, IETF 1998.

13) NTT エレクトロニクス ネットワークエミュレータ,
<http://www.ne1.co.jp/product/emulator/>.



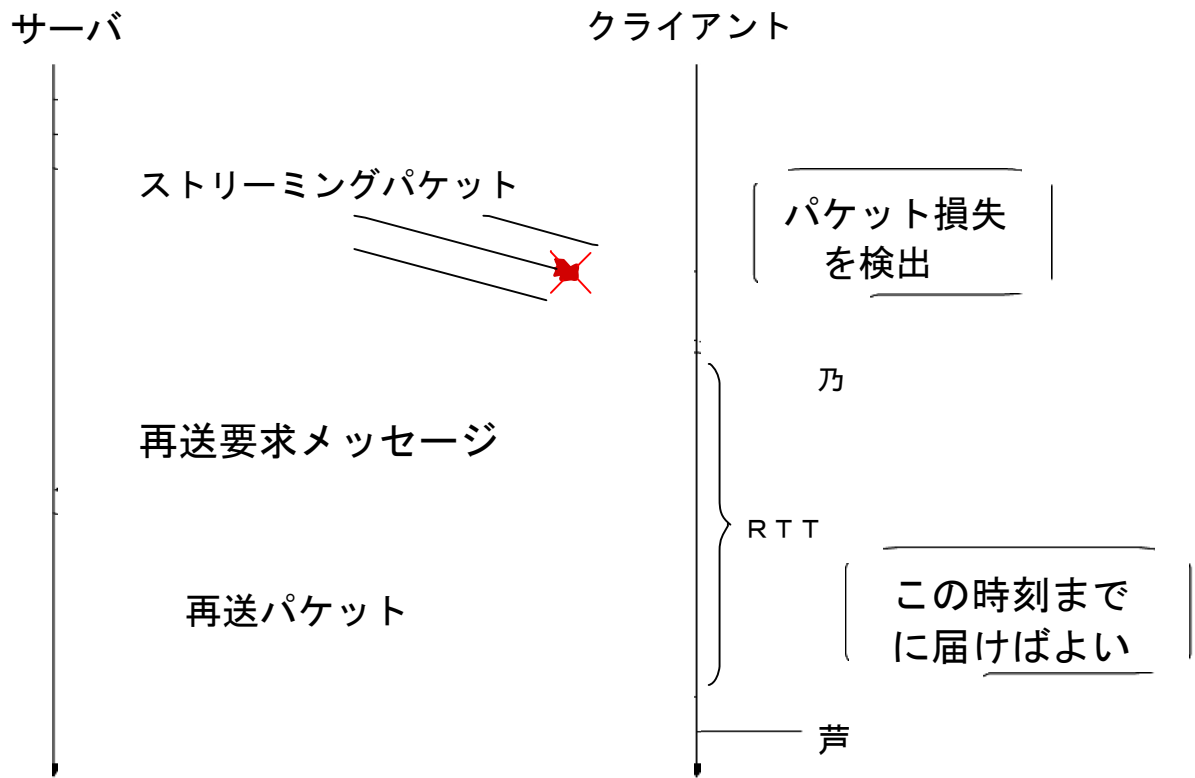


図 1 : パケット再送のシーケンス
 Figure1: sequence of packet retransmission

0 1 2 3
0

1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

V	P	Subtype	PT	Length
Sender SSRC/CSRC				
Name				
RTP SSRC				
PID			BLP	
TTP				

図2 : 再送要求パケットフォーマット
Figure2: format of retransmission request packet

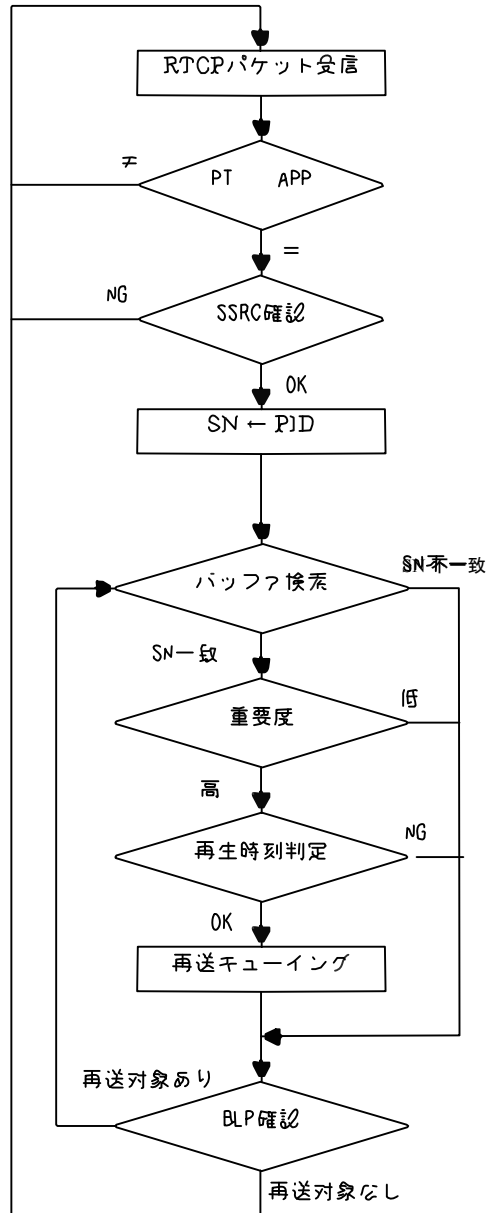


図 3 : 再送処理の流れ

Figure3: flow chart of retransmission control

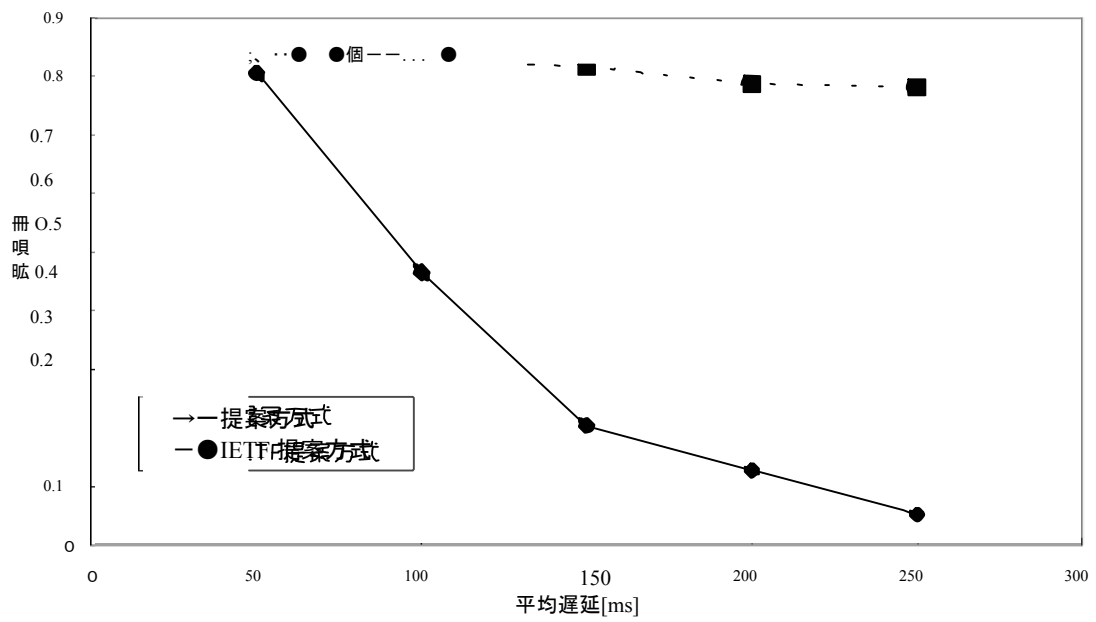


図4：パケットロス率1%における再送率
Figure4: retransmission rate on 1%packet loss rate

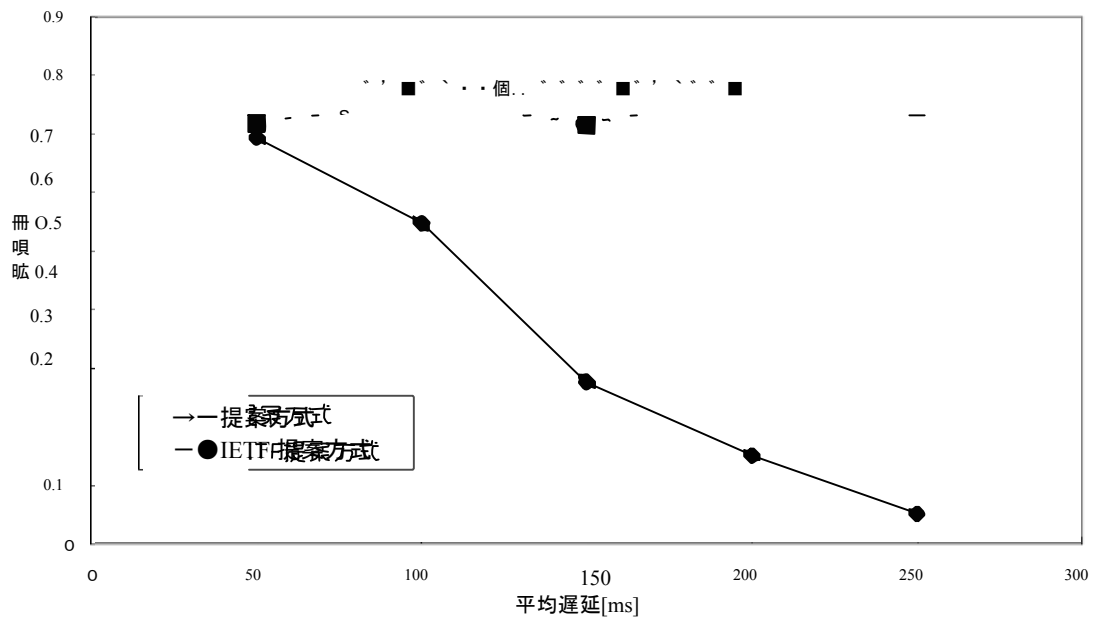


図5：パケットロス率5%における再送率
Figure5: retransmission rate on 5%packet loss rate

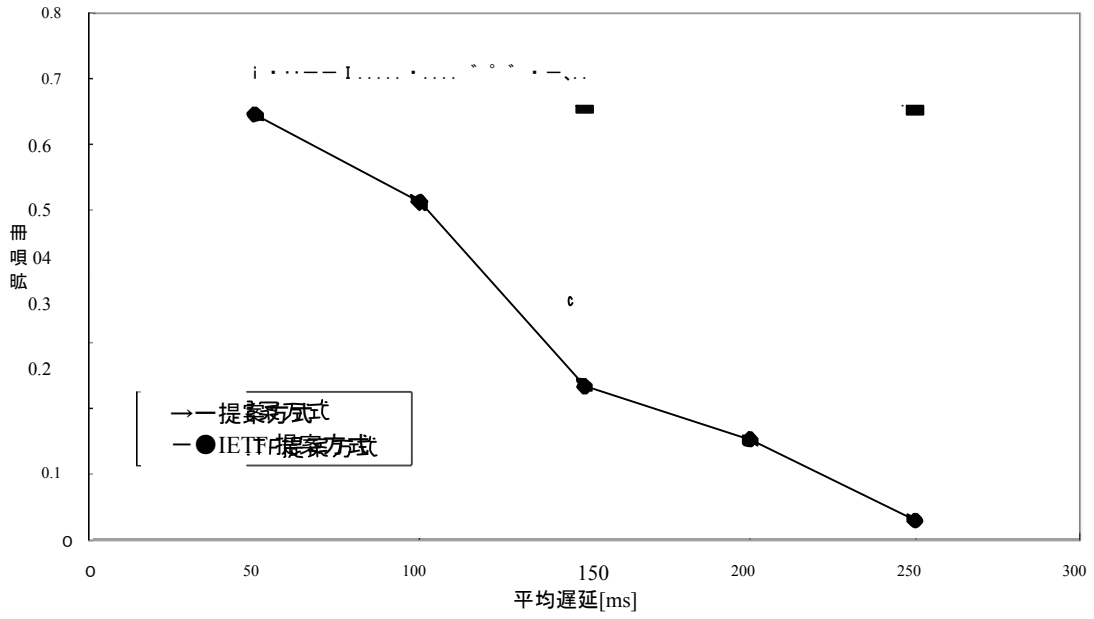


図6：パケットロス率10%における再送率
Figure6: retransmission rate on 10%packet loss rate

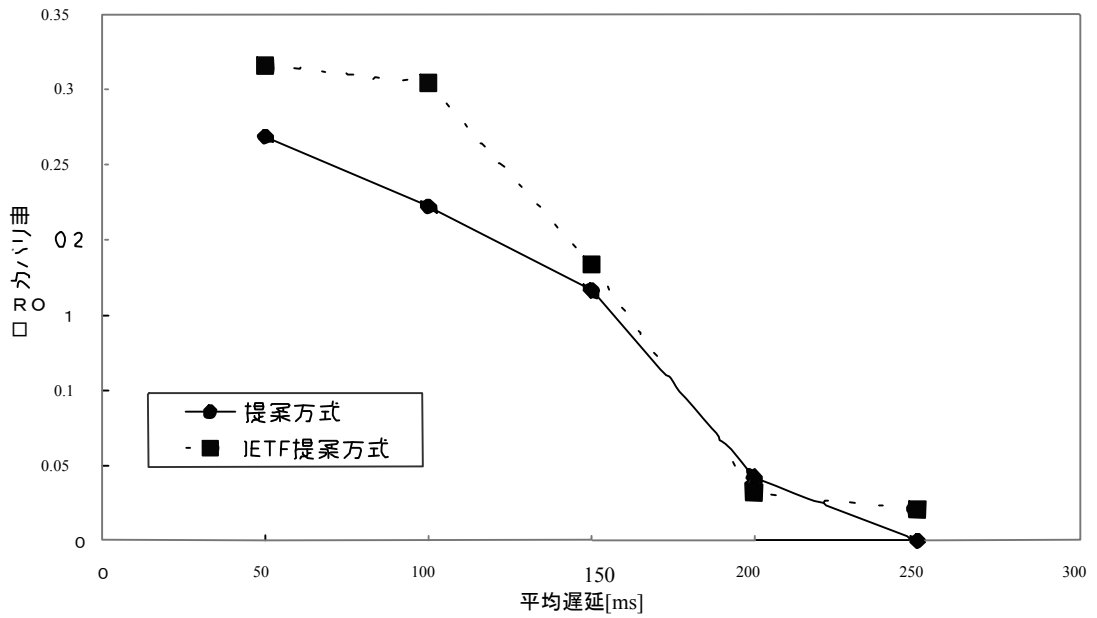


図7：パケットロス率1%におけるロスリカバリ率
Figure7: recovery rate of host packets on 1%packet loss rate

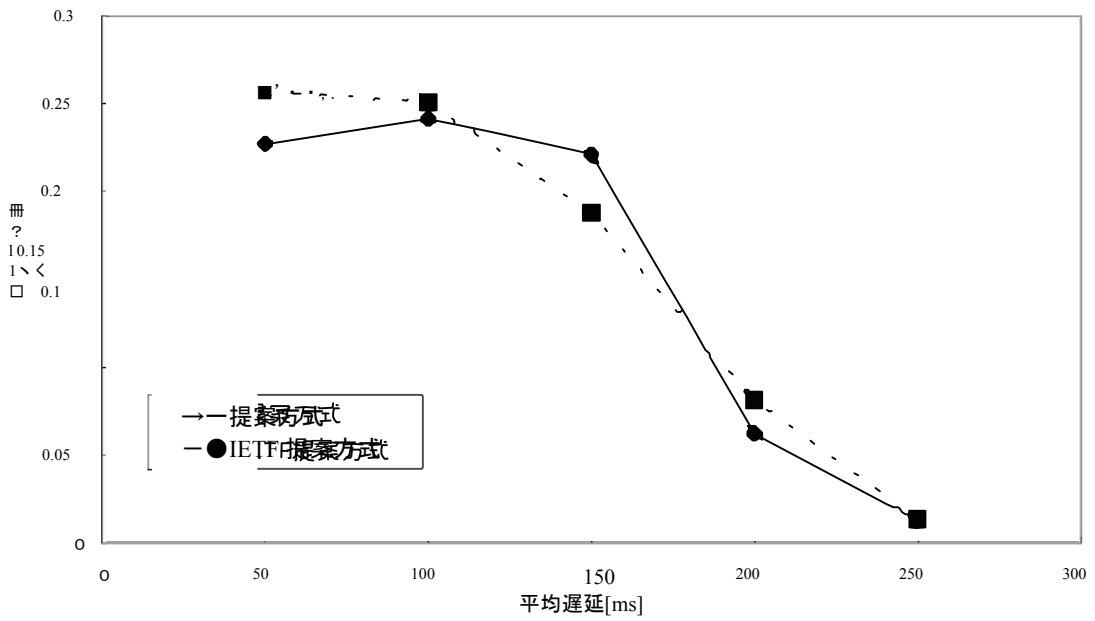


図8：パケットロス率5%におけるロスリカバリ率
 Figure8: recovery rate of host packets on 5% packet loss rate

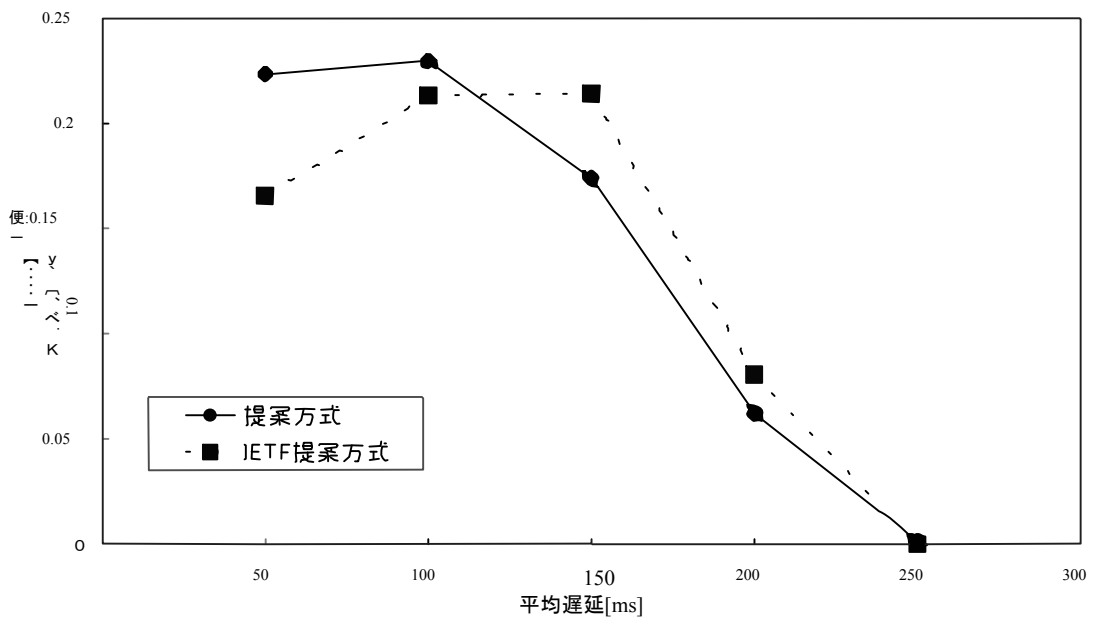


図9：パケットロス率10%におけるロスリカバリ率
 Figure9: recovery rate of on-site packets on 10% packet loss rate

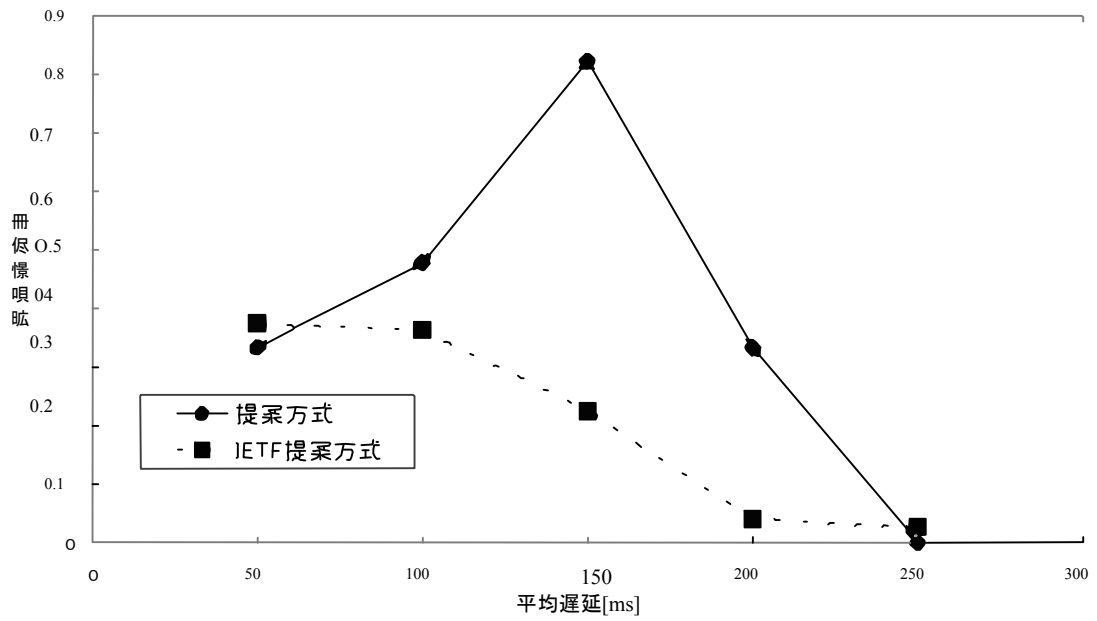


図 10 : パケットロス率 1%における再送成功率
 Figure 10: successful retransmission rate on 1 %packet loss rate

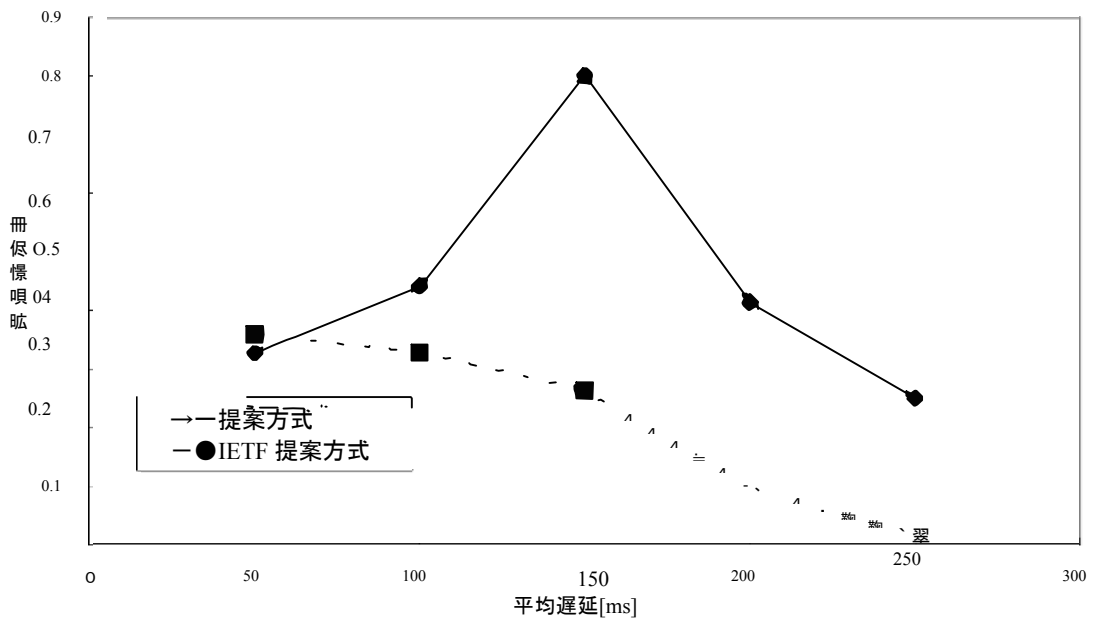


図 11 : パケットロス率 5%における再送成功率
 Figure 11: successful retransmission rate on 5%packet loss rate

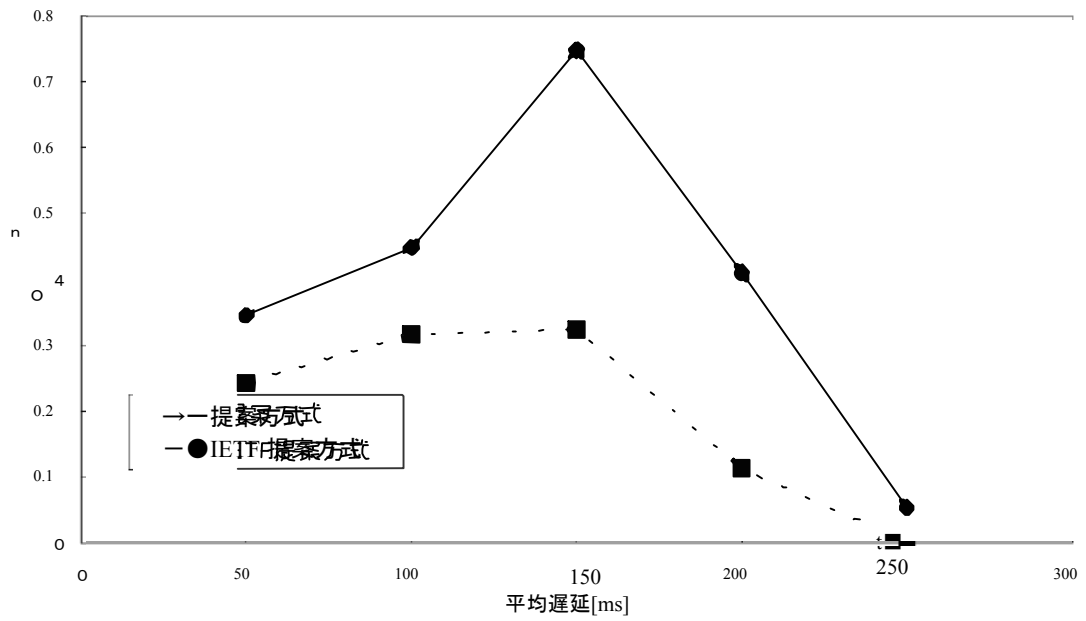


図 1 2 : パケットロス率 10%における再送成功率
 Figure12: successful retransmission rate on 10%packet loss rate

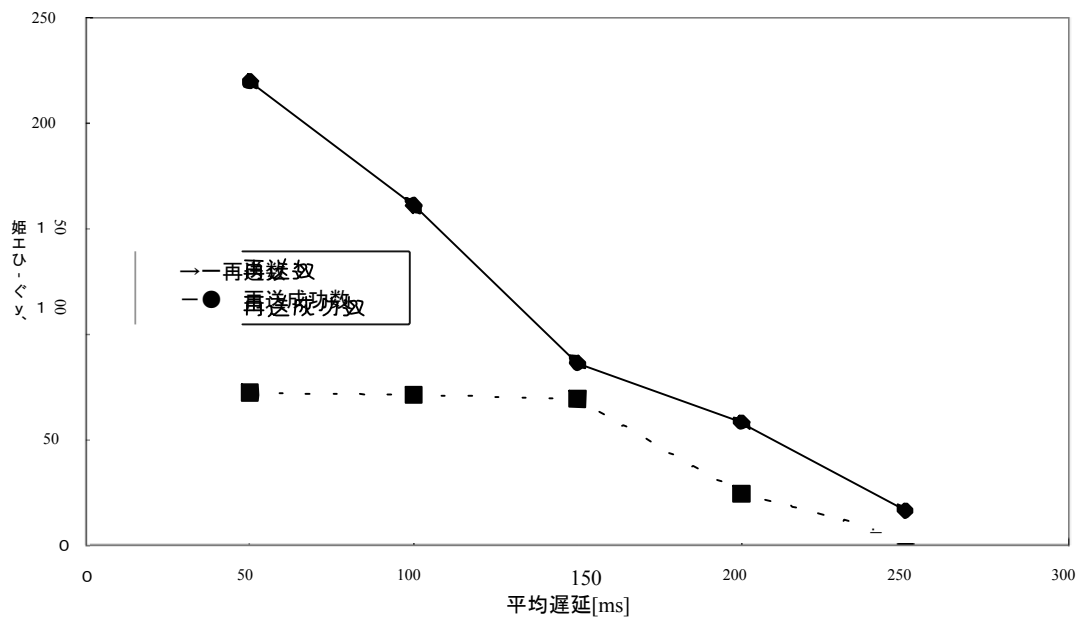


図 1 3 : パケットロス率 5%における再送数と再送成功数
 Figure13: the number of retransmission packets and successful retransmission packets on 5%packet loss rate

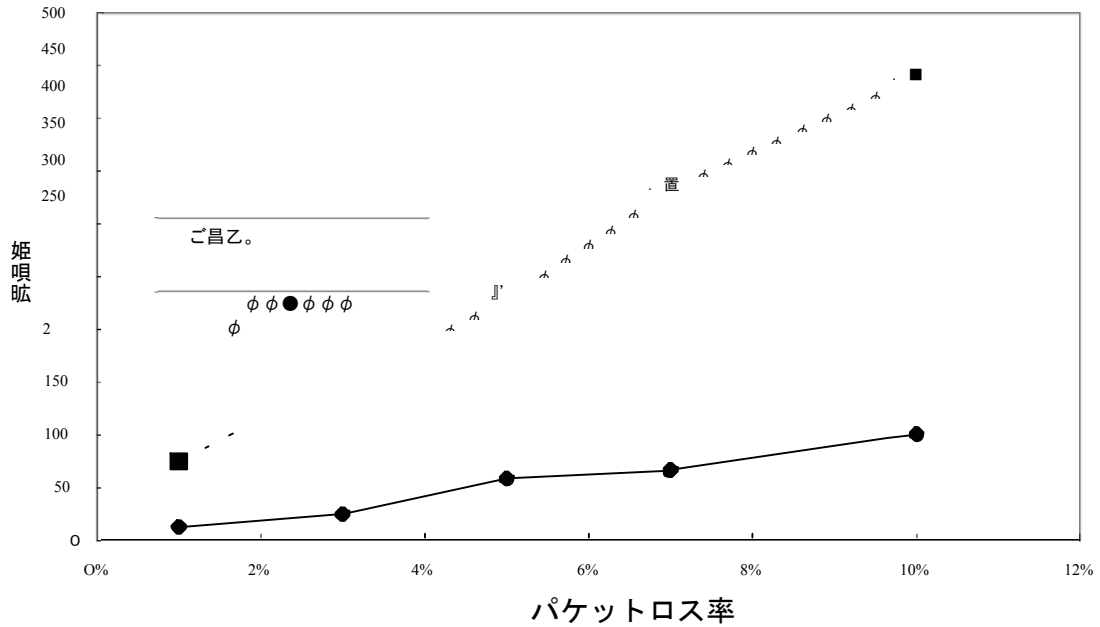


図 1 4 : 平均遅延 200ms における再送数

Figure14: the number of retransmission packets on 200ms latency

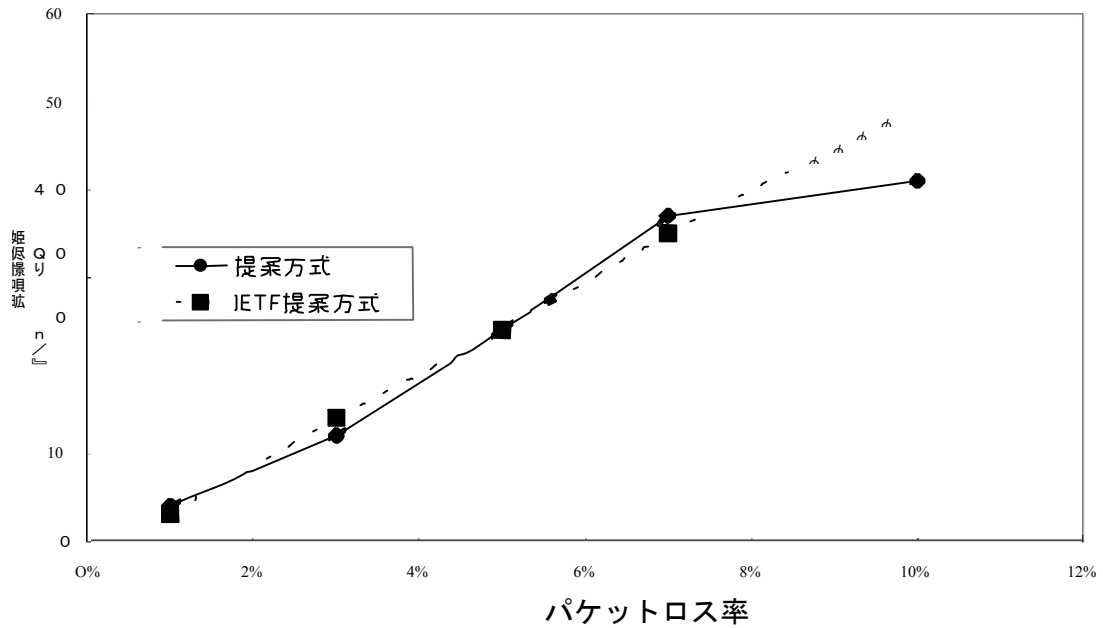


図 1 5 : 平均遅延 200ms における再送成功数

Figure15: the number of successful retransmission packets on 200ms latency