

特別研究報告書

複数経路を活用したバーストパケットロスに
強いストリーミングシステム

指導教員 美濃 導彦 教授

京都大学工学部情報学科

渡部 郁恵

平成17年2月10日

複数経路を活用したバーストパケットロスに強い ストリーミングシステム

渡部 郁恵

内容梗概

インターネットの普及と広帯域化により、インターネット上でのマルチメディアコミュニケーションが様々な形で実現されるようになってきた。マルチメディアコミュニケーションでは映像や音声を伝送するため、大きな帯域が必要である。さらに、双方の教室でインタラクションを行うようなリアルタイム型のアプリケーションの場合、コミュニケーションを円滑に進めるためには遅延をできるだけ少なくしなければならない。

現在のインターネットでは、帯域を確保して品質を保証するといった技術が発展途上にあるため、映像や音声の乱れや途切れの原因となるパケットロスの発生を防ぐことができない。このような品質の低下を引き起こすパケットロスの中でも特に、連続して多くのパケットが喪失するバーストパケットロスは長時間にわたる映像や音声の途切れを引き起こすため、コミュニケーションに重大な影響を及ぼす。パケットロスによる品質の低下を抑える手法はいくつか考えられるが、本論文では、リアルタイム型のアプリケーションで使用することを想定し、遅延を増加させずに品質の低下を抑える、冗長符号化とパケットパスダイバーシティ(Packet Path Diversity)という2つの手法に着目する。

従来から冗長符号化とパケットパスダイバーシティの2つに着目した研究はあった。しかしながら、ストリーム伝送開始後に冗長符号化の冗長度などのパラメータを変更できない、プロトコル拡張により他のシステムとの互換性が失われている、複数のIPアドレスを持っていないとパケットパスダイバーシティを利用することができないなどの問題があった。パケットロス率や帯域といったネットワークの状況は絶えず変化するため、状況の変化に応じて最適な伝送を行うためには、ストリーム伝送中のパラメータ変更が必要不可欠である。また、他のシステムとの互換性を保ち、IPアドレスが1つしかないなどの状況においても利用できるような高い汎用性をシステムに持たせるべきであることは言うまでもない。そこで、本論文では、RTP over RTP、Relay ホスト、経路ごとの冗長符号化の3つを新たに採用し、従来研究の問題を解決した、より頑健なシステムを提案する。

本論文で提案するシステムを使用することにより、バーストパケットロスの影響を大幅に抑えることができる。また、パケットパスダイバーシティを利用することができないシステムと本システムを組み合わせることにより、そのようなシステムにおいてもパケットパスダイバーシティの使用を可能とし、よりバーストパケットロスに強くすることができる。さらに、直接経路を1本しか持っていないため、従来手法では複数の経路を確保することができなかったホスト間でもパケットパスダイバーシティが可能になる。

本論文では、遅延の制約が厳しいリアルタイム型アプリケーションでマルチメディアストリームを伝送するための新たな手法を提案し、その手法を利用したシステムの設計と実装を行った。そして、本システムの有効性を検証するためのシミュレーションを行い、本システムの有効性を確かめた。

A Multimedia Streaming System Robust to Bursty Packet Loss Using Packet Path Diversity

Abstract

Various multimedia communication systems over computer networks are available along with widespread use of the Internet. There is a need for large bandwidth to transmit video and sound. Real-time applications providing interaction between classrooms, for example, require also shorter delay for smooth communications.

We cannot avoid packet losses which cause disorder or cutting-off of streaming because resource reservation techniques are still in development. In particular bursty packet losses, contiguous multiple packet losses, have a critical impact on interactive communications because they cause longer breaks of streams. In this paper delay-sensitive real-time applications are supposed to be the target, so in some ways to prevent such a deterioration in quality, we focus on Forward Error Correction and Packet Path Diversity. These two methods can deal with bad effects of packet losses without latency.

Some contemporary techniques deal with the same combination of Forward Error Correction and Packet Path Diversity. But they have some following problems: parameters cannot be changed once streaming begins, extended protocols make them incompatible with other systems, they cannot use Packet Path Diversity unless they have multiple IP addresses, and so on. The network situation such as packet loss rate or bandwidth is always varying so we must modify the parameters during the streaming in order to offer more optimal and reliable transmission. Needless to say, it is obvious advantage for a system to have more versatility, that is, the system should be compatible with other systems and should apply to various circumstances. That is why we have adopted three new methods, RTP over RTP, Relay host, Forward Error Correction over each path, and proposed a new robust system which solves all the problems of previous techniques.

This system can prevent the influence of bursty packet losses very well. It also enables other systems which cannot execute Packet Path Diversity or

Forward Error Correction by themselves to utilize the methods by using this proposed system together. Furthermore, this system makes it possible for single-linked hosts which could not take advantage of Packet Path Diversity before to use the technique.

In this paper we have proposed a new method for real-time applications with which longer delay is unacceptable, and we have designed and implemented a system successfully. We have simulated our system to verify the validity, and by results of experiments we confirmed the effectiveness of our proposed system.

複数経路を活用したバーストパケットロスに強い ストリーミングシステム

目次

第1章	はじめに	1
第2章	既存手法と関連研究	3
2.1	障害の発生	3
2.2	既存の手法	4
2.2.1	バッファリングと再送	4
2.2.2	冗長符号化	4
2.2.3	Packet Path Diversity	5
2.3	既存システム	6
2.3.1	Robst(Robust Streaming Tools)	6
2.3.2	Path Diversity EMON	7
2.4	既存の手法と関連研究の問題点の整理	8
第3章	システムの提案と設計	10
3.1	提案するシステムの概要	10
3.2	RTP over RTP を利用した経路情報の取得	11
3.3	Packet Path Diversity における Relay ホストの活用	12
3.4	経路ごとのパラメータ設定	13
3.4.1	パラメータと設定変更の参考にする統計情報	14
3.4.2	パラメータの設定	15
3.5	システムの実装	16
3.5.1	システムの構成	16
3.5.2	システムの実現	18
第4章	実験	20
4.1	実験に用いた環境	20
4.2	実験結果	21
4.2.1	ランダムパケットロスに対する効果	22
4.2.2	バーストロスに対する効果	23
4.3	考察	23

第5章	おわりに	25
	謝辞	26
	参考文献	27

第1章 はじめに

近年、インターネットの普及と広帯域化により、音声や映像のようなマルチメディアコンテンツのインターネット上での利用が盛んに行われるようになってきた。これに伴い、遠隔講義システムや遠隔会議システムのように、インタラクティブな処理を行うリアルタイム・アプリケーションも増加している。このようなアプリケーションは、音声や映像を伝送するため大きな帯域を必要とする。さらに、円滑なコミュニケーションを行うためには遅延をできるだけ抑えなければならない。

しかし、現在のインターネットは Best-Effort であり、任意の end-to-end 間で QoS 制御を行う技術が発展途上にあるため、ネットワークの状況によってはパケットロスが発生してしまう。これは音声や映像の乱れや途切れといった品質の低下を引き起こす。特に、連続して多くのパケットが喪失するバーストパケットロスが発生すると、長時間映像や音声の途切れ、コミュニケーションに重大な影響が出る。このような品質の低下を抑える手法としてバッファリングと再送 (Automatic Repeat Request, ARQ[1][2])、冗長符号化 [3][4][2][5]、Packet Path Diversity[4][6] などが提案されている。バッファリングと再送は大きな遅延を許容しないシステムには効果的でない。冗長符号化は再送を伴わないため遅延への影響は少ないが、トラフィックを増加させるため、帯域が不足している場合にはパケットロスの増加を招き、さらに品質を低下させてしまう。また、バーストパケットロスに対処するためには、ブロック長を長くする、インターリーブング [2][7] と併せて使用するなどの対策を講じる必要があるが、これを行うと遅延が大きくなるため、遅延を嫌うリアルタイム型のアプリケーションには馴染みにくい。

Packet Path Diversity とは、送信者・受信者間に複数の経路が存在する場合に、その複数の経路を同時に活用する手法を指す。Packet Path Diversity は、同じデータを複数回送信する冗長配送 [6]、データを複数の経路に分割して送信する負荷分散という2つの手法と組み合わせて使用することで、さらに効果的に利用することができる。負荷分散、冗長配送のどちらを使用する場合でも、遅延が大きくなることはないため、リアルタイム型のアプリケーションに効果的である。また、複数の経路を使用するため、一部の経路で生じたバーストパケットロスや機器の障害等による、比較的長期にわたる通信の断絶の影響を抑

えることもできる。

本論文では、遅延をあまり増加させずに品質の低下を防ぐことができる冗長符号化と Packet Path Diversity の 2 つに着目し、より効率的にデータ伝送を行う手法を提案する。従来も、冗長符号化と Packet Path Diversity を組み合わせたシステムは存在した [4][8][9] が、本論文で提案する手法は、これらのシステムに存在していた以下の 3 つの問題を解決し、さらに効果的な改善をもたらすと期待される。

従来のシステムにおける 1 つ目の問題は、データの伝送が始まると、各経路への配送率や冗長符号化の冗長度などの設定を変更できないことである。ネットワークの状況は常に変化するため、効率的なデータ伝送を行うためには、状況に応じて適切に設定を変更しなければならない。2 つ目の問題は、従来のシステムでは、送信者または受信者が持つ複数の IP アドレスを利用して経路を確保するため、送信者または受信者が複数の IP アドレスを持っていない場合は Packet Path Diversity を利用できないことである。3 つ目の問題は、従来のシステムでは、設定変更の参考にする情報を得るために RTP(Realtime Transport Protocol[10]) を拡張しており、他のシステムとの互換性が失われていることである。

1 点目の問題を解決するために、ネットワークの統計情報を受信者から送信者にフィードバックし、それを参考に動的に設定を変更するという Path Diversity EMON[8][9] の手法を取り入れる。また、2 点目の問題を解決するために、Relay ホスト [4] を採用し、送信者と受信者の間の直接経路と、Relay ホストを経由した間接経路を使った Packet Path Diversity を実現する。3 点目の問題を解決するために、RTP over RTP という手法を利用して、RTP を拡張せずに設定変更の参考となる情報を得る。これにより、RTP を使用してストリーミングを行う他のシステムとの互換性が保たれるので、単体では Packet Path Diversity を利用できない Windows Media Player[11] などと本システムと併用することで、これらのシステムでも Packet Path Diversity を使用することができるようになる。さらに、本論文で提案するシステムでは、経路ごとに冗長符号化を行うことで、経路ごとに冗長符号化のパラメータを変更できるようにした。これは、複数の経路を使用する場合、経路ごとに遅延やパケットロス率のような特性が異なるため、効率よくデータ伝送を行うには、各経路の特性に合わせて経路ごとに細かく設定を変更する必要があるからである。

また、Relay ホストを使用すると、間接経路の利用が可能となるだけでなく、他の経路との共有部分の少ない経路を確保することもできるようになる。このような経路を用いることで、各経路の独立性を高め、ある箇所で発生した混雑や障害の影響を最小限に抑えることが可能となる。さらに、RTP over RTP を利用した Packet Path Diversity では、再帰的に複数回 Packet Path Diversity を繰り返すことができるため、複数の経路を持つ Relay ホスト間で、さらに Packet Path Diversity を利用して、より効率的なデータ伝送を実現することができる。

ここで、章立てについて説明する。第 2 章で既存手法と関連研究について述べる。第 3 章でシステムの提案と設計を行い、さらに適切なパラメータの設定方法について考察する。第 4 章では評価実験を行い、結果について考察する。最後に、第 5 章で本研究のまとめと結果、今後の課題について述べる。

第 2 章 既存手法と関連研究

遠隔講義システムのようなリアルタイム型のアプリケーションにおいて、パケットロスや映像や音声の途切れの原因となる。本章では、パケットロスによる影響を抑えるための手法について概観する。また、これらの手法を用いて設計された既存のリアルタイム型システムとして、Robst[12] と Path Diversity EMON[8][9] を紹介する。

2.1 障害の発生

インターネット上で映像や音声のストリームを伝送する場合、ネットワークの状況に品質が左右される。これはインターネットが Best-Effort であり、任意の end-to-end 間で帯域を確保して品質を保証する技術がまだ確立されておらず、乱れや途切れのような障害を避けられないからである。

インターネット上でのストリーミングにおいて、障害は主にパケットロスによって生じる。パケットロスは、回線が混雑している、遅延やジッターにより規定の時間内に受信者にパケットが届かないなどの理由で発生する。このようなパケットロスに対処する方法には次のようなものがある。

1. ロスがあったことを送信者に通知し、再送してもらう
2. 送信者側でパケットロスに対処する方法を用意する
 - (a) 冗長符号を付加する

(b) 冗長にパケットを送る

これらの手法について次節で説明する。

2.2 既存の手法

パケットロスの影響を抑えるために一般的に使われているバッファリングと再送 [1][2]、冗長符号化 [3][4][2][5] 及び Packet Path Diversity[4][6] を概観する。

2.2.1 バッファリングと再送

バッファリングと再送は、パケットを受信したときに受信者側でバッファを確保し、受信できなかったパケットについて、受信者が送信者に再送を要求する手法である。

これは、Windows Media Player[11] や Real Player[13] のような商用のメディアプレイヤーでも使われている手法であるが、受信者がパケットロスの発生を送信者に対して通知し、再送してもらうため、最低でも往復時間 (Round Trip Time) 程度の遅延が発生する。したがって、本研究で想定するような、遅延の制約が厳しいリアルタイム型のアプリケーションには馴染まない。

2.2.2 冗長符号化

再送を伴わずに受信者側で誤り訂正できるような符号化方式を、本論文では冗長符号化と呼ぶ。冗長符号化は、データブロックから冗長なデータを生成し、もとのデータと冗長なデータの両方を伝送することで、データブロックのビット誤りや欠損を冗長なデータを用いて復元する機能である。

この冗長符号化は、磁気媒体やコンパクトディスクの記録再生誤りなどに広く用いられており、XOR-Parity 符号 [5]、ハミング符号、Reed-Solomon 符号 [7] といった様々な手法がある。XOR-Parity 符号やハミング符号は復元効率は悪いが、復元時の計算が比較的容易である。反対に、Reed-Solomon 符号はオリジナルのデータを維持して冗長符号を付加することができ、復元効率が良い一方、復元のための計算が複雑になる。したがって、データを圧縮せずに送信したり、パケットロスが大量に発生したりする場合には、Reed-Solomon 符号では復元のための計算が再生に間に合わなくなるので、XOR-Parity 符号やハミング符号の方が復元効率がよくなる。

Reed-Solomon 符号では、同じ長さの K 個のデータに対し、 R 個の冗長なデータを生成して伝送する。 $K+R$ 個のデータのうち任意の K 個のデータを受信すれば $K+R$ 個すべてのデータを復元できる。しかし、 $R+1$ 個以上のデータが損

失ってしまうと、データの復元が不可能になる。つまり、この手法は短時間に大量のロスが発生するバーストロスに弱い。

冗長符号化を使ってバーストロスに対処するためには、冗長化のブロックの長さを長くする、冗長なデータの割合を増やす、インターリーブング [7] を行うなどの工夫が必要になる。インターリーブングとは、冗長符号化のブロックを生成した後に、複数のブロックのデータを集めて新たな処理ブロックを形成して送信することで、バーストパケットロスによって連続したデータが喪失するのを防ぐ手法である。冗長化のブロックを長くすると、そのブロック全体を受信してからでないで復号化できないので、その分だけ遅延が発生してしまう。冗長なデータの割合を増やすと送信するデータ量が増加し、より広い帯域が必要になる。インターリーブングを用いると、複数のブロックを受信してからでないで誤り訂正できないので、遅延が大きくなる。

冗長符号化は冗長なデータを伝送することで伝送量を増やすため、狭帯域が原因で十分な伝送ができずにパケットロスが発生している場合に使用すると、帯域をさらに圧迫してしまい、かえってパケットロスが増加し、品質を低下させてしまう。

このように、遅延や帯域の制約がある場合、冗長符号化で対処できるバーストロスには限界がある。

2.2.3 Packet Path Diversity

Packet Path Diversity は送信者・受信者間の複数の経路に同時にデータを送信する手法である。共有部分の少ない複数の経路が確保できる場合、この手法は狭帯域な経路やバーストロスにも有効である。

Packet Path Diversity は負荷分散と冗長配送 [2] という 2 つの手法と組み合わせることで、より効果が上がる。負荷分散とは、すべてのパケットを全経路に配送するのではなく、複数の経路に分散して配送する手法のことである。狭帯域な経路を使用する場合、負荷分散を用いると、帯域の不足に伴うパケットロスを発生させずにすむ。冗長配送とは、同じパケットを複数回冗長に送信する手法のことである。複数の経路が存在する場合、それらの経路に冗長にパケットを送信すると、一部の経路で機器の障害やバーストパケットロスによる比較的長い通信の断絶が生じたときにも、その影響を小さくすることができる。同じ割合で冗長なパケットを送信する場合、冗長符号化を用いて 1 本の経路に全てのパケットを送信するよりも、Packet Path Diversity により、複数の経路に

同じパケットを冗長配送する方が遅延が減少し、品質も向上することが確かめられている [14]。

Packet Path Diversity を利用する場合、各経路の共有部分が少なければより効果的となる。経路に共有部分が多数存在する場合、そこで混雑や障害が発生すると、障害箇所を通る全ての経路にその影響がでてしまう。したがって、経路を選択する場合には、各経路の共有部分ができるだけ少なくなるよう注意しなければならない。

2.3 既存システム

パケットロスによるストリームの品質の低下を抑えるために、冗長符号化や Packet Path Diversity を使用したリアルタイム型システムの例として、Robst[12] と Path Diversity EMON[8][9] を概観する。

2.3.1 Robst(Robust Streaming Tools)

Robst は広島大学で開発された、高品質動画をインターネット上で送受信するソフトウェアである。IPv4/v6、IP マルチキャスト等にも対応し、MPEG2-TS SD(Standard Definition) では 4Mbps-6Mbps の帯域で TV 品質並み、MPEG2-HD(High Definition) では 20-40Mbps の帯域でハイビジョン品質、DV(Digital Video) 形式のフォーマットでは 30Mbps で低遅延、高品質の動画伝送が可能である。

ネットワーク上で発生するパケットロスへの対策として前方誤り訂正符号(冗長符号化)を用いた伝送を実現しており、パケットロスによる映像/音声への影響を抑制することができる。冗長符号化には Reed-Solomon 符号を採用している。また、RTCP(Realtime Transport Control Protocol[10])を利用して送信者にネットワークの統計情報をフィードバックする機能も備えており、送信者側でネットワーク状況を監視することが可能である。図 1 に Robst システムの概略を示す。図 1 中の FEC は前方誤り訂正符号 (Forward Error Correction, FEC) の略である。

Robst では、起動時に冗長符号化の冗長度や RTCP 使用の有無を指定できる。冗長符号化の冗長度を動的に変更するように指定することもできるが、冗長度を変更するアルゴリズムの実装が未完成であり、伝送中に手動で修正することもできないため、現時点での使用は困難である。さらに、冗長符号化は伝送するパケット量を増やすため、帯域が不足している場合には使用することができ

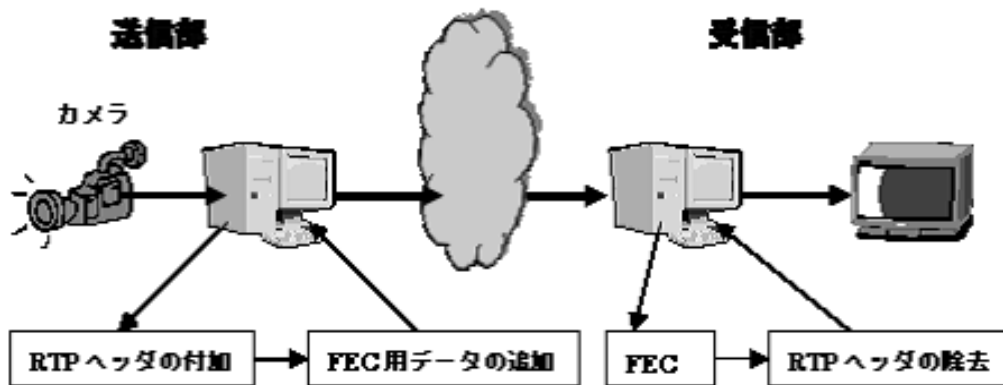


図 1: Robst システムの概略

ない。また、1本の経路しか使用しないので、経路上で機器の障害やバーストパケットロスによる比較的長時間の通信の断絶が発生した場合には、ストリームの中断を避けられない。

このように、ストリーム伝送の信頼性を向上させるために冗長符号化のみを使用する Robst が有効なのは、十分な帯域のある、ランダムパケットロス以外の問題が発生しない経路を使用する場合だけである。

2.3.2 Path Diversity EMON

Path Diversity EMON は、Robst と同様にインターネット上で動画を送受信するソフトウェアである。このシステムではパケットロスへの対策として、冗長符号化 (FEC) と Packet Path Diversity を組み合わせて用いる。冗長符号化には Reed-Solomon 符号を採用している。

Path Diversity EMON は、冗長符号化を利用することができる EMON システム [15] を拡張して、ストリーム伝送中に各経路への配送率や冗長符号化の冗長度などの設定を変更できないという従来の問題を解決するため、RTCP を使用してパケットロス率などの経路に関する統計情報を受信者からフィードバックし、それに基づいてパラメータを送信者が手動で変更する機能や Packet Path Diversity を行うための機能を加えて実現している。

このシステムでは、単方向経路についての情報を得るために RTCP を拡張して、利用可能な全ての経路の情報を送信者にフィードバックする。また、経路ごとの統計情報を得るために RTP を拡張して経路のシーケンス番号を加える、バーストロスを検出するために平均連続パケットロス数を受信者側で計算して

RTCP に入れるなどの拡張も行っている。

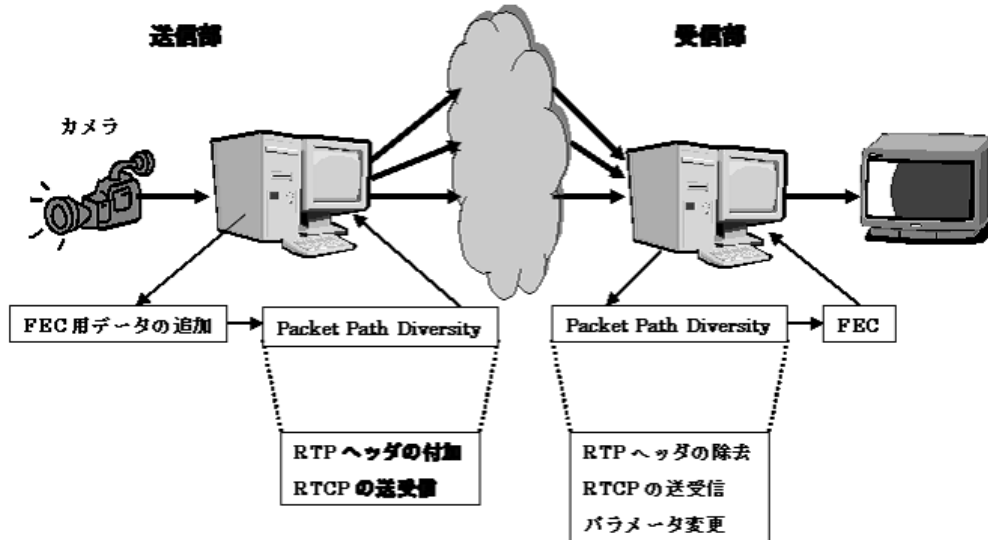


図 2: Path Diversity EMON の概略

しかし、このシステムでは、RTCP を定期的に送受信するのではなく、ユーザーがコマンドを打って指示した場合のみ行うため、パラメータ変更のタイミングや値の決定が困難である。また、送信者・受信者が持つ複数の IP アドレスを利用して経路を確保するため、送信者・受信者が共に 1 つしか IP アドレスを持っていない場合や、ルータのルーティングにより途中の経路の大半が重複してしまう場合には Packet Path Diversity を使えない。経路の重複が少ない場合でも、必ずしも最適な (最も共有部分が少ない) 経路を通るとは限らないという問題がある。さらに、複数の経路を用意した場合、経路ごとに帯域、遅延、パケットロス率などの特性が異なるにもかかわらず、経路ごとに冗長符号化を行うことができないため、経路の特性に応じた最適な設定でデータ伝送をすることができない。

2.4 既存の手法と関連研究の問題点の整理

冗長符号化は復元効率がよく、ブロック長をある程度短くすれば比較的遅延が少ないので、ランダムパケットロスには大変有効である。ただし、遅延の制約が厳しいリアルタイム型のアプリケーションで使用する場合には、バースト

パケットロスの影響を抑えることは難しい。複数の経路を利用してデータを同時に送信する Packet Path Diversity は、負荷分散と冗長配送のどちらを使用しても遅延が少なく、一部の経路でバーストパケットロスが発生しても、他の経路を用いてデータを送信することができるため、バーストパケットロスの影響を抑えることができる。そこで、遅延とパケットロスの影響をともに抑えられるこれら 2 つの手法を併用してストリームの信頼性を向上させることを考える。従来もこの 2 つを組み合わせたシステムは存在した [4] が、ストリーム伝送中に冗長符号化の冗長度や各経路への配送率などの設定を変更できないため、ネットワークの状況が変化したときには十分に対応できないという問題があった。

実環境では、帯域を他のアプリケーションが大きく消費する、障害が発生するなどの様々な要因により、ネットワークの状況は変化しうる [16]。したがって、高信頼で安定した伝送を実現するためには、ストリーム伝送中の設定変更が必要不可欠である。

2.3 節では、この問題を解決するために、ストリーム伝送中の設定変更を可能にしたシステムとして Robst と Path Diversity EMON の 2 つを挙げた。Robst にはパラメータを手動で変更できる仕組みがないため現時点での使用は困難である。また、パケットロスに対処するために使用している手法が冗長符号化だけなので、バーストパケットロスに十分対応することもできない。Path Diversity EMON は任意の時間に、任意の値にパラメータを変更できるが、次のような問題がある。

1. 送信者・受信者が 1 つしか IP アドレスを持っていない場合には Packet Path Diversity を使えない
2. 経路ごとの統計情報を得るために RTP を拡張しているので、他のシステムとの互換性がない
3. RTCP を定期的にやり取りする仕組みがないため、パラメータ変更のタイミングや値の決定が困難である
4. 冗長符号化を経路ごとに行うことができない
5. ルートを指定できないため必ずしも最適な (最も共有部分の少ない) 経路を通るとは限らない

第3章 システムの提案と設計

本章ではまず、提案する手法の概要を述べる。続いて、パラメータ決定方法の考察を行い、最後に、提案するシステムの実装について述べる。

3.1 提案するシステムの概要

ストリーミングを行うときに必ずしも十分な帯域を持つ、高信頼な回線が用意できるとは限らない。また、そのような回線が用意できたとしても、何らかの原因で突然、機器の障害やバーストロスにより通信の途切れることもある。

本論文では、十分な帯域の回線が用意できない場合や、バーストロスなどが起こる信頼度の低い回線を使用しなければならない場合でも、遅延を増加させずに品質の低下を防ぎ、安定した高信頼なリアルタイム・ストリーミングが行える手法とそれを実現したシステムを提案する。

本論文で提案する手法では、Path Diversity EMONと同様に、遅延が大きくなならない Packet Path Diversity と冗長符号化を組み合わせ、品質低下の主な原因となるパケットロスの影響を抑える。冗長符号化には復元効率の良い Reed-Solomon 符号を採用する。

複数の経路を用意する場合、経路ごとに特性が異なることがよくある。各経路の特性に合わせて最適な伝送を行うためには、パケットロスが多い経路では冗長符号化の冗長度を上げるなど、経路ごとに細かくパラメータを設定する必要がある。そこで、本論文で提案するシステムでは、各経路ごとに冗長符号化を行い、経路ごとに冗長符号化のパラメータを設定・変更できるようにする。さらに、全体での冗長符号化のパラメータと各経路への配送率も送信者側で任意に変更できるようにする。また、パラメータの設定を容易にするために、定期的に RTCP を送受信し、送信者側に統計情報を逐次表示する。これにより、RTCP を定期的に送受信できない、経路ごとに設定できるパラメータが少ないという Path Diversity EMON の問題を解決することができる。

経路ごとの統計情報を得るためには、全体のシーケンス番号と経路ごとのシーケンス番号の両方をデータとともに送信する必要がある。しかし、RTP にはシーケンス番号を1つしか入れることができないので、そのままでは経路ごとの統計情報を得ることができない。また、Path Diversity EMON のように RTP を拡張すると、他のシステムとの互換性を失い、他システムとの併用が困難にな

る。そこで、本論文で対案するシステムでは RTP over RTP という手法を用いて、拡張を行わずに 2 つのシーケンス番号をデータとともに送信できるようにする。RTP over RTP については 3.2 節で述べる。さらに、複数の IP アドレスを持っていない場合にも Packet Path Diversity を利用したり、互いに疎な経路を実現するために Relay ホストを使用する。Relay ホストについては 3.3 節で説明する。

3.2 RTP over RTP を利用した経路情報の取得

RTP over RTP とは、RTP パケットを単なるデータとみなし、もうひとつの RTP パケットのペイロードに格納して送信する手法である。

Packet Path Diversity を行うとき、経路ごとにパラメータを設定するためには経路ごとの統計情報が必要となる。このため、経路ごとのシーケンス番号を何らかの形でデータに付加して送信しなければならない。また、受信者側で正しくデータを再生するためには、全体のシーケンス番号も必要である。しかしながら、既存の RTP ヘッダに入れられるシーケンス番号は 1 つだけである。

Path Diversity EMON(PDEMON) では経路ごとのシーケンス番号を追加するために、図 3 の (b) に示すように RTP を拡張している。しかし、このような拡張を行うと、他のシステムと併用することができなくなるため望ましくない。RTCP の拡張についても同様である。

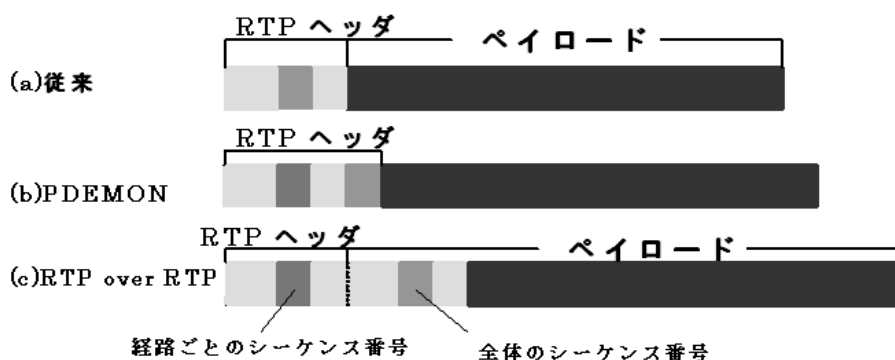


図 3: 使用する RTP パケット

そこで、本システムでは図 3 の (c) のように、全体のシーケンス番号をヘッダに持つ RTP パケットを、経路ごとのシーケンス番号をヘッダに持つ RTP パケッ

トのペイロードに入れて送信する。この手法を用いると、RTP を拡張しないため、他のシステムと併用することができる。これにより、Packet Path Diversity を行えないシステムを本システムと併用して、Packet Path Diversity の使用を可能にすることができる。さらに、RTP over RTP と Packet Path Diversity を組み合わせると、再帰的に複数回 Packet Path Diversity を使用するができる。これにより、Relay ホストで単に転送だけを行うのではなく、経路を複数持つ Relay ホスト間でさらに Packet Path Diversity を実行することも可能となる。

3.3 Packet Path Diversity における Relay ホストの活用

Relay ホストとは、RTP パケットを受け取り、それを転送するホストのことである。送信者が Relay ホストに RTP パケットを送信すると、Relay ホストは、ユーザが設定した送信先にそれを転送する。

Path Diversity EMON では送信ホストまたは受信ホストが複数の IP アドレスを持っている場合でなければ Packet Path Diversity を利用することができなかつた。また、確保した経路の重複を防ぐための手法がなく、経路の大半が重複しているような場合には、Packet Path Diversity の効果がほとんど得られないという問題があった。本節では、Relay ホストを用いてこれらの問題を解決する。さらに、RTP over RTP を用いた Packet Path Diversity と Relay ホストを組み合わせることで、より効果的にデータ伝送を行う方法について述べる。

図 4 は送信者・受信者が 1 つしか IP アドレスを持っていない場合の例である。送信者が Relay ホストにデータを送信することにより、送信者・受信者間の直接の経路とは異なる、Relay ホストが仲介する経路を実現することができる。このように Relay ホストを使用することで、従来の手法では複数の経路を持つことができなかったホスト間でも Packet Path Diversity を利用することができるようになる。このため、図 4 のように、送受信者間の直接の経路上で何らかの障害が発生した場合でも、Relay ホストを用いた間接経路が直接経路と重複していなければ、その障害の影響を抑えることができる。

図 5 は送信ホストと受信ホストが IP アドレスを複数持っている場合の例である。Relay ホストを使用しなければ、2 本の経路は図のバツ印のところで重複することになる。このとき、重複箇所が障害が発生すると、Packet Path Diversity を利用してもその影響を避けることはできない。しかしながら、図 5 のように適切な Relay ホストを選択して使用すると、経路の重複を避けて、障害の影響

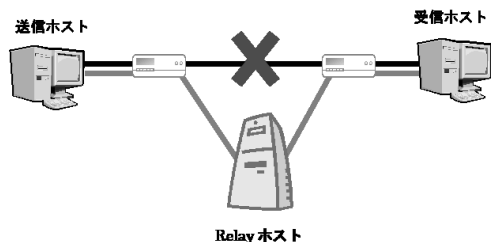


図 4: 経路が 1 本の場合

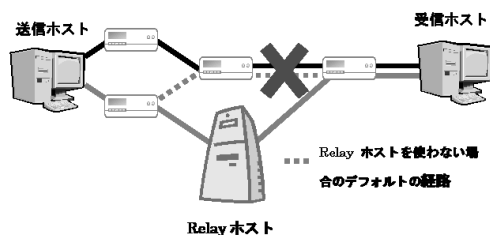


図 5: 経路が複数ある場合

を減らすことができる。Relay ホストをインターネット上に複数置く場合は、これらを複数組み合わせることで、経路を多数構築することができる。これらの経路の中から共有部分(重複)の少ない経路の組を選択することで、より効果的に Packet Path Diversity を利用できる。文献 [4] には traceroute[17] を利用して互いに疎な経路を選択するアルゴリズムが提案されている。

図 6 は送信・受信ホスト間に加えて、Relay ホスト間でも Packet Path Diversity を使用する場合の例である。3.2 節でも述べたように、RTP over RTP を用いて Packet Path Diversity を実現することにより、再帰的に複数回 Packet Path Diversity を利用することができるので、Relay ホスト間での Packet Path Diversity も可能となる。図 6 の例において、Relay ホスト間で Packet Path Diversity を行えない場合、経路 2 では最大 5Mbps の帯域しか使えないため、全体では最大 $10+5=15$ Mbps の帯域になる。しかし、Relay ホスト間でも Packet Path Diversity を使うと、経路 2 の帯域は最大 $3+5=8$ Mbps となり、全体で最大 $10+3+5=18$ Mbps の帯域を使用することができるようになる。このように、Relay ホスト間にも複数の経路がある場合、再帰的に Packet Path Diversity を使うことにより、さらに効率的なデータ伝送を実現することができるようになる。

3.4 経路ごとのパラメータ設定

本節では、ストリームを伝送する際のパラメータの設定について考察する。ネットワークの状況に応じてパラメータを変更することにより、受信者に映像の乱れや途切れをできる限り見せないようにすることと、必要以上に大量のデータを送信することによる帯域の浪費を防ぐことが本節の目的である。

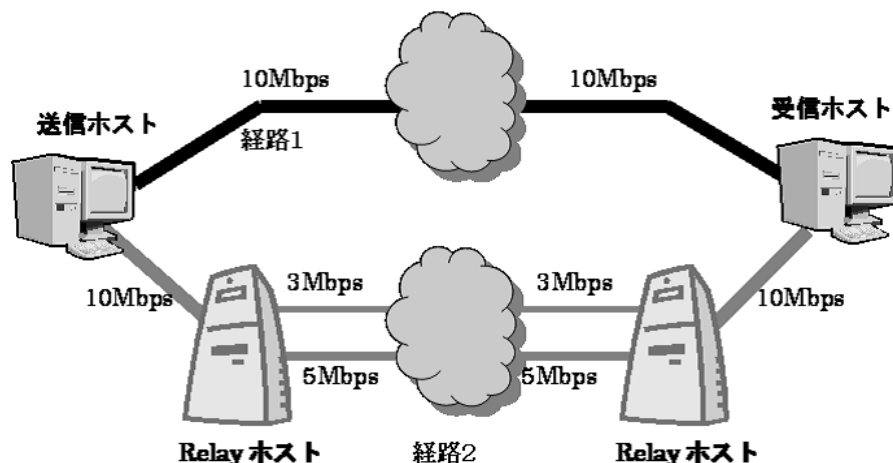


図 6: Relay ホスト間での Packet Path Diversity

3.4.1 パラメータと設定変更の参考にする統計情報

本論文で提案するシステムではパケットロスの影響を抑えるために冗長符号化と Packet Path Diversity を使用している。また、経路ごとに細かくパラメータ設定を行うために、全体の冗長符号化だけでなく、経路ごとの冗長符号化も可能にしている。冗長符号化がもつパラメータは、冗長化ブロックの最大長、実際の冗長化ブロック長、1 ブロックあたりに含まれる冗長なパケット数の3つである。Packet Path Diversity が持つパラメータは、各経路への配送率である。したがって、ユーザがストリーム伝送中に変更できるパラメータは、全体の冗長符号化パラメータ、経路ごとの冗長符号化パラメータ、各経路への配送率である。

ネットワークの状況は常に変化するため、ストリーム伝送中に最適な経路の組み合わせが変わることもある。このため、できるだけ最適な伝送を行うためには、ストリーム伝送中に新たな経路を追加・削除できるようにする必要もある。

本システムでは、汎用性を高めるため、メディアストリームを伝送する場合に一般的に使われている RTP と RTCP を利用して統計情報の収集及び送受信を行う。RTP は、Realtime Transport Protocol をあらわし、インターネット上でのメディアストリーム伝送における標準規約となっている。RTCP は RTP を用いたデータ伝送を制御するために使用され、データ配信の品質についてアプリケーションに情報を提供する。送信者が受信者に送信する RTCP パケットは送信者レポート (Sender's Report)、受信者が送信者に送信する RTCP パケットは受信者レポート (Receiver's Report) と呼ばれ、RTP とともに RFC1889[10]

で規定されている。

RTCP を用いて得られる情報は、パケットロス率・累積損失パケット数・パケット間隔ジッタ・最大受信シーケンス番号・最新送信レポート時刻・送信レポート経過時間である。これらの項目には次のような値がセットされている。

パケットロス率 最後に受信状況を報告する RTCP メッセージ (受信者レポート) を送信してから現時点までに、受信できなかったパケットの割合

累積喪失パケット数 データの受信を開始してから現時点までに受信できなかったパケットの総数

パケット間隔ジッタ 連続して送信されたパケットの間隔の、送信時と受信時の平均偏差

最大受信シーケンス番号 これまでに受信したパケットの中で、最大のシーケンス番号

最新送信レポート時間 送信者からの最新の送信レポートを受信した時刻

送信レポート経過時間 送信者からの最新の送信レポートを受信した時刻から、現在の時刻までの経過時間

Path Diversity EMON では、冗長符号化を適用するかどうかを決定するために、上記の情報に加えて平均連続パケットロス数を使っている。平均連続パケットロス数は連続して起こったパケットロスの数を表し、これが大きいときバーストパケットロスが起きていることになる。バーストパケットロスが起きている場合、冗長符号化はあまり効果がない。

RTCP を用いて平均連続パケットロス数を得るためには RTCP を拡張しなければならない、これは他のシステムとの互換性を考慮した場合に望ましくない。このため、本システムではバーストパケットロスの有無を判断するための情報として平均連続パケットロス数は用いずに、パケットロス率からバーストパケットロスの有無を推測する。つまり、パケットロス率の増減が激しい場合に、パケットロス率の高いところではまとまったパケットロスが発生していると推測できるので、バーストパケットロスが起きていると判断する。

3.4.2 パラメータの設定

ネットワークの状況は、帯域に余裕があるときと、ないときの2つに大きく分けることができる。帯域そのものは統計情報として得られないので、オペレータが元から持っている知識を利用するか、Pathload[18]、Iperf[19] といった手法を利用して得る。データ伝送開始後は、配送率とパケットロス率から推測する。

すなわち、ある経路の配送率を大きくしてもパケットロス率が増加しない場合は帯域に余裕があり、パケットロス率が増加する場合は帯域に余裕がないと推測する。遅延も統計情報から直接得ることができないので、帯域と同様に、オペレータの知識か、最大受信シーケンス番号・最新送信レポート時間・送信レポート経過時間から推測する。すなわち、受信者レポートが送信された時刻と現在の時刻、最大受信シーケンス番号と最大送信シーケンス番号(最も最近送信したパケットのシーケンス番号)を比較し、受信者レポートの送信時刻と現在の時刻の差が大きくないにもかかわらず、最大受信シーケンス番号と最大送信シーケンス番号の差が大きい場合には遅延が大きいと推測する。

全ての経路の帯域に余裕がないときは負荷分散を行う。個々の経路の帯域に余裕はないが、全ての経路の帯域を足すとデータ伝送に最低限必要な帯域を上回るような場合は、冗長配送や冗長符号化と併用すると効果的である。2章で述べたように、冗長符号化は喪失したパケットを復元するのに時間がかかる、パケットロスが多いと復元できないという欠点があるので、パケットロスが多い経路や遅延・ジッタが大きい経路に対しては冗長符号化ではなく冗長配送を用いるべきである。また、経路ごとに遅延やパケットロス率が大幅に違うような場合には、全体の冗長符号化ではなく、冗長配送と経路ごとの冗長符号化を組み合わせ、経路の特性に応じたパラメータ設定をしなければならない。

帯域に余裕がある場合は冗長符号化と冗長配送を組み合わせる。この場合も上と同様に、全ての経路の特性がほぼ同じで、遅延やジッタ、パケットロス率が小さい場合は全体の冗長符号化が有効である。経路ごとに特性が大きく異なる場合は、遅延やパケットロス率の大きい経路には冗長配送を、そうでない経路には経路ごとの冗長符号化を適用するのが良い。

3.5 システムの実装

3.5.1 システムの構成

本研究で提案するシステムは、EMONシステム[15]を拡張し、RTCPの送受信、RTP over RTP、Packet Path Diversity、Relayホスト、パラメータの動的変更に対応した機能を付け加えて実現した。

EMONシステムとは、単機能に分割されたプロセスを組み合わせることで映像・音声などのマルチメディア・データのストリーミングを行うシステムである。図7は映像伝送の例であり、映像キャプチャ部、送信部、受信部、映像再

生部の4つのプロセスをパイプラインでつなげることで映像伝送を実現している。冗長符号化を行う場合は、映像キャプチャ部と送信部の間に冗長符号化プロセスを、受信部と映像再生部の間に冗長復号化プロセスを挟むことで実現できる。

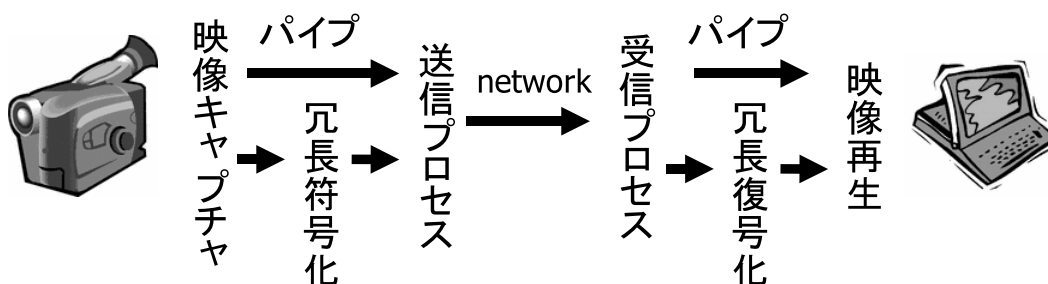


図 7: EMON システム

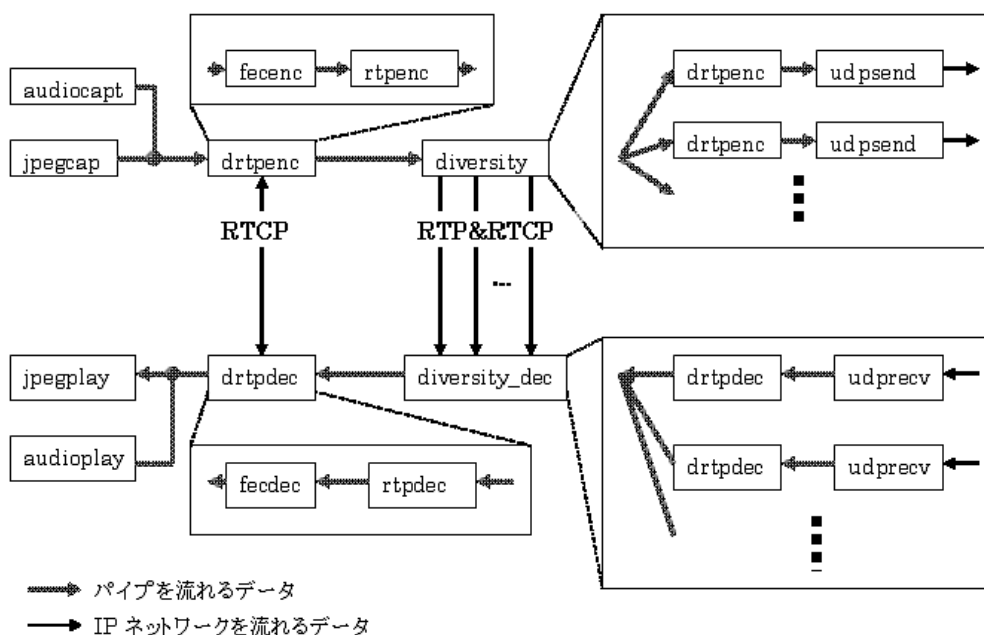


図 8: プログラムの接続関係

本研究で主に使う EMON のプログラムと、新たに追加したプログラムの機能を表 1 に、プログラムの接続関係を図 8 に示す。

表 1: プログラムの機能

プログラム名	機能
fecdec	FEC(Forward Error Correction:前方誤り訂正)を行う
fecenc	FEC用の冗長なデータを追加する
udprecv	マルチキャスト及びユニキャストのUDPデータグラムを受信する
udpsend	マルチキャスト及びユニキャストのUDPデータグラムを送信する
drtpenc	FEC用の冗長なデータの追加、RTPヘッダの追加、RTCPの送受信を行う
drtpdec	FEC、RTPヘッダの除去、RTCPの送受信を行う
diversity	経路ごとにFEC用データとRTPヘッダを追加し、データを複数の経路に送信する。経路ごとにRTCPを送受信する。
diversity_dec	複数の経路からデータを受信し、経路ごとにFECとRTPヘッダの除去を行う。経路ごとにRTCPを送受信する。

3.5.2 システムの実現

本研究では図9のようなシステムを想定して実装を行った。

EMONシステムにはjpegcapt、jpegplayという映像データの取り込み、再生を行うプログラムが用意されているが、使用できる形式がJPEGに限られている。そこで、本システムでは映像の取り込みや再生には、使用する形式にあわせて専用のアプリケーションを用いることとし、伝送部分のみの実装を行った。

まず、送信側プロセスについて述べる。送信ホスト(ホストB)は、映像取り込み用ホストAから送信されたデータを受け取り、全体のシーケンス番号を持つRTPヘッダをデータに付加する。このとき、全体の冗長符号化のための冗長データも追加する。その後、経路ごとの配送率(r)に従ってデータを分割または複製し、経路ごとの冗長符号化のための冗長データを追加する。最後に、各データを経路ごとのシーケンス番号を持つRTPのパayloadに格納して各経路に配送する。

ここで、配送率とは送信するデータ数の元のデータ数に対する割合、または、

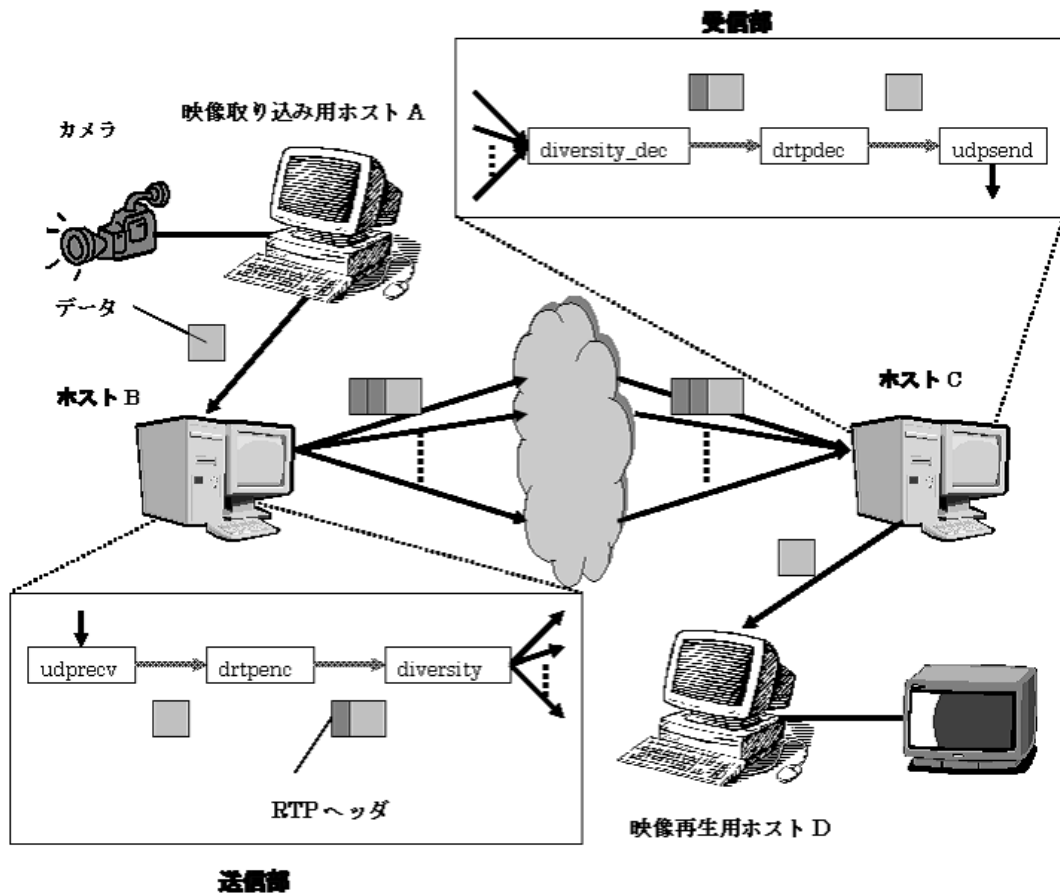


図9: 想定するシステム

あるデータをその経路に送信する確率を意味する。経路 i の配送率を r_i とするとき、パケットを送信する経路と回数を決定するアルゴリズムは次のものを使用した。

1. $r_i = 0$ の場合
経路 i にデータを送信しない
2. $r_i = n_i + p_i$ ($n_i \in \mathbf{N}$, $0 \leq p_i < 1$) の場合
 - (a) 経路 i に n_i 回同じパケットを送信する
 - (b) $P = \sum_i p_i = n + p$ ($n \in \mathbf{N}$, $0 \leq p < 1$) とする。
 - i. 確率 p に基づいて、パケットを 1 回送信するかどうか決める。
 - ii. 送信する場合は n に 1 を足す。
 - iii. p_1, p_2, \dots, p_m の割合で経路 $1, 2, \dots, m$ に n 個のパケットを振り分ける。

上記のアルゴリズムの 2 番において、 $n_i = 0$ のとき、各経路で独立に確率 p_i で

パケットを送信すると、各経路の配送率の和 $\sum p_i$ が 1 以上の場合でも、 $\prod_i(1-p_i)$ の確率でどの経路にもパケットを送信しないことになってしまう。しかし、上記のアルゴリズムを使用すると、各経路の配送率の和が 1 以上になる場合には、必ずどこかの経路に少なくとも 1 回パケットを送信することができる。

次に、受信側プロセスについて述べる。まず、受信ホスト (ホスト C) は、各経路ごとに、受け取ったデータから経路ごとのシーケンス番号を持つ RTP ヘッダを除去し、冗長符号化による損失パケットの復元を行う。その後各経路のデータをまとめて、全体のシーケンス番号を持つ RTP ヘッダを除去した後、全体の冗長符号化による損失パケットの復元を行う。最後に、映像再生用ホスト D にデータを送信する。

Relay ホストを用いる場合は、送信ホストの最後のプロセスのデータ送信先を受信ホストではなく Relay ホストにすればよい。Relay ホストとなるホスト E では、データを待ち受ける IP・ポートと転送先の IP・ポートの組み合わせを指定して、データを Relay するためのプロセスを立ち上げておく。文献 [4] では、複数の経路の Relay を行う場合、それぞれの経路ごとにプロセスを立ち上げるのではなく、送信元と転送先の情報を持ったテーブルを用意して、1つのプロセス内でその組み合わせに応じて全てのデータを転送するという手法を用いている。しかし、本研究では複数の経路を持つ Relay ホスト間でも Packet Path Diversity を行えるようにするため、経路ごとにプロセスを立ち上げるという手法を選んだ。

第 4 章 実験

本システムを使用することによりパケットロスによる影響が減少し、また、バーストパケットロスにも強くなっていることを確かめるために実験を行い、その結果を評価する。

4.1 実験に用いた環境

実験環境は映像取り込み用マシン 1 台と送信用マシン 1 台、受信用マシン 1 台、映像再生用マシン 1 台、帯域制御用マシン 2 台、Relay ホスト用マシン 1 台の計 7 台を用いて図 10 のように構築した。

映像取り込み用マシンと映像再生用マシンでは DVcommXP[20][21] を用いて

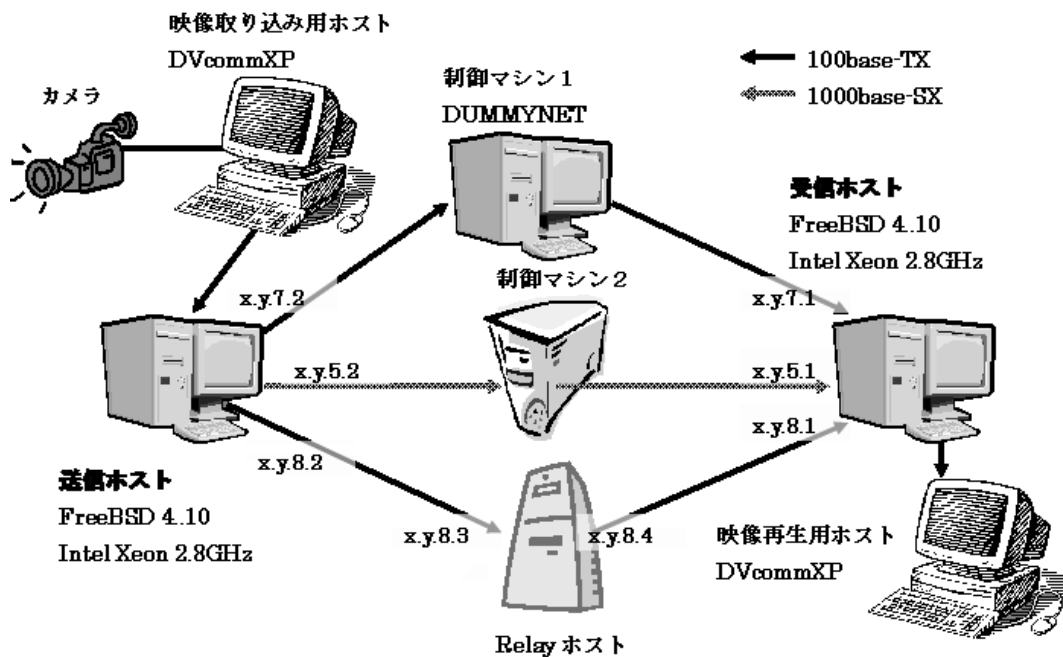


図 10: 実験環境

映像の送受信及び表示を行った。DVcommXP とは、WindowsXP 上で動作する、DV(デジタルビデオ)規格の高画質映像とステレオ音声をネットワーク経由で伝送するために開発されたソフトウェアである。送信用・受信マシンは Intel Xeon 2.8GHz 搭載で、OS には FreeBSD4.10 を用いた。また、帯域制御のために、DUMMYNET[22] を用いた帯域制御用マシン (以後、制御マシン 1) と、Intel IXP2400 Network Processor 搭載の帯域制御用マシン [23](制御マシン 2) を用いた。制御マシンは帯域制御のほかに、遅延やジッタ、ランダムパケットロスなどを発生させることができる。さらに、制御マシン 2 では、バーストパケットロスを発生させることもできる。実験では、この 2 つの制御マシンによりインターネットをエミュレートする。

4.2 実験結果

ランダムパケットロスが発生する経路を用いた場合と、バーストパケットロスが発生する経路を用いた場合のそれぞれについて、パケットロス率を変化させたとき、次の 4 通りの送信方法において、再生できなかった (受信したが、再生に間に合わなかったものを含む) パケットの割合を調べた。

- 1 本の経路で冗長符号化を行って送信する

- 1本の経路で冗長配送を行う
- 2本の経路で冗長符号化を行って送信する
- 2本の経路で冗長配送を行う

条件を等しくするために、冗長符号化と冗長配送の冗長度はともに2になるように設定した。すなわち、全体で2倍の packets を送信する。冗長符号化のブロック数は15、1ブロックあたりの冗長なデータ数は8とした。経路が2本の場合の冗長符号化は、それぞれの経路に半分ずつ packets を分け、冗長符号化により packets 数が2倍になるようにしている。また、冗長符号化には Reed-Solomon 符号を用いた。

4.2.1 ランダムパケットロスに対する効果

ランダムパケットロスの発生するネットワークにおいて、本システムが有効かどうかを確認するための実験を行った。パケットロスはそれぞれの経路で独立に、同じ割合で発生させた。

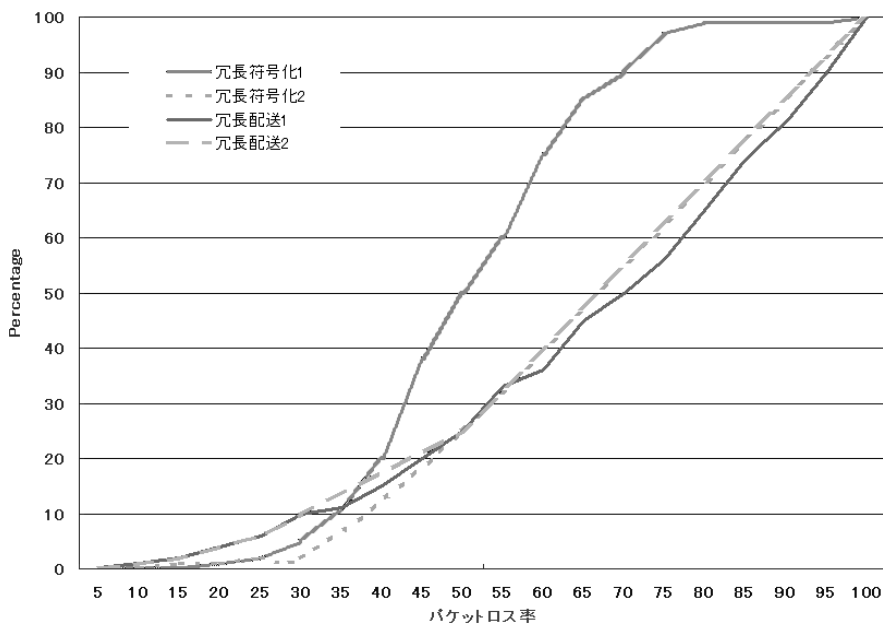


図 11: ランダムパケットロスに対する効果

図 11 において、縦軸は再生までに受信できなかった packets (復元が間に合わなかった packets を含む) の割合を、横軸は経路の packets ロス率を示している。以後、再生までに受信できなかった packets の割合を回復不能 packets

ロス率と呼ぶ。ここで、冗長符号化 1 が 1 本の経路に冗長符号化を適用して送信した結果、冗長符号化 2 が 2 本の経路に冗長符号化を適用して送信した結果、冗長配送 1 が 1 本の経路に冗長配送をした結果、冗長配送 2 が 2 本の経路に冗長配送をした結果のグラフである。

図 11 より、パケットロス率が 35%程度までなら 2 本の経路を用いた冗長符号化の回復不能パケットロス率が最も小さく、1 本の経路を用いた冗長符号化、冗長配送の順になることがわかる。しかし、パケットロス率が 30%をこえたあたりから、冗長符号化の回復不能パケットロス率が急激に増加し、35%程度で 1 本の経路を用いた冗長符号化の回復不能パケットロス率が最も大きくなる。さらに、パケットロス率が 50%程度になると、2 本の経路を用いた冗長符号化と冗長配送の回復不能パケットロス率がほぼ等しくなる。

4.2.2 バーストロスに対する効果

バーストパケットロスが発生するネットワークにおいても、本システムが有効であることを確認するための実験を行った。DUMMYNET を使用して帯域制御を行う制御マシン 1 ではバーストパケットロスを発生させることができない。そこで、制御マシン 1 ではランダムパケットロスを、制御マシン 2 ではバーストパケットロスを発生させた。ただし、パケットロス率は両方の経路で等しくなるように設定した。

図 12 において、縦軸は回復不能パケットロス率を、横軸はパケットロス率を表している。ただし、パケットロス率が $P\%$ のとき、バーストパケットロスを発生させる制御マシン 2 では 100 パケットあたり、連続して P 個のパケットが喪失するようになっている。各グラフの意味は 4.2.1 節と同じである。

図 12 より、2 本の経路を用いた冗長配送、2 本の経路を用いた冗長符号化、1 本の経路を用いた冗長符号化と冗長配送の順に回復不能パケットロス率が大きくなることがわかる。また、1 本の経路を用いた場合は、パケットロス率と回復不能パケットロス率がほぼ比例する。

4.3 考察

ランダムパケットロスが発生する経路において、パケットロス率が 30%をこえると回復不能パケットロス率が急激に増加するのは、パケットロスが多すぎて回復のための計算が再生に間に合わなくなったためであると考えられる。1 本の経路を用いた場合と 2 本の経路を用いた場合の両方で回復不能パケットロ

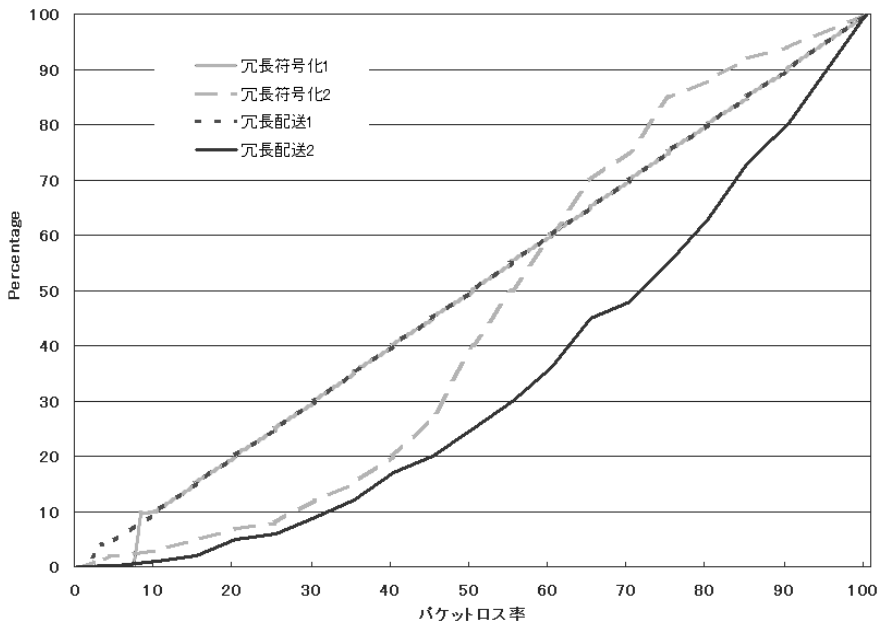


図 12: バーストパケットロスに対する効果

ス率が増加していることから、この推測が正しいことを確認できる。また、1本の経路を用いて冗長符号化を行った場合よりも2本の経路を用いた冗長符号化の方が回復不能パケットロス率が小さい理由は、2本の経路を使用した場合、各経路ごとに冗長符号化を行うので、回復できなかったパケットを双方の経路で補い合うためであると思われる。また、1本の経路を用いた冗長配送と2本の経路を用いた冗長配送の回復不能パケットロス率がほぼ等しいことから、冗長配送における回復不能パケットロス率は全経路の冗長度の和に比例することがわかる。

バーストパケットロスが発生する場合、1本しか経路を使用しなければ、冗長符号化と冗長配送のどちらを用いてもそのロスを回復することはできない。しかし、この実験では、バーストパケットロスが発生しないときにはパケットロスが0になるため、パケットロス率と回復不能パケットロス率がほぼ比例する。

バーストパケットロスが発生する経路と、ランダムパケットロスが発生する経路の両方を使用した場合、バーストパケットロスの発生箇所でのロスを回復することができないのは1本の経路を用いた場合と同様であるが、もう1本の経路を用いて伝送されたパケットにより、バーストロスの箇所をある程度補うことができる。このため、1本の経路を用いた場合よりも2本の経路を用い

た場合の方が回復不能パケットロス率が減少する。しかし、もう1本の経路ではランダムパケットロスが発生しているため、パケットロス率が高くなると冗長符号化による回復が再生に間に合わなくなり、受信できているパケットも再生に使用できなくなる。このため、パケットロス率が高くなると、2本の経路で冗長符号化を用いた場合の回復不能パケットロス率が大幅に増加する。

このように、ランダムパケットロスが起こる経路を用いた場合、バーストパケットロスが起こる経路を用いた場合のどちらでも、2本の経路を使って Packet Path Diversity をする方が、1本の経路のみでデータを送信するよりも回復不能パケットロス率が小さくなった。特に、バーストパケットロスが発生する場合には、図 12 から、Packet Path Diversity の効果は明らかである。

以上の実験から、バーストパケットロスが発生する経路やパケットロス率が高い経路を用いる場合には冗長配送を行い、ランダムパケットロスのみが少し発生する経路を使用する場合は経路ごとの冗長符号化を使用すればよいことがわかる。今回の実験では、バーストパケットロスの起こる経路の組み合わせや、バーストパケットロスとランダムパケットロスが共に発生する経路、帯域に余裕がない経路などの検証を行うことはできなかった。しかし、これらの経路でも Packet Path Diversity が有効であることが予想できる。また、冗長度や冗長符号化のパラメータ、配送率を変更して実験を行い、最適なパラメータを求めれば、より効率的なデータ伝送を行うことができるようになると思われる。

第5章 おわりに

遠隔講義システムやビデオ会議システムのような遅延の制約が厳しいリアルタイム型のアプリケーションにおいては、遅延を増加させずにパケットロスの影響を抑えてストリームを伝送する手法が必要となる。そこで、本論文では、遅延を増加させずに品質の低下を防ぐことができる冗長符号化と Packet Path Diversity の2つに着目し、遅延の増加を伴わずに効果的にストリームの品質を改善できる手法を提案した。

この2つを利用した研究は従来から存在したが、従来のシステムでは、IP アドレスを複数持っていないと Packet Path Diversity を実現できない、Packet Path Diversity を実現するためにプロトコルが拡張されており、他のシステムとの互換性が保たれていないなどの問題があった。これらの問題を解決するた

めに、本論文で提案するシステムでは RTP over RTP と Relay ホストを新たに導入した。また、この2つを使用することにより、共有部分の少ない経路を確保したり、複数の経路を持つ Relay ホスト間でも Packet Path Diversity を利用して、さらに効率の良いデータ伝送を実現することができるようになった。

このシステムの有効性を検証するための実験では、DUMMYNET と帯域制御マシンを用いてインターネットをエミュレートし、ランダムパケットロスが起る場合と、バーストパケットロスが起る場合について検証した。この結果から、Packet Path Diversity を使用することにより、パケットロスによる品質の低下を効率的に防げることを実証した。

今後の課題としては、今回の実験で試さなかった遅延のある経路やジッターのある経路でも本論文で提案した手法が有効であることを確かめること、実環境で実験を行い、本システムが実際の使用に耐え得るものであることを確かめること、今回使用しなかった他の手法を組み合わせることなどがある。実環境下での実験を行うことによって、シミュレーションでは示せなかった、経路間のネットワーク状況の相関による影響などを考察することができる。

本システムに組み合わせることができる手法としては、冗長符号化における他のアルゴリズム (パリティ符号など) やインターリーブリングなどが考えられる。パリティ符号のような比較的単純な誤り訂正符号を用いることで、データ量の多い高品質な映像伝送にも対応できるようになると思われる。また、インターリーブリングを使うことで、よりバーストパケットロスに強いシステムの構築が可能となる。さらに、物理的に経路を複数用意する、Relay ホストを使用するという Packet Path Diversity の実現方法以外に、マルチキャストを利用する方法も考えることができる。この方法を使用してさらに多くの経路を確保し、共通部分の少ない経路を選択することにより、ストリームの品質低下をさらに効果的に防ぐことができるようになる。

今後は、上記のような手法も取り入れてシステムの完成度を高め、広く実用されるようなシステムにしたい。

謝辞

本研究を行うにあたって、多大なるご教示を賜りました美濃導彦教授に深く感謝の意を表します。本研究を進めるにあたり、あらゆる面でご指導・ご鞭撻

いただきました岡部寿男教授、中村素典助教授に心より感謝の意を表します。

また、研究全般において支えてくださった美濃研究室の皆様と、岡部研究室の皆様、特に私の心のケアをしてくださった大平さん、黄さん、勇者まるしんに感謝いたします。

参考文献

- [1] Vitsas, V. and Boucouvalas, A.: Effectiveness of Selective Reject(SREJ) Automatic Repeat Request(ARQ) scheme with RR-coding in Infrared Wireless LANs, *International Symposium on CSNDSP* (2002).
- [2] Perloms, C. and Hodson, O.: Options for Repair of Streaming Media, RFC2354 (1998).
- [3] L.Rizzo: Effective Erasure Codes for Reliable Computer communication Protocols, *ACM Computer Communication Review*, Vol. 27, No. 2 (1997).
- [4] Nguyen, T. and Zakhor, A.: Path Diversity with Forward Error Correction(PDF) System for Packet Switched Networks, *IEEE INFOCOM2003 -The Conference on Computer Communications*, Vol. 22, No. 1 (2003).
- [5] Li, A.: An RTP Payload Format for Generic FEC, IETF Draft (2004). <http://www.ietf.org/internet-drafts/draft-ietf-avt-ulp-10.txt>.
- [6] J.Liang, Y., G.Steinbach, E. and Girod, B.: Multi-Stream Video over IP Using Packet Path Diversity, *ACM Multimedia* (2001).
- [7] G.Liebl, M.Wagner, J.pandel and W.weng: An RTP Payload Format of Erasure-Resilient Transmission of Progressive Multimedia Streams, IETF Draft (2004). <http://www.ietf.org/internet-drafts/draft-ietf-avt-uxp-07.txt>.
- [8] 川西智也: パケットパスダイバーシティを利用した高信頼のメディアストリーミング, 修士論文, 京都大学大学院情報学研究科 (2003).
- [9] 川西智也, 中村素典, 岡部寿男, 美濃導彦: FEC と Path-Diversity を利用した回復可能なストリーミング, 情報処理学会研究報告 2003-DPS 116-014 (2004).
- [10] H.Schulzrinne, S.casner, R.Frederick and V.Jacobson: RTP: A Transport Protocol for Real-Time Applications, RFC1889 (1996).

- [11] Microsoft: Windows Media Player. <http://www.microsoft.com/japan/windows/windowsmedia/>.
- [12] 相原玲二, 大塚玉記, 近藤徹, 西村浩二, 前田香織: MPEG2 over IPv6 システムの開発と皆既日食中継実験 (2002). 広島大学情報メディアセンター.
- [13] RealNetworks: RealPlayer. <http://www.real.com>.
- [14] S.Mao, Y.Wang, S.Lin and S.Panwar: Video transport over ad hoc networks with path diversity, *ACM Mobile Computing and Communication Review*, Vol. 7, No. 1 (2002).
- [15] 笠松健一, 古村隆明, 藤川賢治, 岡部寿男, 池田克夫: コマンドパイプラインによるマルチメディアストリーム処理, 第10回マルチメディア通信と分散処理 (DPS) ワークショップ (2002).
- [16] 藤本康平, 阿多信吾, 村田正幸: インターネットにおける実測に基づいたパケット転送遅延の統計的分析, 電子情報通信学会技術報告研究 (IN2000-48) (2000).
- [17] TRaceRT: traceroute. <http://www.tracert.com>.
- [18] Jain, M. and Dovrolis, C.: Pathload: a measurement tool for end-to-end available bandwidth, *the 3rd Passive and Active Measurements Workshop* (2002).
- [19] Gates, M., Tirumala, A., Dugan, J. and Gibbs, K.: Iperf. <http://dast.nlanr.net/projects/Iperf/>.
- [20] ファットウェア株式会社: DVcommXP. <http://www.fatware.jp/>.
- [21] Kobayashi, K., Ogawa, A., Casner, S. and Bormann, C.: RTP Payload Format for DV (IEC 61834) Video, RFC3189 (2002).
- [22] Rizzo, L.: DUMMYNET. http://info.iet.unipi.it/luigi/ip_dummynet/.
- [23] 小林克志: トランスポートの検証を目的としたギガビットネットワークエミュレーションシステムの実装, 第15回インターネット技術第163委員会研究会 (2004). <http://www.itrc.net/report/meet15/data/12b/kobayashi.pdf>.