

## PR-SCTP を用いた高信頼性ストリーミング伝送

前田 朋孝<sup>†</sup> 小塚 真啓<sup>‡</sup> 岡部 寿男<sup>††</sup>

<sup>† ‡ ††</sup> 京都大学 〒606-8501 京都市左京区吉田本町

E-mail: <sup>†</sup> tomo@net.ist.i.kyoto-u.ac.jp, <sup>‡</sup> ma-kun@kozuka.jp, <sup>††</sup> okabe@i.kyoto-u.ac.jp

**あらまし** 現状のリアルタイム映像伝送では通常 RTP/UDP が用いられる。RTP においてはパケットロスが起こった場合の回復のために FEC を用いることが考えられているが、バースト的なパケットロスに対する耐性には限界がある。そこで、本研究では RTP と PR-SCTP を組み合わせることにより、バースト的なパケットロスに対する耐性を持ちつつリアルタイムストリーム伝送を行うことを提案する。バッファリングを行いつつ PR-SCTP によりあらかじめ指定した遅延制約の範囲内で再送を行うことで、遅延と引き換えにバースト的なパケットロスに対する耐性を高めることができる。

RTP/UDP で実装されている既存のアプリケーションとの互換性を考慮して提案方式を RTP の中継器として実装し、Robst を用いて評価を行った。

**キーワード** PR-SCTP, リアルタイム伝送, 中継器

## Reliable Streaming Transmission Using PR-SCTP

Tomotaka MAEDA<sup>†</sup> Masahiro KOZUKA<sup>‡</sup> and Yasuo OKABE<sup>††</sup>

<sup>† ‡ ††</sup> Kyoto University Yoshida-honmachi, Sakyo-ku, Kyoto-shi, Kyoto, 606-8501 Japan

E-mail: <sup>†</sup> tomo@net.ist.i.kyoto-u.ac.jp, <sup>‡</sup> ma-kun@kozuka.jp, <sup>††</sup> okabe@i.kyoto-u.ac.jp

**Abstract** Combination of RTP and UDP is commonly used in real-time video streaming, today. In fear of packet loss, FEC may be used to recover the lost packet data in RTP. But there is a limit in the tolerance to a burst packet loss. We propose to transmit real-time video streams that have tolerance to a burst packet loss by combining RTP with PR-SCTP. By retransmission within the range of the delay restriction specified by PR-SCTP while doing buffering, the tolerance to a burst packet loss can be improved in exchange for some delay.

We have implemented the proposed method as a translator of RTP in consideration of interoperability with the existing application implemented with RTP/UDP

**Keyword** PR-SCTP, Real-time Video Streaming, Translator

### 1. はじめに

現在の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。なかでも TV 会議や遠隔講義などの映像や音声をリアルタイムで伝送するアプリケーションが開発され、それを利用する機会が急激に増加している。しかし、映像・音声をリアルタイムで伝送する通信が増大すると、ネットワークに流れるトラフィックが増加しパケットロスなどの障害が頻繁に起こるようになり映像や音声の乱れの原因となる。従って、このような障害の影響を抑え、アプリケーションに提供することが必要となる。

Robst[1]や DVTS (digital video transport system) [2]などに代表される既存のリアルタイム映像伝送システムではトランスポート層のプロトコルとして RTP(RFC350)[3]が用いられる。パケットのロスへの対策としては前方誤り訂正符号 (FEC) による信頼性

向上のための機能 (RFC2733)[4]が利用できる。FEC では再送を伴わずに冗長なデータを送ることで受信者側において誤り訂正ができ遅延をあまり増加させないが、ある程度以上連続した複数パケットが損失するバーストパケットロスが起こった場合、誤り訂正ができなくなるという欠点がある。

そこで本稿では、RTP と SCTP (Stream Control Transport Protocol; RFC4960)[5] の拡張である PR-SCTP (Partial Reliable SCTP; RFC3758)[6] を組み合わせることにより、バースト的なパケットロスに対する耐性を持ちつつリアルタイムストリーム伝送を行うことを提案する。PR-SCTP では、データの送信時に時間制約を指定することができ、従来はアプリケーションが行っていた時間制約を考慮しての再送やタイムアウト制御の処理をトランスポート層で行うことが可能であることから、バッファリングを行いつつ PR-SCTP によりあらかじめ指定した遅延制約の範囲

内で再送を行うことで、遅延と引き換えにバースト的なパケットロスに対する耐性を高めることができる。

さらに、RTP/UDP で実装されている既存のアプリケーションとの互換性を考慮して提案方式を RTP の中継器として実装し、Robst を用いて評価を行った。

以下、2章においてパケットロスに対する対策、既存のシステムと問題点と SCTP と拡張である PR-SCTP について述べる。3章でシステムの提案を行い、これに基づき第4章でシステムの設計について述べる。5章において本稿のまとめを行う。

## 2. 背景

遠隔講義システムなどに代表される映像や音声をリアルタイムで伝達するアプリケーションにおいてパケットロスは、映像や音声の乱れや途切れの原因となる。そのためアプリケーションではこの乱れや途切れを最小限に抑えるためにパケットロスに対する対策を講じている。

本章ではパケットロスの影響を抑えるための手法として FEC について簡単に説明し、これを用いて設計されたシステムとして Robst について紹介する。

また、本稿において提案するシステムにおいて RTP/UDP の代わりに使用する SCTP およびその拡張である PR-SCTP について説明をする。

### 2.1. パケットロスへの対策と既存のシステム

#### 2.1.1. FEC

FEC(Forward Error Collection)とは、再送を行わずに受信者側において誤り訂正ができる符号化方式のことである。FEC では送信者側においてデータブロックに冗長なデータを作成し本来のデータと冗長なデータの両方を送信することで、受信者側においてデータブロックの欠損を少ない遅延で復元することを可能とする。

FEC で用いられる代表的な符号化である Reed-Solomon においては、同じ長さの  $N$  個のデータに対して、 $M$  個の冗長なデータを生成して  $N+M$  個のデータを送信する。受信側に置いてこの  $N+M$  個のうち任意の  $N$  個が受信できれば受信者側において  $N+M$  個のすべてのデータの復元を行うことができる。しかし、受信者側において  $N$  個未満しかデータが受信できなかったとき、つまり  $M+1$  個以上のデータが損失されてしまった場合データを復元することが不可能となる。このため、パケットが連続して大量に欠落するバーストパケットロスに対して弱いと言える。また上記における冗長データブロック数  $M$  の数を多くすることで復元度を上昇させることができるが送信する伝送量が増えてしまい、狭帯域においてはバーストパケットロスを引き起こす原因となってしまう。

このことから、FEC はバーストパケットロスが頻繁に起こる経路や狭帯域における経路では仕様しづらい

という性質を持つ。

#### 2.1.2. 既存のシステムとその問題点

FEC を用いたリアルタイム伝送のアプリケーションとして Robst を挙げる。

Robst は広島大学で開発された FEC を用いることでパケット損失への耐性を強化した動画伝送ツールである。このアプリケーションでは MPEG2-TS 形式および DV 形式のストリーム映像をインターネット上で送受信することが可能となっている。このシステムでは送信開始時に冗長度を設定することが可能となっているため、広帯域でありかつバーストパケットロス起こらない時であれば FEC による回復が見込める。しかし、途中で使用できる帯域が狭まった場合においては使用できる帯域にかかわらず常に一定のレートでデータを送信し続けるためバーストパケットロスを引き起こすという欠点がある。

### 2.2. SCTP および PR-SCTP

#### 2.2.1. SCTP

SCTP(Stream Control Transmission Protocol)とは RFC4960 によって定義された次世代トランスポートプロトコルである。このプロトコルにおいてデータはチャンクという単位で伝送され、チャンクには TSN (Transport Sequence Number)が割り振られる。この TSN によってチャンクのロスや重複を検知し、TSN に対して ACK が返される。

またチャンクは自由に大きさを指定することができる。そのため、1チャンクを1パケットにすることや動画の1フレームなどにすることも可能である。しかし、チャンクデータの1部分でも欠けるとすべて届かなかったとしてアプリケーションに伝えられる。

#### 2.2.2. PR-SCTP

PR-SCTP は RFC3758 によって提案された SCTP の拡張の一つであり、パケットに対して部分的に信頼性を持たせることが可能となっている。このとき部分的信頼性を与える対象は時間・回数などである。

部分的な信頼性をもたすために Forward TSN (Forward Cumulative TSN)チャンクを新たに設定し、メッセージを送信または再送信しようと試み続けるべき時間および回数に期限をつけることが可能となっている。この Forward TSN はまだ受信または確認応答がなされていない過去の一つ以上の DATA チャンクをとばすべきであると指示している。この再送手順として、送信ホストは相手ごとに Advanced.Peer.Ack.Point を管理する。そして、すべてのチャンクに対してアプリケーションは再送を行うか行わないかを決定することができる。次に、再送を行わないデータチャンクを “abandoned” とする。このとき送信ホストはこのデータチャンクを最終的に確認応答されたとし送信処理中

ではないとして扱う。再送を行うデータチャンクはアプリケーションから得た再送信を行いたい時間内および回数内であれば再送が行われる。

信頼性を持たせる時間を 0ms とすることでデータを送った直後に Forward TSN を送ることで ACK が帰ってくるのを待つことなくデータを送り続けることで UDP のように動作させることもできる。

### 3. システムの提案

#### 3.1. PR-SCTP の利用

現状のリアルタイム伝送アプリケーションでは通常トランスポート層プロトコルとして UDP/RTP が用いられる。これはリアルタイム映像伝送の少量のロスが起こってもかまわないので大量のデータを扱いたいという目的が、TCP に比べて UDP に適しているからである。しかし、UDP だけでは送信元および送信先のアドレスとポートしかヘッダ情報としてないためデータの順番がわからない他、再送なども行われぬ。そこで RTP と組み合わせることで RTP のヘッダからパケット番号・タイムスタンプ・識別子などの情報を得ることができる。UDP/RTP ではトランスポート層で再送が行われぬため FEC など受信者側において誤り訂正を行うことでパケットのロスを最低限回復している。バースト的なパケットロスには弱いという欠点がある。

そこで、私たちはバーストパケットロスに弱いという欠点を克服するために PR-SCTP における時間的部分信頼性に着目した。この PR-SCTP の部分信頼性では、アプリケーションによって指定した時間内であれば再送を行い、時間を超えると再送を行わないという動作をトランスポート層で行われる。もし RTP/UDP 方式で再送を行おうとすれば、受信側アプリケーションから送信側アプリケーションに再送要求を行わなければならないが遅延が大きくなるとともにアプリ作成ユーザの負担も大きくなる。

そこで本稿では RTP と PR-SCTP を組み合わせ、時間的部分信頼性によってあらかじめ指定した遅延制約の範囲内で再送を行うことにより遅延と引き換えにバースト的なパケットロスに対する耐性を高めることができるシステムを提案する。

#### 3.2. 中継器による送信

提案するシステムではトランスポート層プロトコルとして PR-SCTP が用いられる。しかし現状において SCTP はよく知られているプロトコルとは言い難いことから RTP/UDP で実装されている既存のアプリケーションとの互換性を考慮し、中継器として使用できるシステムを提案する。図 1 で中継器において流れるデータの図をしめす。中継器の中で PR-SCTP によるデータを送信するホストを送信変換ホスト、PR-SCTP のデータを受け取るホストを受信変換ホストと定義する。

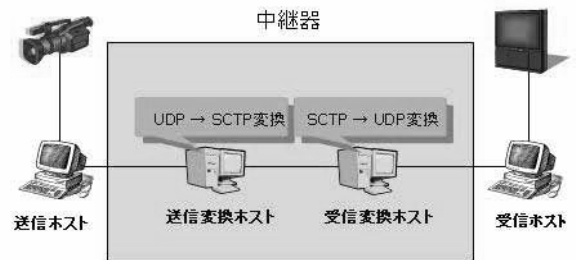


図 1 中継器

#### 3.2.1. 送信変換ホストの動作

送信ホストのアプリケーションから受信ホストへとデータを送る時、送信ホストは受信ホストの IP アドレスおよびどの程度の時間的信頼性を与えるかというメッセージを送信変換ホストへ送る。その後、送信ホストから中継器内の送信変換ホストへとデータを受け取ると、受け取ったデータを受信変換ホストに対して SCTP パケットで送信を行う。

#### 3.2.2. 受信変換ホストの動作

受信変換ホストは送信変換ホストから PR-SCTP によってデータが送られてくるとそのメッセージから SCTP ヘッダをはずし UDP データとして送信ホストへ映像の伝送を行う。

### 4. システムの設計

#### 4.1. 想定する環境

本稿ではベストエフォート型サービスの下で一本の経路かつ一方方向でのリアルタイム映像を伝送する。ただし、送信ホストから受信ホスト間において送信ホストで使用するアプリケーションが使用する帯域を下回ることが無く、またランダムおよびバーストパケットロスは起こるが断線は起こらないという場合を想定している。今回想定した環境について示したものが以下の図 2 である。

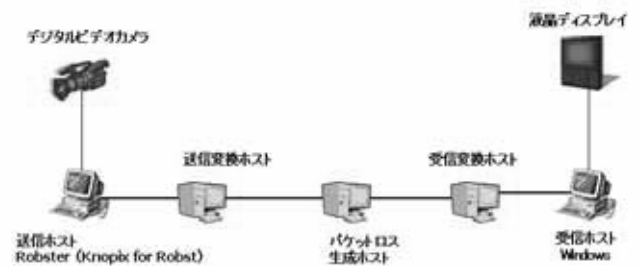


図 2 想定する環境

#### 4.2. 中継器

現在利用されているリアルタイム映像伝送システムの大半はアプリケーション層で RTP を用いトランス

ポート層で UDP を用いている。このシステムをすべて PR-SCTP で送るよう拡張することは非現実的である。

#### 4.2.1. 中継器として実装

本稿において提案するシステムを中継器として実装し、送信ホスト・受信ホスト間にこのシステムを入れることで中間部において PR-SCTP が使用することができる。

この環境を示したものが図 2 である。送信ホストとして Robster2.0 (Knoppix for Robst) を使い、デジタルビデオカメラから 1394 接続で取り込んだ HDV 映像を送信ホストから中継器の受信変換ホストに向けて MPEG2-TS で送信を行う。そして受信ホストでは Windows 上で映像受信アプリケーションとして VLC media player[6]を用いた。

また中継器の一部である送信変換ホストおよび受信変換ホストは FreeBSD7.0 において開発を行った。詳細な中継器の動作については 4.3 節において説明する。

### 4.3. 中継器の動作

#### 4.3.1. 送信変換ホストの動作

送信変換ホストの動作について説明する。

まず今回使用する Robst のデフォルトのポート番号が 14000 であることから FreeBSD に組み込まれていたパケットフィルタ機能である IPfw を用いて UDP かつ送信ホストのアドレスの 14000 ポートへ配送されるパケットを任意の divert ポートへと送る設定をしておく。

- (1) 送信が開始される前に送信ホストから何秒までの遅延を許容するかのメッセージを受け取る
- (2) 送信ホストからの RTP/UDP パケットが受信され、任意の divert ポートへ送られる
- (3) divert ポートから開いたソケットのデータを一つの SCTP チャンクとして受信変換ホストへ向けて送信を行う

図 3 において送信ホストから送られてくるパケットの中身を、図 4 において送信変換ホストから送信されるパケットの中身を示す。

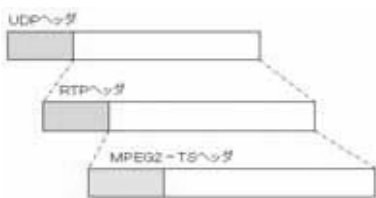


図 3 送信ホストからのデータの中身



図 4 送信変換ホストからのデータの中身

#### 4.3.2. 受信変換ホストの動作

- (1) 送信変換ホストからのデータを受け取る
- (2) 受け取ったデータの中の RTPヘッダのシーケンス番号から順序通りに並び替える
- (3) SCTP ヘッダを取り去り、更にデータ部の UDPヘッダから受信ホストのアドレスおよびポート番号を読みとる
- (4) 読みとった送信先アドレスのポートに対してデータの送信を行う

### 4.4. パケットロスに対する耐性実験

提案したシステムにおいて生じる遅延は PR-SCTP によるあらかじめ指定した遅延制約の範囲内で再送しない状態で 2.3 秒の遅延が存在する。これは、ビデオカメラからの映像を robst へ取り込みと中継器に含まれる 2 台でのヘッダの書き換えなどで生じる遅延である。

本稿で提案したシステムであらかじめ指定した遅延制約の範囲内で再送を行うことで、遅延と引き換えにバースト的なパケットロスに対する耐性を高めることができるということを示すために、実際にランダムパケットロスおよびバーストパケットロスを実際に起こしてみても耐性実験を行った。

実験方法として図 2 のパケットロス生成ホストにおいてパケットロスを発生させた。パケットロスを発生させる手段として netem の tc コマンドにより擬似的に発生させる。

### 4.5. 考察

RTP/UDP では輻輳制御は行われなため送信レートが変化することはない。しかし、RTP/PR-SCTP においてはランダムパケットロス・バーストパケットロスの両方の場合に再送が行われる。この時、PR-SCTP では再送イベントの段階であるため、普通輻輳ウィンド縮んでおり、Forward TSN が送信される時には更に輻輳ウィンドは縮むと考えられる。

また今回はローカル環境による非常に近いノード間で実験であるため、RTT がかなり短い環境である。そのため、再送があってもすぐに輻輳ウィンドが伸びることから実環境においても実験を行わないといけない。

### 5. まとめ

本稿において、リアルタイム型マルチメディア・アプリケーションでリアルタイムストリーム伝送を行う際に RTP/UDP でデータを送る代わりに RTP/PR-SCTP でデータを送ることにより、RTP/UDP で使用される FEC の持つバースト的なパケットロスは復元しにくいという欠点を克服しつつリアルタイムストリーム伝送を行うことができることを示した。

さらに、既存のアプリケーションとの互換性を考慮し RTP の中継器としてこれを実装した。

今後は本システムの有効性を検証する実験を行うと共に SCTP にはパス切り替えの機能があることから経路状況が悪化している場合に経路切り替えを自動的に行えるシステムを構築する予定である。

## 文 献

- [1] 広島大学情報メディアセンター  
<http://lab.net.hiroshima-u.ac.jp/robst/>
- [2] DVTS Consortium  
<http://www.dvts.jp>
- [3] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, “RFC3550 - RTP: A Transport Protocol for Real-Time
- [4] J. Rosenberg, H. Schulzrinne, “RFC 2733 - An RTP Payload Format for Generic Forward Error Correction”, December 1999
- [5] R. Stewart, Ed “RFC4960 - Stream Control Transmission Protocol”, September 2007
- [6] R. Stewart, M. Ramalho, Q. Xie, M. Tuexen, P. Conrad, “RFC 3758 - Stream Control Transmission Protocol (SCTP) Partial Reliability Extension”, May 2004

