

# 不均質ネットワーク環境における効率的なマルチキャスト 高信頼化のための非インライン型 FEC 装置

佐々木 力<sup>†1</sup> 田上 敦 士<sup>†1</sup>  
長谷川 輝之<sup>†1</sup> 阿野 茂 浩<sup>†1</sup>

近年、映像配信サービスの増加にともない、同一コンテンツを多数の受信者に同時配信可能な IP マルチキャストが注目されている。IP マルチキャストでは、通信を高信頼化するために FEC (Forward Error Correction) が広く利用される。通常、FEC の符号化処理は配信元で一括して行われるため、多様な通信品質が混在する環境では、高品質な受信者に対して必要以上の冗長配信が行われることが問題となる。これに対し、低信頼区間に別途 FEC 装置を導入し、配信元での冗長化率を抑制するアプローチが考えられる。FEC 装置の実現形態としては、区間両端の回線上にインライン設置する方式があげられるが、高信頼化対象のマルチキャストだけでなく、当該区間を通過する他のトラフィックすべてを対象とした転送処理能力が求められる。そこで本稿では、マルチキャストを対象に、その経路・転送制御を利用することで、区間両端のルータへ付加的に接続した FEC 装置に対象マルチキャストのみを誘導し高信頼化する、非インライン型の手法を提案する。試作装置を用いた評価実験により、提案手法が他のトラフィックの転送にいったい影響を与えることなく、対象マルチキャストの高信頼化が実現できることを確認した。

## A Study of Non-inline FEC System for Efficient Reliable Multicast in Heterogeneous Network Environment

CHIKARA SASAKI,<sup>†1</sup> ATSUSHI TAGAMI,<sup>†1</sup> TERUYUKI HASEGAWA<sup>†1</sup>  
and SHIGEHIRO ANO<sup>†1</sup>

IP multicast is used for video delivery since it can distribute content to many receivers simultaneously. FEC (Forward Error Correction) has been widely employed to realize efficient reliable multicasts. Generally, the multicast source employs FEC encoding using a common redundancy rate. However, unnecessary transmission of redundant data will occur in high transmission quality receivers in a heterogeneous network environment. The solution could be to introduce selectively an add-on FEC system to both edges of the network segments with low transmission quality to reduce the redundancy rate at the source. In this paper, we study non-inline implementation of an FEC system, which is attached to the edge routers, pulling only target multicasts to improve reliability.

### 1. はじめに

近年、IPTV (Internet Protocol TeleVision)<sup>1)</sup> に代表されるように、大容量の映像・音声コンテンツを IP ネットワーク上でストリーミング配信するサービスが増加しつつある。また、携帯電話向けには最新のコンテンツを定期的に同報するサービスが始まっている<sup>2)</sup>。このような配信サービスに対しては、ルータの packets 複製機能や無線の同報性により多数の受信者に同時配信可能な IP マルチキャスト技術が有効であ

る。一方、高信頼なコンテンツ配信を IP マルチキャストで実現するためには、パケットロスに対する誤り回復技術が必要となる。誤り回復技術としては再送 (ARQ: Automatic Repeat reQuest)<sup>3),4)</sup> や配信データの冗長化 (FEC: Forward Error Correction)<sup>5),6)</sup> 等があげられるが、多くの受信者が想定されるマルチキャスト配信では、サーバでの応答爆発等の問題から、FEC の利用が一般的である。

通常、FEC の符号化処理は配信元で行い、全受信者で同じ冗長化率を用いる。このため、パケットロスが最も頻繁に発生する低品質な受信者に合わせた冗長化が必要となる。これは、高品質な通信環境の受信者に対して余分なデータが送信されることを意味する。

<sup>†1</sup> 株式会社 KDDI 研究所  
KDDI R&D Laboratories Inc.

多種多様な回線・デバイスが存在するユビキタスネットワークや有線・無線環境が ALL IP で統合された FMC (Fixed Mobile Convergence) ネットワークにおいては、回線のコストや帯域あるいは通信品質が多様化した不均質ネットワーク環境が想定され、一部の低信頼区間がボトルネックになって多数の受信者に不必要なデータ転送が発生することは問題となる。このような環境でネットワークリソースを有効利用するためには、配信元での冗長化率を抑制し FEC の冗長化率を特定区間のみ変更する必要がある。

特定区間のみ高信頼化する手法としては、FEC 装置を区間両端の回線上にインライン設置する方法が考えられる。この場合、FEC 装置は対象マルチキャストの高信頼化と同時に、それ以外のパケットの中継転送も行わなければならない。そのため、トラフィックの集中する回線に設置した場合は転送ボトルネックとなる。また、FEC 装置に障害が発生すると、その区間を通るすべてのトラフィックに影響が及ぶことも問題である。

これに対し我々は、高信頼化対象区間の両端のルータに付加的に設置可能な非インライン型の FEC 装置を検討する。本 FEC 装置のエンコーダとデコーダは、マルチキャスト転送の仕組みを応用し、対象マルチキャストだけを引き込み FEC 処理を付加する。他の通信は本装置を経由しないため、装置設置による影響を受けない。本稿では、具体的な非インライン型 FEC 装置として、動作原理の異なる 2 種類の方式を提案し比較検討する。

本稿の構成は以下のとおりである。2 章では、対象ネットワークと FEC の概要を述べた後、不均質ネットワーク環境におけるマルチキャスト配信の問題点とインライン型 FEC 装置について説明する。3 章で非インライン型の FEC 装置として、対象マルチキャストの転送はそのままに冗長データのみを追加伝送する「冗長データ追加型」と、対象マルチキャスト転送のみインライン型と同様に FEC 装置を通過させる「リダイレクト型」を提案する。4 章において試作機による性能評価を行い、その結果を 5 章で考察し 2 つの方式を比較する。6 章でまとめを行う。

## 2. 対象ネットワークと既存技術

本章では、対象ネットワークとパケットレベル FEC の概要を述べ、その後で不均質ネットワーク環境におけるマルチキャスト配信の問題点とインライン型の FEC 装置について説明する。

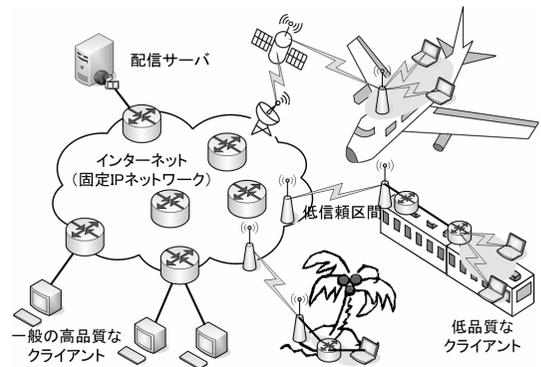


図 1 想定ネットワーク例  
Fig. 1 Example of target network.

### 2.1 対象ネットワーク

対象ネットワークの例を図 1 に示す。図 1 において、一般のクライアントは全区間で高品質な固定網を経由してマルチキャストを受信する。他方、飛行機内や電車内の LAN (Local Area Network) に接続しているクライアントも一部存在し、当該 LAN と外部インターネットは無線通信でつながっている。このとき、機内・車内で使用する LAN は比較的高品質である。しかしながら、海上や山間部等無線アクセスポイントが少ないあるいは遮蔽物の多いような低品質な地域を通過しなければならない飛行機や電車も存在し、これらの機内・車内に収容されるクライアントは外部通信の品質劣化にともない一定時間低品質な通信を強いられる。また離島との通信に無線を用いる場合もあり、天候等によって島内クライアントのインターネット通信品質が低下することもある。このように、本稿では多数 (e.g. 数 100 万規模) の比較的高品質なクライアントとごく少数 (e.g. 数 100 規模) の低品質なクライアントが混在する不均質ネットワーク環境を対象としている。

本稿が対象とするマルチキャスト通信では、一般にトランスポート層のプロトコルとして UDP (User Datagram Protocol) を用いるため、全クライアントに対して配信元で FEC 処理を行い通信を高信頼化することを想定している。

本稿では、以下の理由によりパケット単位での FEC 処理を対象とする。

- (1) IP ネットワーク上の通信エラーは、基本的にパケットロスに統一して扱うことができる。
- (2) レイヤ 2 におけるフレーム単位での FEC 処理の冗長化率を変更する方法では、一般的に、
  - (a) デバイスを別途用意しなければならない可能性が高いこと

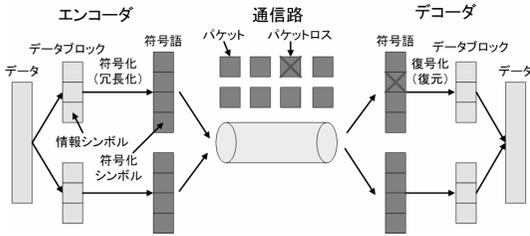


図 2 FEC 処理の流れ  
Fig. 2 FEC processing flow.

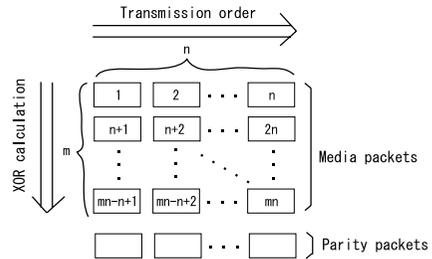


図 3  $(mn + n, mn)$  パリティ符号  
Fig. 3  $(mn + n, mn)$  parity codes.

- (b) 実装を簡便化するために、TCP (Transmission Control Protocol) によるユニキャスト通信 (マルチキャストに比べ高信頼) に対しても冗長化が適用される可能性が高いこと<sup>\*1</sup>
- (c) 衛星中継のように他のインフラを経由する場合、当該区間が異なる管理下のため仕様変更が困難な可能性があること  
等が問題となる。

2.2 FEC の概要

図 2 に FEC 処理の流れを示す。符号化と復号化は固定長のデータブロック単位で行うものとする。エンコーダ (配信サーバ) では、データを固定長のデータブロックに分割し、データブロックをさらに情報シンボルに分割する。次に、同一データブロック内の情報シンボルどうしを演算し、その結果を符号化シンボルとする。このとき、生成する符号化シンボル数  $n$  を情報シンボル数  $k$  よりも多くすることで冗長化が行われる。これら同一のデータブロックから生成された符号化シンボルを結合したものを符号語と呼ぶ。本稿では、情報シンボルと符号化シンボルの単位はパケットとする。通信経路上でいくつかのパケットロスが発生したとしても、デコーダ (クライアント) は冗長化した程度のロスであれば、データブロックを復元することができる。以上のような冗長化率  $n/k$  の符号を  $(n, k)$  符号と呼ぶ。

次に、単純な  $(mn + n, mn)$  パリティ符号の例を図 3 に示す<sup>7)</sup>。エンコーダは、データブロックをそのまま  $mn$  個のメディアパケットに分けて送信し、さらに、冗長性を持たせるために  $n$  個のパリティパケットを送信する。パリティパケットのペイロード部は、 $m \times n$  行列を用いた符号化処理により生成され、 $m$  個のメディアパケットのペイロード部を行列の列方向に

XOR 演算したものに、FEC ヘッダや RTP (Real-time Transport Protocol) ヘッダ<sup>8)</sup> を付加したものである。これらのヘッダには FEC パラメータやシーケンス番号等が含まれる。図 3 の例では、メディアパケットの送信はデータ入力後すぐに行うことができる一方、パリティパケット送信はこれに必要な  $m$  個のメディアパケット送信後となる。デコーダは、同列内の 1 メディアパケットのロスについては、受信した  $m - 1$  個のメディアパケットと対応するパリティパケットの XOR 演算により復元することができる。また、同行列内の  $n$  個のバーストロスを復元することができる。

2.3 不均質ネットワーク環境におけるマルチキャスト配信の問題点

全クライアントに対して同一の冗長化率を用いる一般的なマルチキャスト配信形態では、品質が最も劣るクライアントに合わせた冗長化率での運用が必要となる。たとえば、図 1 の対象ネットワークにおいて 5 Mbps のコンテンツを配信する場合を考える。数 100 万規模の一般クライアントがエンド・エンドで 1.2 倍の冗長化 (6 Mbps) で十分な高信頼性を確保できたとしても、数 100 規模のわずかに存在する低品質クライアントで 1.4 倍の冗長化 (7 Mbps) が必要であれば、配信元での冗長化拡大が必要となり、ネットワーク全体にわたって 7 Mbps の帯域が必要になる。すなわち、一部の低信頼区間がボトルネックになって多数の受信者に不必要なデータ転送が発生する。

一般に配信経路上の一部の無線区間の品質を確保する場合や、衛星中継・宅内無線 LAN のように配信サーバとは異なる管理下にある低信頼区間に対応する場合は、全体の冗長化率を変更しなければならない。しかしながら、装置の仕様や管理範囲の関係から送受信装置の FEC 冗長化率の変更が困難なシナリオや可能な限りマルチキャスト帯域を削減したいという要求シナリオが考えられる。この場合、配信元で行う FEC 処理とは別に、特定の低信頼区間 (以下単に特定区間

\*1 対象マルチキャストのみを冗長化するためには、IP ヘッダ (レイヤ 3) を参照して識別する必要があるため。

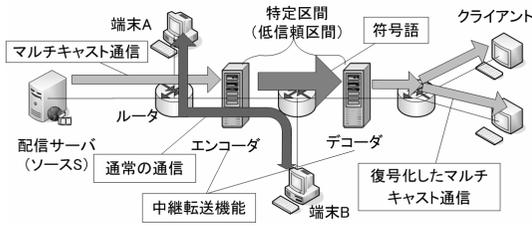


図4 インライン型 FEC 装置

Fig. 4 Inline type FEC system.

と呼ぶ)のみさらに外付け FEC 装置を適用する方法が考えられる。上述の例では、配信元での冗長化率を 1.2 倍に維持したままで、特定区間のみを外付け FEC 装置により 1.4 倍の冗長化率にすることで、一般クライアントに対する帯域を 6 Mbps に抑制したままで全クライアントに対する高信頼化が可能となる。なお、必要となる FEC 装置の台数は特定区間<sup>\*1</sup>の数に応じて必要となる。たとえば山間部を通過する電車を想定したシナリオでは、数箇所アクセスポイントと当該区間を運行する数台の電車に設置することを想定している。

#### 2.4 インライン型 FEC 装置

これまでに提案されている外付け FEC 装置としてはインライン型があげられる<sup>9)</sup>。これをマルチキャストに適用することも可能性である。図4にインライン型の概要を示す。エンコーダは、通過するパケットの中から対象とするマルチキャストパケットを抽出し、符号化処理を行ってデコーダ側へマルチキャスト転送する。デコーダは、同様に通過するパケットの中から符号化処理されたパケットの抽出やロスを検出等を行ったうえで、復号化処理を行う。本形態では、対象とするマルチキャストパケットを符号化・復号化しつつ、それ以外のパケット(図4: 端末 AB 間の通信)を中継転送する必要がある。そのため、特に広帯域回線においては、FEC 処理対象となるマルチキャストの増加による中継転送スループットの低下や、その逆の現象が発生する可能性がある。また、FEC 装置に障害が発生すると、対象マルチキャストだけでなく、装置を経由する全通信に影響が及ぶこととなる。このように、インライン型 FEC 装置は、スループットや可用性の観点から性能ボトルネックとなりうる。

\*1 ポイントツーマルチポイントあるいはポイントツーポイントとなる。

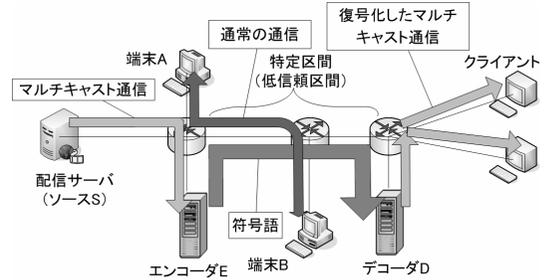


図5 非インライン型 FEC 装置

Fig. 5 Non-inline type FEC system.

### 3. 提案手法

#### 3.1 非インライン型

本章では、FEC 装置がボトルネックとならないように、対象マルチキャスト以外の中継転送をほとんど必要としない非インライン型 FEC 装置を検討する。具体的には、図5に示すように、FEC 装置を対象区間の両端のルータに接続し、対象マルチキャストやその冗長データのみをエンコーダとデコーダ経由で転送させる一方、端末 AB 間の通信は従来の経路を保持するように設計する。

非インライン型の形態として、以下2つを検討する。

- 対象マルチキャストをエンコーダに引き込む一方、そのまま特定区間に流す。エンコーダでは冗長データを追加生成しデコーダへ転送する。
- 対象マルチキャストをエンコーダにすべて引き込み、特定区間にはそのまま流さない。エンコーダは冗長データを含む符号語全体をデコーダへ転送する。

前者を冗長データ追加型と呼び、後者をリダイレクト型と呼ぶ。以下、3.2 節で冗長データ追加型を説明し、3.3 節でリダイレクト型を説明する。

方式検討にあたり、経路制御・グループ管理プロトコルとして以下を想定した。

- OSPF (Open Shortest Path First)<sup>10)</sup>
- PIM-SM (Protocol Independent Multicast - Sparse Mode)<sup>11)</sup>
- IGMP (Internet Group Management Protocol)<sup>12)</sup>

以下、マルチキャストのソースを  $S$ 、エンコーダを  $E$ 、デコーダを  $D$  とし、 $(S, G)$  でソース  $S$ 、マルチキャストアドレス  $G$  のマルチキャストを表す。PIM-SM では本来、マルチキャストがソース  $S$  を根とする SPT (Shortest Path Tree) 以外にも、RP (Rendezvous Point) を根とする共有木 (RPT) 経由で転

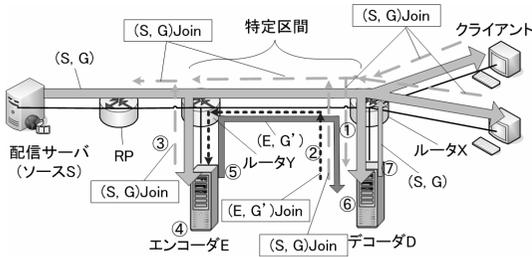


図 6 冗長データ追加型 FEC 装置  
Fig. 6 Parity injection type FEC system.

送されることがある。ただし、本検討では簡単のため、

- RPT から SPT への切替えが即時に行われ、主な配信は SPT 経路で行われること、
- RPT と SPT は経路が共通であること、

を前提に、SPT ( $(S, G)join$ ) に対する高信頼化手順を述べることにする。

### 3.2 冗長データ追加型

冗長データ追加型では、高信頼化対象マルチキャスト  $G$  を対象区間にそのまま転送しつつ、追加の冗長データをエンコーダからデコーダへ送信する。エンコーダからデコーダへの転送は、マルチキャストアドレス  $G' (\neq G)$  を用いる。図 6 に冗長データ追加型の概要を示す。具体的手順は以下のとおりである。

冗長データ追加型の手順：

- (1) デコーダは高信頼化対象マルチキャスト  $(S, G)$  について、ルータ X からルータ Y へ向かう  $(S, G)join$  を監視する。
- (2) デコーダは  $(S, G)join$  を検出すると、デコーダ受信用に  $(S, G)join$  と  $(E, G')join$  を行う。
- (3) エンコーダは  $(E, G')join$  を検出すると  $(S, G)join$  する。
- (4) エンコーダは受信した  $(S, G)$  に符号化処理を行う。
- (5) 生成した冗長データは、別マルチキャスト  $(E, G')$  を用いてエンコーダから送信される。
- (6) デコーダは受信した  $(S, G)$  のシーケンス番号からパケットロスを検出すると、 $(S, G)$  と  $(E, G')$  からロスパケットを復元する。
- (7) デコーダは復元パケット  $(S, G)$  をルータ X における  $S$  宛での経路に対応したインタフェースへと送信する。

□

ルータでは、ループ防止のために、RPF (Reverse Path Forwarding) チェック<sup>13)</sup> が動作しており、ソース  $S$  (もしくは RP) に向かうユニキャスト経路に対応するインタフェースのみマルチキャスト  $(S, G)$  の受

信が許可される。そこで、手順 (7) においてデコーダが送信した復元パケットが破棄されないように、ルータ X では  $S$  向きの上流ルータ Y とデコーダを同一 LAN セグメントに收容する。本收容方法により、手順 (1) でクライアントからルータ Y へ送信された  $(S, G)join$  も、デコーダで検出することができる。また、高信頼化対象はマルチキャストであるため、手順 (3) では、エンコーダが  $(S, G)join$  するだけで、当該マルチキャストパケットを受信することができる。

冗長データ追加型で利用できる誤り訂正符号は、単純パリティ符号や LDPC 符号<sup>14)</sup> のように、符号語自身にデータブロックが含まれる符号に限定される。エンコーダへの入力と同一のデータはルータ Y によってそのまま対象区間へ転送されるので、エンコーダは生成データの中から入力と同一内容のデータを除いた冗長分のデータのみ転送する。エンコーダからデコーダへの転送には別マルチキャスト  $(E, G')$  を用いる。そのため、1 台のエンコーダに対して、下流に多数のデコーダを設置することが可能であり、またデコーダ以外には転送されないようにできる。

デコーダは、受信した  $(S, G)$  の中でロスパケットのみ復元し転送すればよいが、ロス検出や復号処理等により、復元したパケットの到達に遅延が発生する。すなわち、クライアント側では到達パケットに順序逆転が発生する可能性があり、アプリケーション層で到達逆転を補償する必要がある。なお、デコーダからの復元パケットは、ルータ Y にも転送される。多数のロスが発生すると復元パケット数も増大するため、ルータ X でのフィルタリングが有効となる場合もある。

### 3.3 リダイレクト型

#### 3.3.1 動作説明

冗長データ追加型では、復元パケットが遅れて到達し、また使用できる誤り訂正符号が限られる。そのため、本節では、マルチキャストの経路制御を利用して対象マルチキャストの転送経路を変更させることで、マルチキャスト全体をインライン型のようにエンコーダとデコーダ経路で高信頼化可能とするリダイレクト型の FEC 装置を検討する。図 7 にリダイレクト型の概要を示す。リダイレクト型では、ソース  $S$  からのマルチキャスト  $(S, G)$  をエンコーダに引き込み、エンコーダとデコーダ間は符号語をマルチキャスト  $(E, G')$  に変換してリダイレクトし、デコーダからの復号データをマルチキャスト  $(S, G)$  に戻してクライアントへ転送する。具体的手順は以下のとおりである。

リダイレクト型の手順：

- (1) デコーダは接続するルータ X に適切なコスト

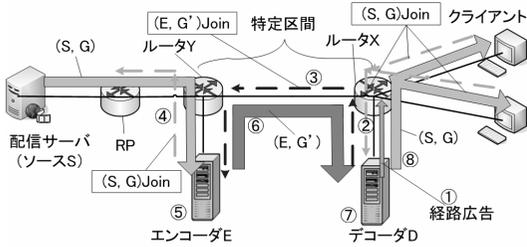


図 7 リダイレクト型 FEC 装置  
Fig. 7 Redirection type FEC system.

に基づく経路情報を広告し、ルータ X の保持するソース S への経路をデコーダに向ける。

- (2) ルータ X はクライアントからの join を受信後、(S, G) を配信するために、ルーティングテーブルに従って、デコーダへ (S, G) join する。
- (3) ルータ X から (S, G) join を受信したデコーダは、(E, G') に join する。
- (4) エンコーダは (E, G') join を検出すると (S, G) join する。
- (5) エンコーダは受信した (S, G) に符号化処理を行う。
- (6) 生成した符号語は、別マルチキャスト (E, G') を用いて送信する。
- (7) デコーダは受信した (E, G') を復号する。
- (8) デコーダは復号した全パケットを (S, G) に変換してルータ X に送信する。

□

手順 (1) において、ルータ X の保持する S 宛ての経路情報を変更させることによって、RPF チェックを回避し、デコーダから送信されたマルチキャストを (S, G) としてクライアントに転送することができる。ただし、ルータ X でソース S 宛ての経路をデコーダ方向に変更しているため、ルータ X を経由して S に至るユニキャスト通信もデコーダに転送される。そのため、デコーダとエンコーダ間にユニキャストの IP in IP トンネル<sup>15)</sup>を設定し、S への到達性を確保する。

### 3.3.2 広告するコスト

以下、図 8 を用いて、デコーダが広告するコスト C について検討する。コスト C は以下の 2 つの条件を満足する値に設定する。

条件 1: ルータ X の保持する S 宛ての経路をデコーダに向けさせる。

条件 2: 広告したコストによって、X 以外のルータの経路にできるだけ影響を与えない。

ルータ X に隣接するルータ集合を  $Z_1$  ( $Z_1 = \{\text{ルータ 1, ルータ 3, ルータ Y}\}$ )、 $Z_1$  の中でソ-

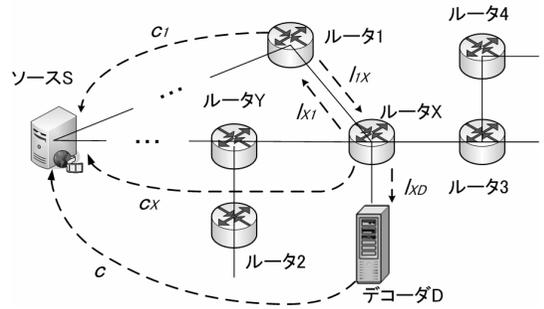


図 8 OSPF のコスト設定例  
Fig. 8 Example of OSPF cost setting.

ス S までの経路にルータ X を含むルータの集合を  $Z_2$  ( $Z_2 = \{\text{ルータ 3}\}$ ) とする。ルータ i から S に至る経路のコストを  $C_i$ 、ルータ i からルータ j へのリンクコストを  $l_{ij}$ 、ルータ X からデコーダ D へのリンクコストを  $l_{XD}$  とする。なお、これらのルータは OSPF の同一エリア内に属しているものとし、 $C_i$  や  $l_{ij}$  は LSA (Link State Advertisement) を収集し計算することとする。条件 1 に関しては、ルータ X で保持しているソース S 宛ての経路コスト  $C_X$  よりもデコーダからの広告に基づく経路コストの方が小さくなればよいため、

$$C + l_{XD} < C_X \tag{1}$$

を満足すればよい。ただし、 $C > 0$  である必要があるため、式 (1) より本方式は、

$$C_X > l_{XD}$$

を満足している環境でのみ有効である。

次に、条件 2 を検討する。式 (1) を満足するようにコスト C を小さく設定したとしても、 $Z_2$  に含まれるルータのネクストホップはルータ X のままである。したがって、 $Z_1 - Z_2$  に対してのみ、経路変更が起こらないようにする。条件 2 に対しては、

$$C + l_{XD} + l_{iX} > C_i \quad \text{for all } i \in Z_1 - Z_2$$

$$\Leftrightarrow C > \max_{i \in Z_1 - Z_2} (C_i - l_{iX}) - l_{XD} \tag{2}$$

を満足すればよい。ただし、 $\max_{i \in A} f(i)$  は空間 A における  $f(i)$  の最大値を表す。

式 (1)、式 (2) より、コスト C は、

$$\max_{i \in Z_1 - Z_2} (C_i - l_{iX}) - l_{XD} < C < C_X - l_{XD} \tag{3}$$

の範囲内で設定する。

なお、ルータにおいて転送パケットがルーティングテーブル上の複数のプレフィックスに一致する場合、各プレフィックスのコストの大小にかかわらず、最長一致<sup>13)</sup>するものが選ばれる。そのため、デコーダが

ら広告するソース  $S$  に関する経路情報のプレフィックス長は、エリア内で交換されている当該経路情報と同一にする必要がある。デコーダからの広告プレフィックスが長い場合は、同一エリア内での  $S$  宛での通信がすべてデコーダ経由となり、また短い場合はより長いプレフィックスを持つ従来の経路情報が有効となり、デコーダへのリダイレクトが実現できないことに注意する。

#### 4. 性能評価

##### 4.1 評価環境

本章では試作した FEC 装置の評価を行う。評価環境を図 9 に、装置諸元を表 1 に示す。具体的には、ソース  $S$  にトラフィックジェネレータ IXIA<sup>16)</sup> を用いて、マルチキャスト ( $S, G$ ) を 10 Mbps で 100 万パケット送信する。フレーム長は 1,250 Byte (1 秒あたり 1,000 パケット送信) とする。誤り訂正符号は  $(mn+n, mn)$  パリティ符号を使用する。特定区間でのパケットロスは確率  $p$  のランダムロスを想定し、dummynet<sup>17)</sup> によって模擬する。また他のユニキャスト通信を発生させるために端末 A と B をソース  $S$ ・クライアントとは別の LAN セグメントに設置する。なお、定常状態を評価するために、エンコーダとデコーダは常時マルチキャストの送受信と符号化・復号化を行っているものとする。

冗長データ追加型では、デコーダで復号されたパケットは対応するパリティパケットの受信待ち時間や処理時間分遅れてネットワークに送信されるため、クライアントで到達順序逆転が発生する。一方、全マルチキャストパケットがデコーダを経由するリダイレクト型では、デコーダで送信パケットのバッファリング機能と整列機能を実装し、バッファリング範囲内での到達順序を保証している。通常は、復号パケットの送信順序さえ保証すれば十分であるためバッファリング範囲は同一データブロック内 ( $mn$  個のパケットから構成される符号化行列) とすればよい。一方で、特定区間においてデータブロックの境界を越える到達順序逆転が発生する可能性もある。そこで、リダイレクト型において、データブロック境界付近のメディアパケットに対しても最低限の順序逆転を保証するために、さらに次のデータブロックの  $d$  個のメディアパケットを受信する間だけバッファリングを延長する機能を実装した。以下の評価では  $d = 7$  に固定する。

リダイレクト型では、OSPF の LSA を広告するルータ (OSPF ルータ) をデコーダとルータ  $X$  の間に設置する。通常、OSPF のリンクコストは基準値を

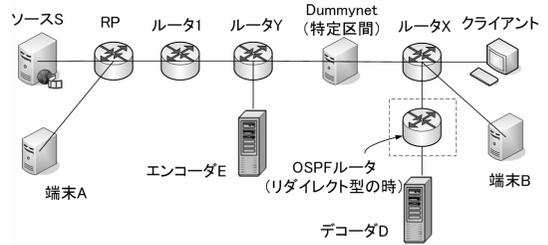


図 9 評価環境

Fig. 9 Network configuration for evaluation.

表 1 装置諸元

Table 1 Hardware specification.

Source	IXIA	IxOS 3.80
RP	Catalyst3560	IOS 12.2(25) SEB4
Other routers	Catalyst3750	IOS 12.2(25) SEA/SEB2
Encoder	CPU 2.0 GHz メモリ 1 GB	Linux 2.4.27
Decoder	CPU 2.0 GHz メモリ 1 GB	Linux 2.6.11
Dummynet	CPU 533 MHz メモリ 760 MB	FreeBSD 5.3
Terminal A, B	IXIA	IxOS 3.80

100 Mbps として以下により計算される。

リンクコスト

= 基準値/インタフェースの帯域幅。

使用したルータの各インタフェースの帯域は基準値以上の 1 Gbps になっているため、通常リンクコストは 1 と計算される。そこで、基準値を 100 Mbps から 10 Gbps に変更する。これにより、ソース  $S$  の所属するネットワークまでの経路コストとしてルータ  $X$  とルータ  $Y$  はそれぞれ 40 と 30 を保持する。よって OSPF ルータでは、 $S$  のネットワーク宛でのネクストホップをデコーダ  $D$  に設定するとともに、式 (3) を満たすように、ルータ  $X$  に対して当該経路をコスト 25 で広告することとした。

##### 4.2 評価結果

本節では、誤り訂正能力と遅延および他通信への影響について評価結果を述べる。

###### 4.2.1 誤り訂正能力

本評価試験では、ソース  $S$  からのパケット転送に対して dummynet によりパケットロスを発生させ、クライアントでパケットロス率 (復号失敗率) を測定する。表 2 にパラメータと得られた復号失敗率を示す。なお、冗長データ追加型において復号処理遅延によってパケットの到達順序が逆転したとしても、ロスパケットが正しく復元できれば復号成功と見なした。誤り訂正能力は用いる符号によって決まるため、表 2 にお

表 2 誤り訂正能力  
Table 2 Error correction capability.

		$p = 1\%$	$p = 5\%$
$m = 5,$ $n = 100$	冗長データ追加型	$5.0 \times 10^{-4}$	$1.1 \times 10^{-2}$
	リダイレクト型	$4.9 \times 10^{-4}$	$1.1 \times 10^{-2}$
$m = 10,$ $n = 100$	冗長データ追加型	$9.8 \times 10^{-4}$	$2.1 \times 10^{-2}$
	リダイレクト型	$9.7 \times 10^{-4}$	$2.0 \times 10^{-2}$

いて冗長データ追加型とリダイレクト型の性能はほぼ同じになった。FEC 装置の導入により、パケットロス率 1%が 0.05 ~ 0.1%程度に修正されていることが分かる。

#### 4.2.2 遅延

2.1 節で述べたように、本稿では高信頼化対象マルチキャスト自体にも配信元で別の FEC が適用されていることを想定している。クライアントでは、ネットワークジッタの吸収や当該 FEC/アプリケーションの処理のためにバッファリングを行う。よって、クライアントにおけるデータ再生時刻という観点では、提案 FEC 装置導入により増加する最大遅延と最小遅延の差がバッファリング量相当の時間内に収まることが重要となる。バッファリング時間としては、リアルタイム系の配信サービスで数 100 ms ~ 数秒程度、蓄積型サービスでは、最大で配信コンテンツサイズの全受信時間相当の長い時間を想定している。

なお、エンド・エンドの遅延量を評価するためには、元々適用されている FEC やアプリケーション処理を含めた遅延量を考慮する必要がある。エンド・エンド遅延量の最大値（の上限）や最小値（の下限）に関しては、単にこれらの処理時間やバッファリング時間の和として評価できる。以上より、本稿の遅延の評価試験では、提案 FEC 装置導入前後の転送遅延の変化に焦点を絞り、ソース  $S$  からクライアントまでの最大転送遅延と最小遅延を評価し、あわせて平均転送遅延も測定した。

試験環境を図 9 に示す。本試験ではクライアントにソース  $S$  と同一筐体の IXIA を使用し、送信時刻と受信時刻の差から遅延を測定した。本評価においては、dummysnet の処理等で転送遅延が発生するため、FEC 装置を設置しない場合の遅延もあわせて測定した。FEC 装置を配置しない場合は、ロス率  $p$  によらず最大遅延約 25 ms、最小遅延および平均遅延 1 ms 未満であった。パケットロス率  $p = 1\%$ 、 $5\%$  時の結果を表 3、表 4 に示す。なお、最小遅延はすべての場合で 1 ms 未満であった。表 3、4 より、リダイレクト型は冗長データ追加型に比べて最大遅延、平均遅延ともに大きくなることが分かる。

表 3 ロス率  $p = 1\%$  時の遅延量 (ms)  
Table 3 Delay at loss ratio  $p = 1\%$  (ms).

	冗長データ追加型		リダイレクト型	
	最大遅延	平均遅延	最大遅延	平均遅延
$m = 5,$ $n = 10$	41	1	57	4
$m = 5,$ $n = 100$	420	2	510	170
$m = 10,$ $n = 10$	91	1	118	14
$m = 10,$ $n = 100$	903	4	1,010	431

表 4 ロス率  $p = 5\%$  時の遅延量 (ms)  
Table 4 Delay at loss ratio  $p = 5\%$  (ms).

	冗長データ追加型		リダイレクト型	
	最大遅延	平均遅延	最大遅延	平均遅延
$m = 5,$ $n = 10$	48	1	68	14
$m = 5,$ $n = 100$	424	8	516	250
$m = 10,$ $n = 10$	92	2	122	41
$m = 10,$ $n = 100$	913	14	1,025	554

表 5 ロス率  $p = 1\%$  時の他通信への影響  
Table 5 Influence on another communication at loss ratio  $p = 1\%$ .

	符号化 パラメータ	ロス率 (%)	最大遅延 (ms)	平均遅延 (ms)
FEC 装置なし	—	1.01	27.8	0.3
FEC 装置あり 冗長データ追加型	$m = 5,$ $n = 10$	1.02	27.8	0.3
	$m = 5,$ $n = 100$	1.00	27.6	0.3
	$m = 10,$ $n = 10$	1.00	27.9	0.3
	$m = 10,$ $n = 100$	1.00	27.8	0.3
FEC 装置あり リダイレクト型	$m = 5,$ $n = 10$	1.01	27.8	0.3
	$m = 5,$ $n = 100$	1.00	27.8	0.3
	$m = 10,$ $n = 10$	1.00	27.8	0.3
	$m = 10,$ $n = 100$	1.00	27.8	0.3

#### 4.2.3 他通信への影響

マルチキャストを 10 Mbps で転送している状態で、端末 A から B (同一筐体) へフレーム長 1,250 Byte の UDP パケットを 10 Mbps で送信し、FEC 装置の有無による他通信への影響を測定する。端末 B の測定結果を表 5 に示す。表 5 より、パケットロス率と遅延の点では、符号化のパラメータによらず、端末 A から B への通信にほとんど影響がないことが分かる。

また、各ルータの経路情報を確認し、FEC 装置導入前と導入後で端末 A, B 間の通信経路に変更がないことを確認した。以上より、提案 FEC 装置導入による他通信への影響はほぼないといえる\*1。

## 5. 考察

### 5.1 インライン型 FEC 装置の中継転送能力

3章で述べたように、提案 FEC 装置は対象マルチキャスト以外のパケットを中継転送する必要がない。一方、インライン型の FEC 装置では対象マルチキャスト以外のパケットも中継転送する必要があり、スループットボトルネックとなる可能性がある。本ボトルネックを定量的に評価するために、エンコーダと同じ装置にレイヤ 2 ブリッジ機能をユーザレベルで実装し、通信実験によりパケットロス率や遅延等の中継性能を測定した。評価条件を以下に示す。

- 2 台の通信端末は、4.2.2 項と同様に同一筐体の IXIA を用いて模擬し、UDP パケットを 100 万个送受信した。
- ブリッジ装置はエンコード処理を行わない。
- ブリッジ装置の 2 つの外部接続インターフェースは 32 bit 33 MHz (1 Gbps) の PCI バスを共有している。
- Linux カーネルのパラメータは以下を使用した。
  - `net.core.netdev_max_backlog = 300`
  - `net.core.[rw]mem_default = 107,520`
  - `net.core.[rw]mem_max = 131,071`

送信速度とフレーム長をパラメータとした測定結果を図 10, 図 11 に示す。フレーム長が 100 Byte のときは送信速度 200 Mbps 程度でパケットロスが 0.01% 程度発生した。また、top により別実験として測定した CPU 使用率は 96.2% であった。これらの結果から、同一送信速度であってもフレーム長が短い場合は、処理すべきパケット数が多くなり、ブリッジ処理能力の上限に達したと考えられる。一方、フレーム長 1,250 Byte では 500 Mbps でロスが発生したが、CPU 使用率は 23.7% であった。よってこの場合は、1 Gbps の PCI バスを送受信で共有していることが主要な原因であると考えられる。また、大量にロスが発生する環境では 3 ms 程度の付加遅延も発生した。なお、Linux カーネルのパラメータを大きくした場合でも性能の変化はほとんど見られなかった。

近年の固定網ではエッジ付近の回線も GbE (Giga-

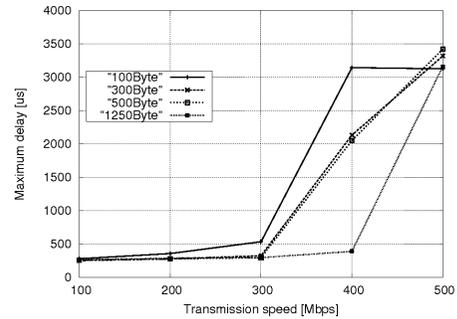


図 10 中継転送時の最大遅延

Fig. 10 Maximum delay of bridge.

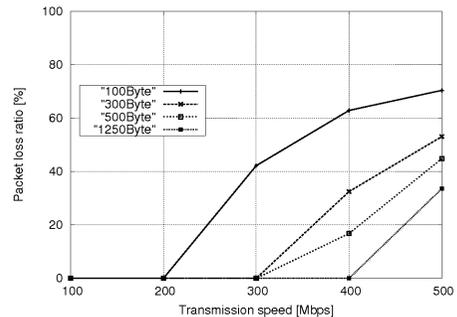


図 11 中継転送時のパケットロス率

Fig. 11 Packet loss ratio of bridge.

bit Ethernet) を利用することが多く、コアだけでなくエッジ付近に設置されたエンコーダであってもインライン型の場合はギガビットレートでの中継転送能力は求められる。一方、上記結果から明らかなように、CPU 処理能力や PCI バス帯域等の制約により安価な実現は困難である。これに対して、提案する非インライン型手法は対象マルチキャストのみ中継可能な処理性能を備えればよく、低コストな装置実装が可能である。

### 5.2 誤り訂正能力

パリティ符号の復号失敗率を検討する。 $m$  個のメディアパケットの中で  $k$  個のロスが発生する確率  $g(m, k)$  は  ${}_m C_k \cdot p^k (1-p)^{m-k}$  である。そのため、その平均ロス個数  $E_l(m)$  は  $\sum_{k=1}^m k \cdot g(m, k) = mp$  となり、1 パケットあたりに換算すると、 $E_l(m)/m = p$  となる。一方、パリティ符号使用時は、デコーダにおいてパリティパケットを受信できれば (確率  $1-p$  で発生)、同列内の  $m$  個のメディアパケットのうち 1 個のロスは復元可能である。したがって、 $k = 1$  個のロスが発生しても復号後のロス個数は 0 となり、復号後の平均個数  $E'(m|reception)$  は  $\sum_{k=2}^m k \cdot g(m, k)$ 、復号失敗率は  $E'(m|reception)/m$  となる。一方、パリティパ

\*1 ただし、狭帯域な環境では冗長データの増加によってパケットロス率に影響を与える可能性がある (インライン型も同様)。

ケットをロスした場合（確率  $p$  で発生），復号後のロス個数  $E''(m|loss)$  は  $E_l(m)$  と同じになる．以上より，パリティ符号の復号失敗率は，

$$\begin{aligned}
 & \text{(復号失敗率)} \\
 &= (1-p) \cdot (\text{パリティパケット受信時の復号失敗率}) \\
 & \quad + p \cdot (\text{パリティパケットロス時の復号失敗率}) \\
 &= (1-p)E'(m|reception)/m + pE''(m|loss)/m \\
 &= (1-p)\{E_l(m) - 1 \cdot g(m, 1)\}/m + pE_l(m)/m \\
 &= E_l(m)/m - (1-p) \cdot g(m, 1)/m \\
 &= p\{1 - (1-p)^m\} \tag{4}
 \end{aligned}$$

となる．式 (4) に各パラメータを代入することで表 2 と近い値を得ることができる．

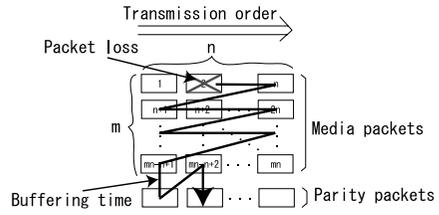
また，表 2 より， $m$  が小さな値ほど（冗長化率が大きいほど）優れた訂正能力を得ることを確認した．本評価では特定区間において順序逆転がないランダムロスのみを前提としたため， $n, d$  によって誤り訂正能力は変わらない．なお，非インライン型 FEC 装置だけではパケットロス率を十分に小さくすることができない場合も考えられるが，本稿では，対象マルチキャスト自体にクライアント・ソース間の別の FEC が適用されていることが前提となっている．そのため，非インライン型 FEC 装置は，対象区間のパケットロス率を当該別の FEC の復元可能範囲まで下げられれば十分である．

### 5.3 遅延

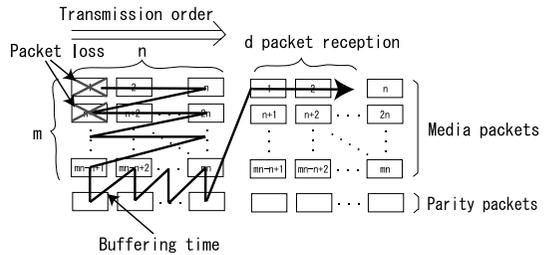
FEC 装置による遅延の大部分は，デコーダにおけるバッファリング時間である．これは，ロスパケットを復元するために，デコーダがロスパケットと同じ符号化列のメディアパケットとパリティパケットを受信しなければならないことに起因する．

冗長データ追加型において，この受信待ち時間が最大となるのは，符号化の  $m \times n$  行列の 1 行目のメディアパケットがロスした場合であり，その値はおおよそ  $(m-1)n$  個のメディアパケットの送信時間となる<sup>\*1</sup>（図 12 (a)）．メディアパケットの送信速度が  $t$ （パケット/秒）であれば，デコーダでのバッファリングによる最大遅延は約  $(m-1)n/t$  秒である．

たとえば， $t = 1,000$ （10 Mbps 時）， $m = 5$ ， $n = 100$  とすると，最大遅延増加量は約 400 ms となる．なお，表 3, 4 は 100 万パケット送信時の最大値であるため，大きな  $p$  ほどデコーダにおけるバッファリング時間が  $(m-1)n/t$  に近い値となる確率が高くなる．



(a) Parity injection type.



(b) Redirection type.

図 12 デコーダの最大バッファリング時間

Fig. 12 Maximum buffering time in decoder.

一方，リダイレクト型では， $m \times n$  行列の 1 行 1 列に対応するメディアパケットがロスし復元できない場合に最大遅延が発生する．この場合，デコーダでは，同一データブロックの範囲内であれば到達順序を保証しなければならないため， $mn$  個のメディアパケット（および  $n$  個のパリティパケットと最後に  $d$  個分のメディアパケット）の受信を待たなければならない（図 12 (b)）．したがって，デコーダでのバッファリングによる最大遅延は約  $(mn + d)/t$  秒<sup>\*2</sup>である．たとえば， $t = 1,000$ （10 Mbps 時）， $m = 5$ ， $n = 100$ ， $d = 7$  とすると，最大遅延増加量は約 507 ms となる．

冗長データ追加型の実験結果では，FEC 装置を経由しないマルチキャストパケットの遅延はほぼ 0 であったことから，復元パケットの遅延量が到達順序逆転やネットワークジッタの大きさに対応する．クライアントの実装によっては遅れて到着した復元パケットが破棄される可能性もある．ただし 4.2.2 項で述べたとおり，提案 FEC 装置の導入に際しては，クライアント・ソース間で別に適用される FEC やクライアントで動作するアプリケーションでのバッファリング処理を想定しており，ある程度の順序逆転やジッタは対応可能である．したがって，冗長データ追加型では，クライアントで対応可能な順序逆転やジッタの大きさを考慮して，メディアパケットの送信速度  $t$  に応じた FEC

\*1 エンコーダにおいてパリティパケットは生成後即座に送信されるものとする．

\*2 パケットロス率  $p$  が小さいとき， $d/(1-p) \doteq d$  を利用した．

のパラメータ ( $m, n$ ) を設定する必要がある。

一方、リダイレクト型では到着順序は保証されることが特長である。また、今回の実装では省略しているが、リダイレクト型ではデコーダから送信する際にバッファリングを行えるため、送信間隔を整えることでジッタを吸収可能である。すなわち、設定した符号化パラメータにより最大遅延が増加しても、ジッタ吸収機能によってジッタ幅を小さくすることができる。ただし、増加する遅延量がエンド・エンド（サービス）の遅延要求量以内に収まるように、FEC のパラメータ ( $m, n$ ) を設定する必要がある。

#### 5.4 障害時の動作

冗長データ追加型では対象マルチキャストおよび通常のユニキャスト通信がそのまま特定区間に流れる。そのため、エンコーダもしくはデコーダに障害が発生した場合も、冗長データの転送が停止するのみであり、対象マルチキャスト自身や通常のユニキャスト通信は継続して行うことができる。一方、リダイレクト型では、障害時に対象マルチキャストやマルチキャストソース宛でのユニキャスト通信の転送経路を、エンコーダ・デコーダを経由しない通常の経路に戻す必要がある。具体的には、デコーダからソース  $S$  宛での経路が消失したことを OSPF で広告するか、デコーダ・ルータ間の OSPF 隣接を解消する等の処理が要求される。デコーダ障害時は前述のいずれかの方法が適用できるが、エンコーダ障害時にはデコーダがこれを検出する必要がある。検出方法としては、エンコーダから死活監視のための試験パケットをマルチキャストアドレス  $G'$  を用いてデコーダへ定期的送信する方法等が考えられる。可用性の観点からは、装置障害時に OSPF の経路更新にともなう配信断が発生しうるリダイレクト型よりも、配信が継続する冗長データ追加型が優れているといえる。

#### 5.5 提案 FEC 装置の比較

表 6 に本稿で提案した冗長データ追加型とリダイレクト型の FEC 装置の特徴をまとめる。リダイレクト型では、使用する誤り訂正符号に制限はない。一方、冗長データ追加型では元のマルチキャストをそのまま特定区間に流すので、符号語に元のデータがそのまま含まれる符号に限定される。誤り訂正能力については使用する誤り訂正符号に依存するため、同一符号であれば同じである。最大遅延についても符号・実装依存であるが、順序逆転に対応するためにリダイレクト型が不利となるものの大きな差は生じない。

冗長データ追加型の大きなデメリットとしては、デコーダから送信される復号パケットが遅れることに

表 6 冗長データ追加型とリダイレクト型の比較

Table 6 Comparison of parity injection type with redirection type.

	冗長データ追加型	リダイレクト型
使用可能な誤り訂正符号	制限あり (そのときの最大遅延にも制限あり)	制限なし (インライン型と同様のものが使用可)
誤り訂正能力	同じ (使用する誤り訂正符号に依存)	
最大遅延	冗長データ追加型 $\leq$ リダイレクト型 (使用する誤り訂正符号と実装に依存)	
経路変更	不要	必要
FEC 装置の障害時	影響なし	経路を元に戻す必要がある
クライアントでの到達順序逆転	発生する (クライアントでの整列が必要)	発生しない (ような実装が可能)

よってクライアント側でマルチキャストパケットの到達順序が逆転することである。すなわち、元々マルチキャストに適用されている FEC の復号処理やクライアント側でのアプリケーションが到達順序逆転を許容できない場合、冗長データ追加型の適用は困難となる。また、許容できる場合についても、最大遅延がほぼそのままネットワークジッタに対応するため、元の FEC やアプリケーションのバッファリング範囲内にジッタが収まるように、冗長データ追加型の符号ならびにそのパラメータを決めなければならない。これに対してリダイレクト型ではジッタ吸収が可能である (5.3 節参照)。

他方、リダイレクト型のデメリットとしては、ソース  $S$  向けの経路変更に関わる経路制御コストの増加があげられる。また、FEC 装置障害時には変更した経路を元に戻す必要があるが、OSPF のタイムアウトを利用した場合は障害回復までに数十秒程度の時間を要してしまうことも問題となる。

したがって、経路変更が運用上許容できない場合や耐障害性を重視する場合は、冗長データ追加型が優れているといえる。一方、提案 FEC 装置の設置主体がネットワーク運用者であるとの前提に立てば、クライアント側の FEC やアプリケーションにおける到達順序逆転対応やバッファリング範囲を考慮する必要のないリダイレクト型の適用が容易であり適用範囲も広い。

## 6. おわりに

本稿では、特定区間におけるマルチキャスト通信の品質向上を可能とする非インライン型の FEC 装置を提案した。本 FEC 装置では、マルチキャストの複製転送機能や経路制御機能を応用し、対象となるマルチキャストのみを特定区間の両端のルータに付加接続し

たエンコーダ、デコーダ経由で転送させ高信頼化する。これにより、一部の低信頼区間を高信頼化しエンド・エンドでの FEC 冗長化率を抑制できるため、様々なコスト・品質の回線が混在する環境に対して有効である。また、対象マルチキャスト以外の通信は、原則としてエンコーダとデコーダを経由しないため、可用性・転送性能の面で FEC 装置がボトルネックにならない。

試作機を用いて、2 種類の提案方式の動作確認と性能評価を行い、理論値に近い誤り訂正能力を確認した。また、使用する符号化行列のサイズに応じて、冗長データ追加型では復元パケットの到達順序逆転が大きくなり、他方、リダイレクト型ではパケット到達順序を保証するためのバッファリング機能によって遅延が大きくなった。

提案 FEC 装置の特徴としては、冗長データ追加型が耐障害性に優れているのに対し、リダイレクト型はエンド・エンドでの FEC やクライアントでのアプリケーションに依存せず柔軟に適用できる点が優れている。実運用では、これら特徴を考慮し適用方式を選択するのが望ましい。

謝辞 日頃ご指導いただく KDDI 研究所秋葉所長に感謝する。

### 参 考 文 献

- 1) KDDI: MOVIE SPLASH.  
<http://www.hikari-one.com/tv/index.html>
- 2) KDDI: EZ チャンネルプラス.  
[http://www.au.kddi.com/ezweb/service/ez\\_channel\\_plus/index.html](http://www.au.kddi.com/ezweb/service/ez_channel_plus/index.html)
- 3) Floyd, S., Jacobson, V., McCanne, S., Liu, C. and Zhang, L.: A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing, *IEEE/ACM Trans. Netw.*, Vol.5, No.6, pp.784–803 (1997).
- 4) Nonnenmacher, J., Biersach, E. and Towsley, D.: Parity-based loss recovery for reliable multicast transmission, *Proc. ACM SIGCOMM'97*, Cannes, France, pp.289–299 (Sep. 1997).
- 5) Forward error correction (FEC) building block, IETF RFC3542 (Dec. 2002).
- 6) The use of forward error correction in reliable multicast, IETF RFC3543 (Dec. 2002).
- 7) An RTP Payload Format for Generic Forward Error Correction, IETF RFC2733 (Dec. 1999).
- 8) RTP: A Transport Protocol for Real-Time Applications, IETF RFC1889 (Jan. 1996).
- 9) 宮坂, 吉村, 鎌田, 照日, 植松: 無中断輻輳回避制御方式の有効性について, *信学技報*, Vol.104, No.433, NS2004-141, pp.31–34 (Nov. 2004).
- 10) OSPF Version 2, IETF RFC2328 (Apr. 1998).

- 11) Protocol Independent Multicast – Sparse Mode (PIM-SM): Protocol Specification, IETF RFC2362 (June 1998).
- 12) Internet Group Management Protocol, Version 2, IETF RFC2236 (Nov. 1997).
- 13) Cisco Systems: Catalyst 3560 スイッチソフトウェアコンフィギュレーションガイド Cisco IOS Release 12.2(25)SEE(OL-8553-01-J) (2006).
- 14) Gallager, R.G.: Low-density parity check codes, *IRE Trans. Theory*, IT-8, pp.21–28 (1962).
- 15) IP in IP Tunneling, IETF RFC1853 (Oct. 1995).
- 16) IXIA. <http://www.ixiacom.com>
- 17) Rizzo, L.: Dummynet: A simple approach to the evaluation of network protocols, *ACM Computer Communication Review*, Vol.27, No.1, pp.31–41 (Jan. 1997).

(平成 19 年 5 月 17 日受付)

(平成 19 年 11 月 6 日採録)



佐々木 力 (正会員)

平成 16 年東京工業大学大学院理工学研究科集積システム専攻修士課程修了。同年 KDDI (株) 入社。QoS, マルチキャストの研究に従事。現在, (株) KDDI 研究所 IP 品質制御システムグループ研究員。



田上 敦士 (正会員)

平成 9 年九州大学大学院システム情報科学研究科知能システム学専攻修士課程修了。同年 KDDI (株) 入社。以来, 研究所にて, 高速通信プロトコル, オーバーレイネットワークに関する研究に従事。現在 (株) KDDI 研究所 IP 品質制御システムグループ研究主査。



長谷川輝之 (正会員)

平成 5 年京都大学大学院修士課程修了。同年 KDDI (株) 入社。以来, 研究所にて, 高速通信プロトコル, 次世代インターネットの研究に従事。現在 (株) KDDI 研究所 IP 品質制御システムグループ主任研究員。平 15 年度電波産業会電波功績賞受賞。



阿野 茂浩（正会員）

平成元年早稲田大学大学院修士課程修了．同年 KDD（株）入社．以来，研究所にて，ATM 交換方式，IP ネットワーク管理・制御，次世代インターネットの研究に従事．現在，（株）KDDI 研究所 IP 品質制御システムグループリーダー．平成 7 年度情報処理学会学術奨励賞受賞．

---