PAPER

Reliable IP Multicast Communication Over ATM Networks Using Forward Error Correction Policy

Hiroshi ESAKI†, Takeko FUKUDA††*Member*

SUMMARY This paper discusses and evaluates an effect of cell level FEC (Forward Error Correction) capability on error-free (i.e. reliable) IP multicast service over ATM networks. In the error-free IP multicast service, every receiver is delivered IP packet from the sender synchronously.

Without applying the FEC policy, the expected IP packet error/loss probability becomes large, when the number of multicast receivers is large. For example, when the cell error/loss probability of each ATM data-link segment is 10^{-6} and the number of receivers is 10^3, the IP packet error/loss probability observed at the sender is about 0.5, which means that about 50% of IP packet sent from the sender will be subject to retransmission. One possible solution would be using the intermediate multicast-TCP entities, that terminate TCP protocol, among the sender and the receivers. However, this approach requires the additional entities within the network and can not provide the ordered message delivery for a multipoint-to-multipoint communication.

On the contrary, with applying the FEC policy, the expected IP packet error/loss probability is dramatically reduced. Therefore, an error-free IP multicast service can be provided with a simple architecture, even when the number of multicast receiver is large, e.g. 10^5. For example, when the cell error/loss probability of each ATM data-link segment is 10^{-6}, the packet error/loss probability observed at the sender is less than 10^{-2} even for 10^6 receivers. Finally, even when the cell error/loss probability of ATM data-link segment is large, e.g. 10^{-3}, the IP multicast service without the FEC policy can not apply even for 10 receivers. However, the IP multicast with the FEC policy can apply upto few hundred of receivers.

key words: ATM, multicast, FEC, reliable data transmission

1. Introduction

ATM, Asynchronous Transfer Mode, networks will be the high speed communication platform both for B-ISDN and for high speed LAN/WAN [1][2]. The interface speed to be provided by ATM networks to the user interface will be from dozens Mbps to few Gbps. An ATM network handles any user information through a unique fixed sized packet, called cell. It is said that the ATM network can provide flexible communication channel for a wide variety of communication requests, e.g. a real-time video communication. ATM network is a switched star configurated network, rather a bus or ring based network (e.g. Ethernet and FDDI). In the bus or ring based network, every information (L2-SDU ; Layer 2 Service Data-Unit) is delivered to every SAP (Service Access Point). In other word, the delivery of L2-SDU is performed using a broadcast capability of data-link layer. Therefore, in bus or ring based networks, the bandwidth resource will be the bottle-neck of system throughput for scaling up. On the contrary, in ATM networks, the L2-SDU submitted to the network is selectively delivered to a single or multiple SAPs. Therefore, it is said that ATM network uses L2 bandwidth resource efficiently. And, it is said that ATM network is easily scaled up because the bandwidth resource will not be the bottle-neck of a system throughput.

Conventional communications in telephone networks and computer networks could be said a point-to-point communication. But, recently, the requirement of multicast communication, point-to-multipoint (pt-mpt) and multipoint-to-multipoint (mpt-mpt), has increased [9]. For example, in the Internet, the M-Bone (Multicast Backbone) experiment has been progressed and the real-time multicast communication, i.e. video and voice, has been provided. On the contrary, the reliable multicast (mpt-mpt) communication (i.e. ordered, reliable, atomic multicast) is definitely required by a distributed computation (e.g. distributed database system) [11][14][15].

At this time, the large scale multicast communication over the Internet is usually allowing some error or loss of transmitted IP packets, and the error-free IP packet delivery to every SAPs would not be required (i.e. UDP over IP for IP multicast service). This is because the current applications using a large scale IP multicast are voice and video communications. However, in the future, the other applications (e.g. multicast service of digital media such as a large scale interactive transaction game over the Internet) will require the *"large scale"* error-free IP packet delivery for a multicast service. Digital media generally requires an error-free data transmission. Therefore, when the large scale multicast service for such a reliable digital media is required, this kind of service must be served on the IP multicast channel, that may be operated with a resource reserved path provided by some protocol (e.g. RSVP). It may be possible to serve such a service through the individual pt-pt communications. However, this approach causes extremely large latency of

Manuscript received April 24, 1995.
Manuscript revised June 26, 1995.
†The author is with R&D Center, TOSHIBA CORPORATION, Kawasaki-shi, 210 Japan.
††The author is with Electronics & Telecommunications Systems Group, TOSHIBA Corporation, Minatoku, 105-01 Japan.

data delivery and requires huge computational capability for the source entity, when the number of receivers is large.

ATM network will provide a multicast channel (pt-mpt and mpt-mpt), as well as a unicast channel (pt-pt). Actually, the signaling protocol defined in ATM Forum UNI specification version 4.0 shall include the pt-mpt communication channel establishment capability.

This paper focuses on the provision of an error-free IP packet delivery for multicast service over ATM networks, and evaluates the performance of forward error correction (FEC) policy regarding the IP multicast service. Section 2 discusses the issues of error-free (i.e. reliable) multicast and related works. Section 3 discusses the protocol structure of FEC-SSCS (FEC-Service Specific Convergence Sub-layer) for AAL Type 5. Section 4 discusses the IP multicast over ATM and evaluates the effect of cell level FEC policy by the numerical evaluation. Section 5 gives a brief summary.

2. Issues of Reliable Multicast and Related Work

Conventionally, the reliable multicast service was required by some distributed computation environments (e.g. distributed database system). The number of receivers for these applications would not be large, let say it would be decades of receivers.

The challenges regarding a reliable multicast in computer networks would be the following three items.

1. How to provide ordered and atomic multicast
   Ordering of message delivery from multiple senders would be sometimes required.

2. How to provide reliable (i.e. error-free) multicast
   In order to provide an error-free data delivery, some error control policies (i.e. error detection and error recovery) must be applied.

3. How to provide large scale multicast
   The control policies must scale regarding both the large number of receivers and the physically widespread senders and receivers.

This paper focuses on items 2 and 3, i.e. how to provide large scale error-free multicast service. The remainder of this section discusses above three items, briefly.

2.1 Ordered and Atomic Multicast

The processes in a multicast group (i.e. senders and receivers) share the same sequence of messages. The messages may be originated at one or more sender processes and may be processed at one or more destination processes. Some applications require that the protocol provides some guarantees on the order of message delivery. Lack of the ordered message delivery may cause data's inconsistency or dead-lock status. A multicast protocol must responsible to these (some degree of) ordered message delivery, as well as an error-free message delivery.

In general, there are the following three levels of ordering properties.

1. Single source ordering
   When the messages are issued from the same source process to the same multicast address, all destination processes receive these messages in the same relative order originated by the source process.

2. Multiple source ordering
   Even when the messages are issued from the different source processes to the same multicast address, all destination processes receive these messages in the same relative order originated by the source processes.

3. Multiple group ordering
   Even when the message are issued to the different multicast address from the different source processes, all destination processes receive these messages in the same relative order.

There are many approaches to provide the above properties : e.g. (a) time-stamping [12][13], (b) centralized token [11][15], (c) distributed two-phase commitment [14], (d) propagation graph algorithm [16]. These approaches are applicable to the architecture discussed in this paper. This means that the technique discussed in this paper can adopt some mechanism to provide the ordered message delivery.

2.2 Error-Free Multicast

For some applications, the message issued by the source process must be delivered to all destination processes without any error. Error detection an error correction are the key functionalities to provide an error-free data delivery. Each layer includes some level of error control functionalities. For example, TCP does have both an error detection (by checksum) and an error correction (by go-back-N retransmission), but IP does only have an error-detection. There are many error-detection and error correction schemes. Below, the existing error correction techniques are described briefly.

- Go-Back-N [4]
  IP packets in it's transmission window are consecutively transmitted from the sender process without any acknowledgement (positive or negative). When some packet is errored or lost, all the IP packets after the very errored or lost IP packet in

the transmission window are subject to retransmission. That means that, even when the IP packets after the errored or lost IP packet are correctly delivered to the destination process, these packets must be retransmitted.

- Selective Repeat [11]
  IP packets, that are errored or lost, are selectively retransmitted. Since the order of transmitted IP packets from the sender process will not be maintained at the destination process, the destination process must re-order the received IP packets to provide the ordered data delivery to the upper layer application interface.

- (FEC) Forward Error Correction [7]
  Additional information (or packet) is attached to the transmitting information to recover the errored or lost character(s) without data retransmission. Error recovery could be both the bit(s) correction and the block correction. When the errored data is beyond the correction capability of the applied FEC algorithm, the data (e.g. IP packet) must be retransmitted either by go-back-n or selective repeat policy. Though the FEC always requires some overhead to transmit the data, the data error probability observed at destination process can be reduced. Therefore, the probability of data retransmission can be reduced.

In order to perform the IP packet retransmission at the sender process, some feed-back signal (acknowledgement) from the destination process is required. There are basically two approaches. One is the positive ACK, and the other is the negative ACK (NACK). In the positive ACK approach, the destination process always sends back the ACK signal to the sender process while it receives IP packet correctly. The errored or lost IP packets are recognized by the sender process both(either) by the lack of the ACK signal from the destination process and(or) by the explicit control signal issued by the destination process. With TCP or major data transmission protocols, the positive ACK approach is applied, because of easy implementation. On the contrary, in the negative ACK approach, the destination process sends back the NACK signal to the seder process, only when the destination process detects the error or loss of transmitted IP packet(s) from the sender process. Though the control signal between the sender and the destination process is reduced, the method to guarantee the error-free data delivery must be a little bit tricky. Therefore, applying the NACK approach in the actual applications is relatively fewer than applying the positive ACK approach.

2.3 Large Scale Multicast

For multicast service, the scaling is one of the important aspects to solve. There are two aspects for the scaling. One is for the large number of receivers (and senders), and the other is for the geographically widespreaded senders and receivers. The later one would be regarding the timer design for the feedback loop to recognize error (this is not touched in this paper). And, the former one would be regarding both the implementation complexity at the sender's control process and the actual data transmission performance. This paper focuses on the former issue, rather than on the later issue.

When the number of destination process becomes large, the following three issues occur.

(a) Increase of Protocol State Information

When the positive ACK approach is applied, the sender process must maintain all of protocol state information for each destination processes. That means that the maintained protocol state information at the sender process will be *order of* N. Here, N is the number of receivers.

One feasible approach to reduce the maintained protocol state information at the sender process would be using the intermediate layer 4 (e.g. modified TCP for multicast) entities. Between the sender and the receivers, there are the intermediate routers that act as the branching point of the multicast tree. In order to reduce the protocol status information maintained at the sender, the intermediate routers could terminate the layer 4 protocol. Multiple acknowledge packets from the destination process to the sender process are merged into a single acknowledge packet at the intermediate router. By the above approach, the protocol state information that should be maintained at a sender can be distributed among the intermediate routers, and it will be the *order of* $log_m N$. Here, m is the number of multicast leafs from the intermediate routers (or the sender process). However, every intermediate router must perform layer 4 protocols for every multicast service, and it must keep many IP multicast packets until the IP multicast packets are correctly delivered to the leaf entities on the down-steam toward the destination processes.

Another approach is proposed in RFC1301 (i.e. MTP : Multicast Transport Protocol). MTP uses the NACK approach with some smart technique (i.e. sending a referenced protocol state information from a special process) [11]. Using this technique, the maintained protocol state information in the sender process could be *order of constant* $0(c)$. However, this approach could not be applied to the case where the receivers are of large number and are geographically widespreaded, because too many control packets (NACK packets) will be transferred to the sender from the receivers as briefly indicated in the following item.

(b) Increase of Control Packets from Receivers to Sender

Control packets (ACK/NACK) will be transferred from the receivers to the sender, as well as from the sender to the receivers. The control packets from the sender to the receivers could be done by the multicast channel. On the other hand, the control packets from the receivers to the sender will be issued by each receiver toward the the sender. This means that the sender would receive many number of the control packets (ACK/NACK) from the receivers. The required bandwidth for the control packets from the receivers and the required processing power for the control packet will be large in a large scale multicast.

Also, when the number of received control packet is large, the sender process will experience so frequent interruptions to deal with the control packets. The interruption due to the reception of the control packet invokes a context switching, which is costly operation in the computer system. As a result, so many control packets reception at the sender process will result to reduce the sender's processing performance.

Here, the NACK approach, that is adopted in MTP of RFC1301, can reduce the average amount of control packets. However, in the worst case where all receivers could not correctly receive IP packets, all receivers will transfer the control packets to the sender. Therefore, the largest number of control packets from the receivers to the sender will be the same number as the case where the positive ACK approach (e.g. TCP) is adopted. However, the NACK approach could reduce the probability that so many control packets will be sent back to the sender, compared to the ACK approach.

(c) Increase of Observed Packet Error/Loss Probability

An IP packet issued by the sender process is multicasted to the large number of receivers in a large scale multicast. Therefore, the observed (could be said as accumulated) IP packet error or loss probability at the sender process increases approximately in linear, according to the number of destination processes. The observed IP packet error or loss probability at the sender process would be *order of* $N \times d$. Here, d is the diameter of datalink segments in the multicast tree. When the approach using the layer 4 entities in intermediate routers is applied, it could be reduced to *order of* $m \times d$. But, since this approach requires the modification of routers, it would not be a good approach.

This issue, i.e. the observed packet loss/error probability will increase for the large number of receivers, is essential and has never touched before. The focus of this paper is to solve this issue.

(d) Management of multicast members

When there are large number of receivers associated with a multicast service, it will be hard to manage all receivers at the sender. Therefore, a large scale multicast needs to apply the receiver oriented control for joining/quitting a multicast service and needs to apply the distributed receiver management policy, e.g.[10]. The receiver will send a request message to the local multicast service management entity (maybe located at the branching point in the multicast tree), in order to join to (or quit from) the multicast service.

(e) Receiver's un-reliable behavior

Another issue for a large scale multicast service is the un-reliable behavior of multicast receivers. This is very serious for a reliable multicast service. When the number of receivers in the given multicast connection is large, some receivers will suddenly quit from a multicast service without any indication to the multicast service management entity. By this un-reliable behavior by the receiver, the sender (or the intermediate management entities) may interpret the receiver did not correctly receive the packet(s) with the positive ACK policy, even though it quitted from the multicast service.

(f) Service quality assurance

For a geographically large scale reliable multicast service, it will be hard to provide the sufficient data-link/network service quality along the multicast tree, with the conventional best effort service. Here, in the best effort service, there is no guarantee that the data (e.g. IP packet) delivery is performed with some quality of service (e.g. delay of data delivery or data loss probability). In the best effort service, the data may experience a substantial bufferring delay or may be discarded due to buffer overflow at the network entities (e.g. at the router). This is because, for the large scale multicast service, the mutlicast tree has the large number of branching points and links. Even when only one of entity within a large multicast tree offers a poor service quality, the service quality (e.g.throughput) is degraded due to the entity offering a poor service quality.

The likely solution proposed in the paper against the above issues is as follows.

1. Reduction of protocol state information
   Protocol state information should be maintained at each branch point entities (maybe branch point routers) along the multicast tree. The branching point entities need not buffer the IP packets, but just maintain the protocol state and merge the control packets. The control packets from the downstream toward the sender should be merged into one control packet, that is transferred toward the sender (i.e. toward up-stream). The method of the protocol state maintenance at the branch point entities is fulfilled based on the soft-state state maintenance technique with the NACK control message, which is similar approach as RFC1301. Here, the soft state state maintenance means that the protocol state is checked and refreshed only when there is received control message. If there is no control message, no-action is fulfilled and it is assumed that the receiving entity, which does not send back the control message (i.e.NACK message), correctly receives the packets or quits from the service.

   When the fan-out (number of leaves that branching point has) at the branching point entities is m, the protocol state information maintained in the branching point entity will be *order of m*.

2. Reduction of control packets
   The branching point entities merge the control packets. The control packets from the downstream toward the sender should be merged into one control packet, that is transferred toward the sender (i.e. toward up-stream).

3. Reduction of observed packet loss/error probability
   The end-to-end based FEC (Forward Error Correction) policy in ATM cell level is applied. As evaluated in section 4, it is expected that the observed packet loss/error probability can be significantly reduced by applying the cell level FEC policy.

4. Receiver oriented member management
   In order to join to (or quit from) the multicast service, each receiver will send a request message to the local multicast service menagement entity that may be located at the branching point of the multicast tree.

5. NACK (negative ACK) control policy
   In order to work well even if receivers quit from the multicast service without any quitting indication to the multicast service management entity, the negative ACK policy (i.e. NACK policy) is applied to so as to get the packet reception status of the receivers. In the NACK policy, when the NACK packet is not received, it is assumed that the end-host receives the packet correctly. As a result, the end-host without sending a quit indication can be assumed that it always receives packets correctly.

6. Resource reservation oriented protocol (e.g.RSVP)
   In order to provide a sufficient service quality for the multicast service, some resource reservation oriented protocol (e.g.RSVP) is applied to the reliable multicast service. With the resource reservation oriented protocol, it can be expected that some level of service quality (i.e. cell loss ratio and delay quality) over the multicast tree will be provided.

3. Forward Error Correction Protocol in ATM Networks

3.1 Error Control Capability in ATM Networks

The purpose of error control in data networks is to provide an error free data-unit delivery between a pair of SAPs. Error detection and error correction are the key functionalities of the error control. For example, TCP has both functionalities, however, both ATM network and IP fundamentally have only error detection capability.

The performance matrices of error control should be throughput and latency of data delivery. As discussed in [7][8], the conventional error control scheme, e.g. go-back-N, can not yield a good performance for the point-to-point wide area high speed communication service. Also, as evaluated in this paper, when an error-free IP multicast service is provided using the conventional architecture (i.e. without FEC policy), the throughput of data delivery will be suddenly degraded for the large scale multicast service.

This section discusses the error control scheme based on the FEC scheme in SSCS of AAL Type 5 in order to provide a high throughput IP multicast packet delivery. Here, since the discussed scheme is an SSCS, which is below an IP layer, it is basically unnecessary to modify the existing software of end-station. Additionally, a new SSCS (called in this paper as FEC-SSCS) should be implemented in the AAL layer.

Table 1 shows the error detection and the error correction capability below CPCS (Common Part Convergence Sub-layer) in ATM networks. As shown, the error(s) both for header field and for payload are detected [2][3]. On the other hand, the error for payload (CPCS-SDU) is not corrected, excluding the option of AAL Type 1. SSCS of AAL could have an error correction capability for the assured mode operation [3]. However, at this time, neither ITU-T nor ATM Forum has been discussed the detailed protocol specification

Table 1 Error Detection and Correction Capability in ATM Networks

		Error Detection		Error Correction	
		Payload(SDU)	Header	Payload (SDU)	Header
PHY		No	Yes(by HEC)	No	1 bit(ATM header)
ATM		No	Yes(by HEC)	No	1 bit
SAR	Type 1	No	Yes(by SNP)	No	1 bit (SN)
SAR	Type 3/4	Yes(CRC-10)	Yes(CRC-10)	No	No
SAR	Type 5	No	No	No	No
CPCS	Type 1	No(optional)	No(optional)	No(optional)	No(optional)
CPCS	Type 3/4	Yes(LI for lost)	No	No	No
CPCS	Type 5	Yes(CRC-32/LI)	Yes(CRC-32)	No	No

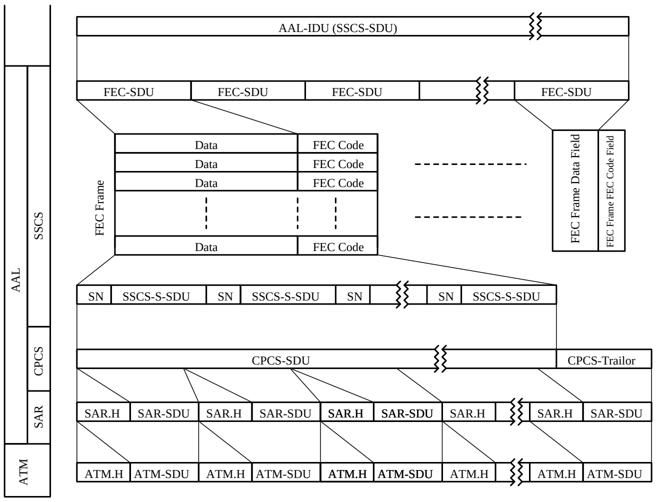


Fig. 1 Protocol Structure Applying FEC-SSCS

of error correction functionality in SSCS.

The error control is necessary for data delivery. On the contrary, for the voice or video communication, the error control may not be necessary. This is very true for the current IP multicast service over the Internet. Regarding the data delivery service, AAL Type 5 will be used for many cases. Since AAL Type 5 does not have an error correction capability, the dropped or errored data-unit (CPCS-SDU) will be retransmitted by the request of the upper layer protocol (e.g. TCP).

3.2 SSCS Protocol for FEC Control (FEC-SSCS)

Figure 1 shows the proposed protocol structure. AAL type is AAL Type 5. Since the proposal is about the SSCS in AAL Type 5, only the processing sequence at SSCS is described below. The detailed FEC frame format is described in Appendix A, as an example.

SSCS-SDU (SSCS-Service Data-Unit), which corresponds to an AAL-IDU (AAL-Interface Data-Unit) may be equivalent to an IP packet, whose default MTU size would be 9,180 bytes in ATM networks [6]. SSCS-SDU will be divided into multiple FEC-SDUs. Each FEC-SDU is stored in the FEC Frame, and the FEC code is calculated to be attached, e.g. [3][7]. The writing order and reading order for the FEC frame are the same order, in order to avoid the unnecessary user data interleaving. When the data (SSCS-S-SDU) is read out from the FEC Frame, the FEC frame header field, containing the SN (sequence number), is inserted in order to identify which SSCS-S-SDU(s) is(are) dropped at the destination SSCS entity. Also, the FEC frame header field contains the CRC-10 check sequence, in order to detect the bit(s) error in the SSCS-S-SDU. The FEC-SSCS can transmit the FEC-SDU in pipe line, since the writing order and the reading order for the FEC frame are the same order. When the actual SSCS-SDU (i.e.AAL-IDU) or FEC-SDU is smaller than the size of user-data part in the FEC frame that is defined in the FEC calculation algorithm, the remaining field of the FEC frame's data part is not have to be sent.

The FEC-SSCS uses the Reed-Solomon error recovery algorithm of the block erasure mode [3], and does not apply the interleaved data transmission. The variable length of FEC frame (both user-data part and FEC code part) can be allowed for the efficient transmission of small sized SSCS-SDU. The FEC frame has a user-data part and an FEC code part. A user-data part containing the user information will be more than one cell, and an FEC code part will be one or more than one cells. The symbols in the FEC code part are generated by the generator polynomial using the user-data symbols. The error detection for the FEC frame is performed by the AAL5's CRC-32 error check sequence, and the error detection for each SSCS-S-SDU is performed by the CRC-10 error check sequence in the FEC frame header field. The CRC-10 error check sequence is adopted, in order to improve the error correction capability of the Reed-Solomon algorithm.

When the number of errored or dropped SSCS-S-PDUs is larger than the correction capability of the applied FEC, certain retransmission policy of SSCS-S-PDUs is fulfilled. For example, whole of FEC Frame will be retransmitted, or selected SSCS-S-PDUs will be retransmitted.

The size of FEC frame should be determined aligning to the IP packet length and should be determined based on the requirement of the end-to-end IP packet delivering latency. The larger FEC frame causes larger latency for FEC code calculation, when there is bit(s) error or cell(s) discarding in the received FEC frame. When there is no bit error nor cell loss in the received FEC frame, the FEC calculation at the receiver is unnecessary. The default MTU size in ATM networks will be 9,180 byte [6], which is close to one page of modern file system (i.g. 8 Kbyte). The possible and reasonable IP packet length will be 576 byte ([5]), 1 Kbyte, 4 Kbyte, 8 Kbyte, and 64 Kbyte. For the convenience, the following FEC frame sizes are picked up in the evaluation, i.e. 10 cells (for 576 byte), 160 cells (for 8 Kbyte) and 1330 cells (for 64 Kbyte).

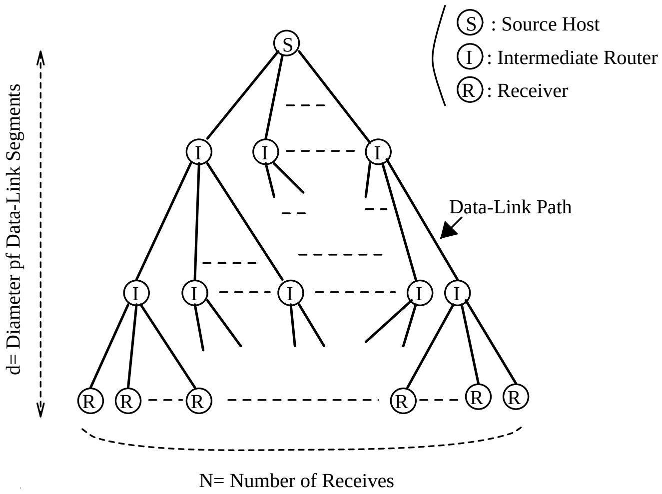


Fig. 2 Evaluation Model (IP Multicast)

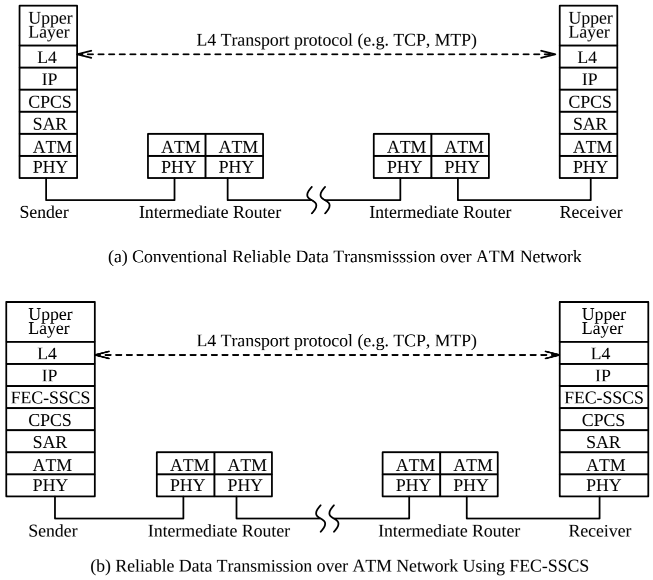


Fig. 3 Evaluation Models (IP Multicast)

4. IP Multicast over ATM Networks

4.1 Error-Free IP Multicast over ATM Networks

As discussed in the previous section, ATM protocol (below AAL) does not have the error-correction capability, basically. And, also, the current AAL Type 5, which will be used for data communication, does not have the error correction capability. Therefore, the protocol entities (e.g. TCP entity for pt-pt service) implemented at the end-station perform the error detection and the error correction procedure. When AAL can not provide an error-free packet delivery for the transport layer entity, the transport layer entity performs a retransmission procedure [4]. By the retransmission procedure, the error-free packet delivery is provided. The transport layer entity has the responsibility to the error-free packet delivery and, therefore, some transport layer protocol (e.g. TCP for pt-pt) is running between every source and destination end-station pair.

With the conventional AAL Type 5, the whole of IP packet will be dropped and it will be subject to retransmission to receive the correct IP packet, even when only one cell belonging to the IP packet is errored or dropped. Therefore, too large IP packet size will experience the large IP packet error or loss probability.

The transmission quality of user packets (i.e. excluding control packets) associated with the observed loss/error probability and the transmission overhead (i.e. additional information and retransmitted information) is evaluated in the following subsection. Figure 3 shows the evaluated protocol models. (a) is the conventional approach without FEC policy, (b) is the proposed approach applying the FEC policy in AAL-SSCS.

The transmission delay performance and the end-to-end throughput performance should be the good performance evaluation parameters, however these are not evaluated in this paper. It can be said that, when the observed IP packet loss/error probability becomes a sufficiently small value by the use of FEC control, the end-to-end transmission delay shall be small enough and the end-to-end throughput shall be large. This is because, when the observed IP packet loss/error probability is small, the probability of IP packet retransmission due to the error/loss of IP packet observed at the destination end-station shall be small. The IP packet retransmission procedure causes the increase of transmission delay, which is larger than the RTT (round trip time) between the source and destination end-station pair. Also, during the IP packet retransmission procedure, the source end-station can not transmit the packets that should be transmitted after the packets in the IP packet retransmission procedure. Due to the above reasons, the end-to-end transmission delay shall be large and the end-to-end throughput shall be degraded, when the observed IP packet loss/error probability is large. Therefore, in this paper, the evaluations of end-to-end transmission delay and of end-to-end throughput are not performed. The performance evaluation for these two parameters should be for further study.

4.2 Performance Evaluation of IP Multicast over ATM Networks

4.2.1 Evaluation Model

IP multicast packets are transferred from the sender to N receivers. An IP packet is segmented into M of ATM cells. The evaluation is performed by the pt-to-mpt case, rather than by the mpt-to-mpt case. This is because the performance of the mpt-to-mpt case could be evaluated through the performance of the pt-to-mpt case.

With FEC-SSCS, the IP packet will be segmented into FEC-SDU(s), and FEC Frame (FEC-SDU plus FEC Code) is segmented into f of ATM cells. ATM cell will be errored or lost at each data-link segment with a probability of β. Without FEC-SSCS, the IP packet is errored, when any cell belonging to the IP packet is errored or lost. With FEC-SSCS, the IP packet is errored, when any FEC frame belonging to the IP packet is errored or lost. The FEC frame is errored, when the number of errored or lost cell are beyond the FEC's correction capability. In the evaluation, it is assumed that, when more than one cells in the FEC frame are errored or lost, the correct FEC-SDU can not be delivered. Then, the IP packet is errored.

The detailed calculation methods are described in Appendix B. Here, it is assumed that cells are randomly dropped or errored in the evaluation, even though cells will not dropped or errored at random in the actual system. The evaluation when the cell loss/error process is not random should be for further study item.

4.2.2 Performance of IP Packet Error or Loss Probability

Figures 4, 5 and 6 show the IP packet error or loss probability observed at the sender versus the cell error or loss probability β in each data-link segment. Approximately, M = 1330cells corresponds to 64 Kbyte IP packet, M = 160cells corresponds to 8 Kbyte IP packet and M = 10cells corresponds to 500 byte IP packet. The diameter of data-link segment d, which is the number of data-link segments between the sender and receivers, is 5. FEC frame size f is 10 cells, i.e. FEC-SDU is 9 cells. Figure 4 is the case of 10 receivers, figure 5 is the case of 10^3 receivers, and figure 6 is the case of 10^6 receivers. As shown, by applying the FEC policy, the IP packet error or loss probability can be reduced dramatically.

Figures 7, 8 and 9 show the IP packet error or loss probability observed at the sender versus the number of receivers N. The diameter of data-link segment d is 5, and FEC frame size f is 10 cells. Figure 7 is the case where the cell error or loss probability (β) in data-link segment is 10^{-3}, figure 8 is the case where β is 10^{-6} and figure 9 is the case where β is 10^{-9}.

For example, when the cell error or loss probability in data-link segment β is large (i.e. 10^{-3}), the IP packet error or loss probability without FEC policy is larger than 0.5 even for 10 receivers. On the contrary, with the FEC policy, the IP packet error or loss probability is less than 0.05 even for 100 receivers. And, when the cell error or loss probability in data-link segment β is 10^{-6}, the IP packet error or loss probability without FEC policy is larger than 0.5 for 10^3 receivers with 160 cells IP packet size. With the FEC policy, the IP packet error or loss probability is less than 10^{-5}

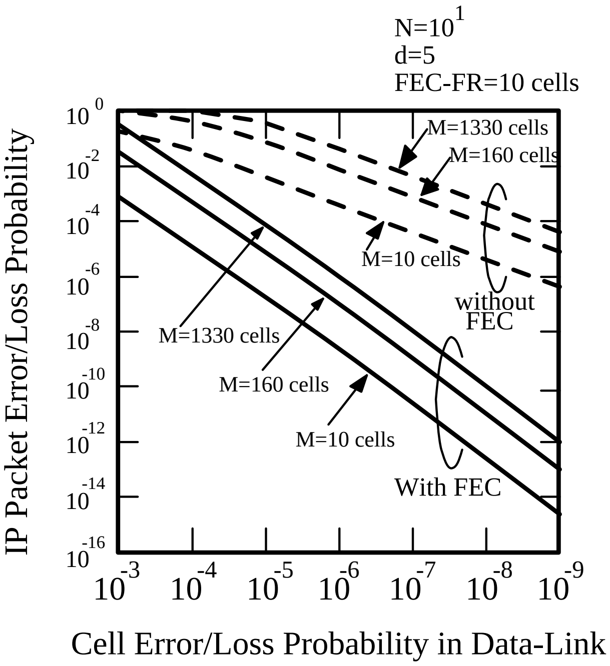


Fig. 4 IP Packet Error or Loss Prob. vs. CLR

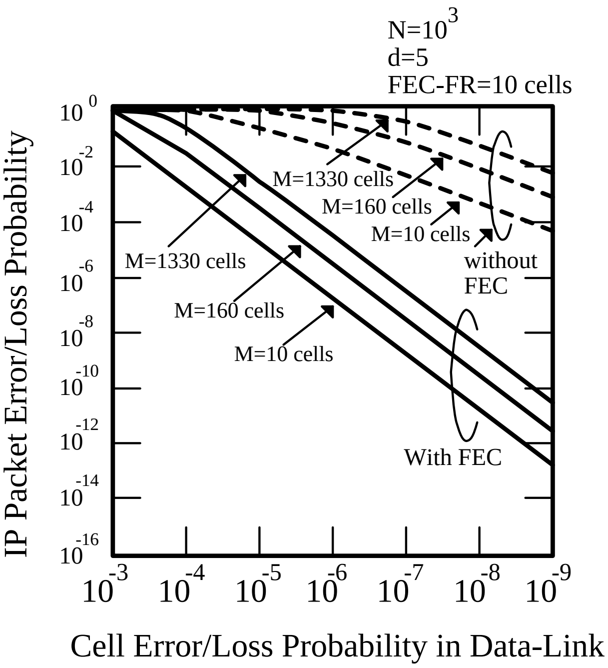


Fig. 5 IP Packet Error or Loss Prob. vs. CLR

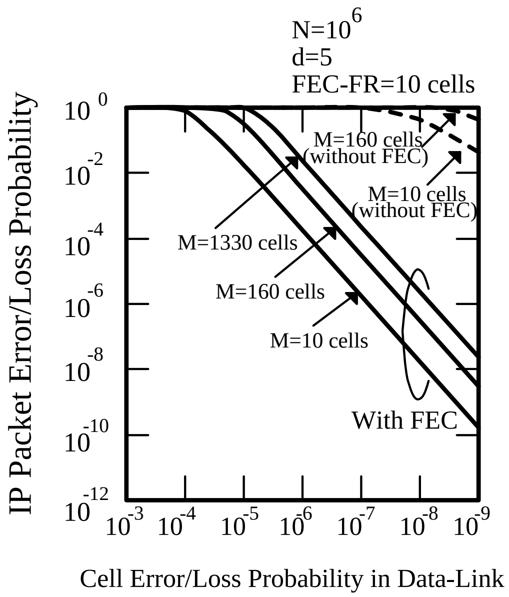


Fig. 6 IP Packet Error or Loss Prob. vs. CLR

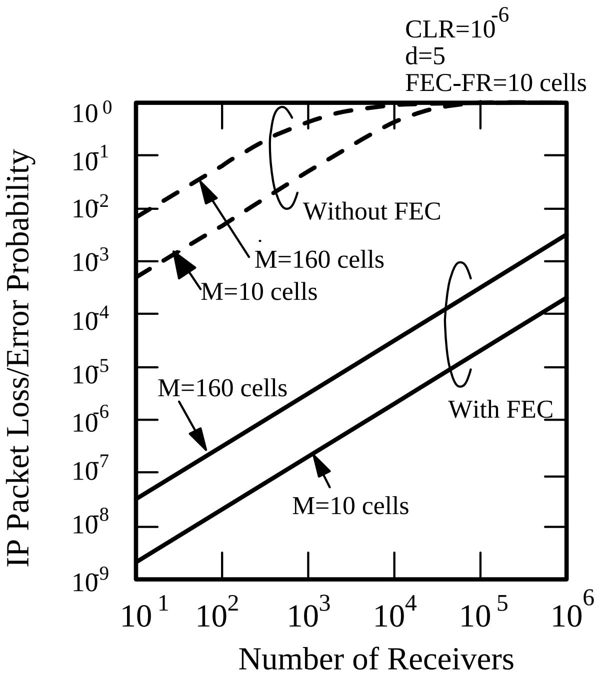


Fig. 8 IP Packet Error or Loss Prob. vs. Number of Receivers

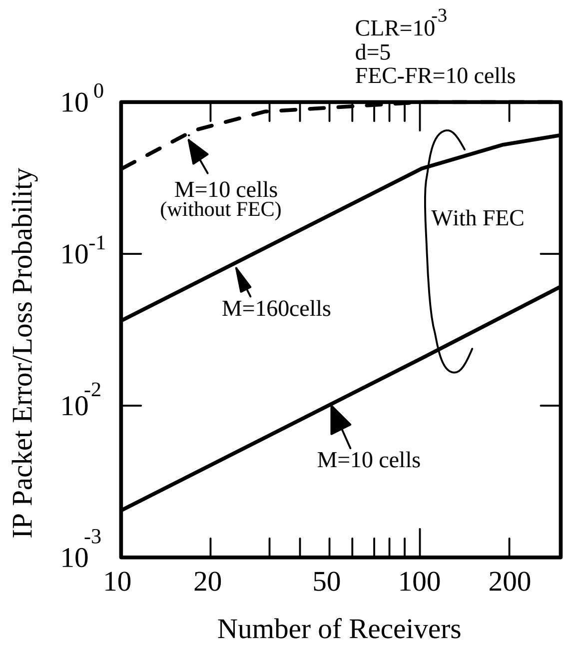


Fig. 7 IP Packet Error or Loss Prob. vs. Number of Receivers

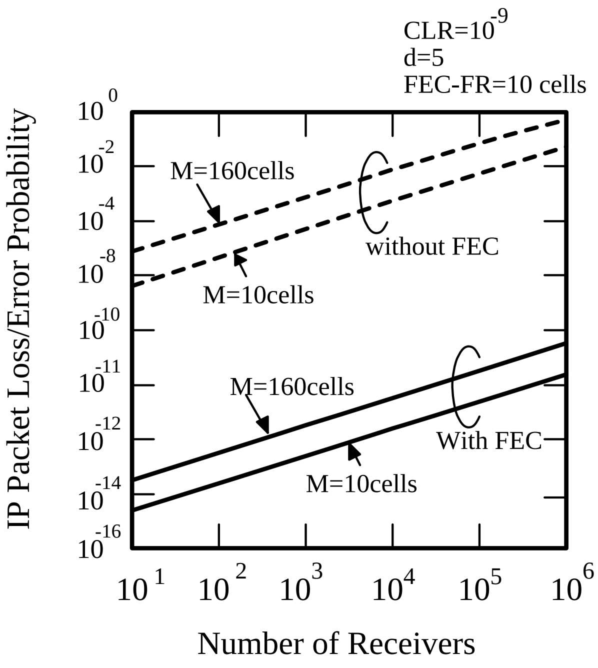


Fig. 9 IP Packet Error or Loss Prob. vs. Number of Receivers

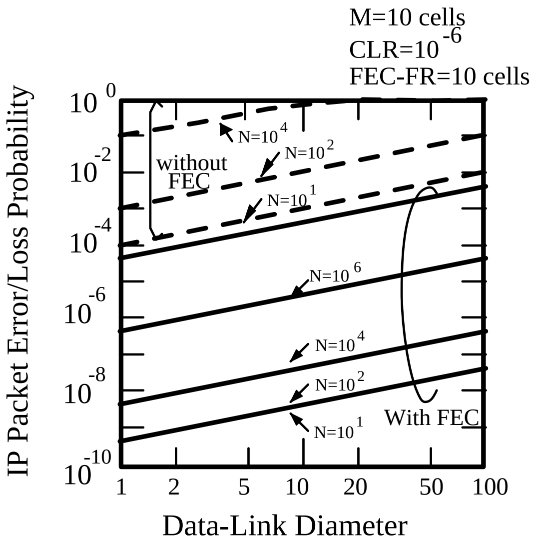


Fig. 10 IP Packet Error or Loss Prob. vs. Diameter of data-link

for 10^3 receivers, and it is less than 10^{-2} even for 10^6 receivers. These results indicate that the sender can take care of IP packet retransmission procedure without the intermediate TCP entities between the sender and the receivers, when the FEC policy is applied. Without FEC policy, the intermediate TCP entities between the sender and receivers seems to be required.

Figures 10 and 11 show the IP packet error or loss probability observed at the sender versus the diameter of data-link segments d. Since the contribution of d is approximately linear regarding the IP packet error or loss probability as described in appendices B.1 and B.2, the IP packet error or loss probability is not increased, even when the d becomes quite large value (e.g. 100). Here, d=100 would be for a quite wide area multicast. For example, d from a host (*isl.rdc.toshiba.co.jp*) in Japan to a host (*chorus.chorus.co.fr*) in France was 19 in the current Internet.

Figures 12 and 13 show the IP packet error or loss probability observed at the sender versus the FEC frame length f. The contribution of f is quite small, as shown in the figures. From the view point of transmission overhead of FEC codes, the larger FEC frame size will be preferable. But, as discussed in section 3, the FEC frame size should be aligned to the IP packet size. And, from the view point of the latency due to the FEC code calculation at the receiver, the FEC frame should not be large. Since many current system uses 576 byte packet size, around 10 cells size of FEC frame would be appropriate.

4.2.3 Performance of Transmission Overhead

In this subsection, the transmission overhead in order to provide an error-free packet delivery is evaluated.

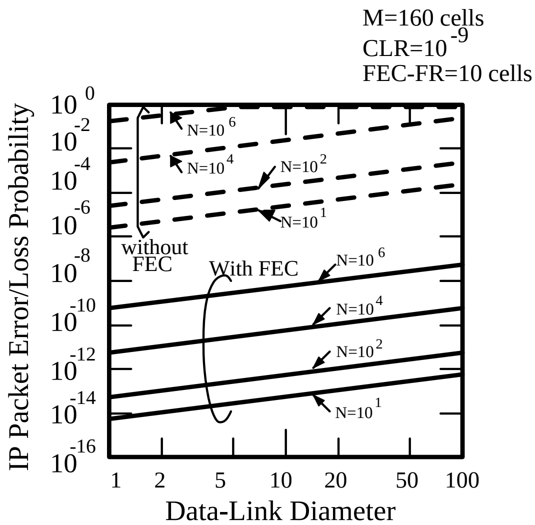


Fig. 11 IP Packet Error or Loss Prob. vs. Diameter of data-link

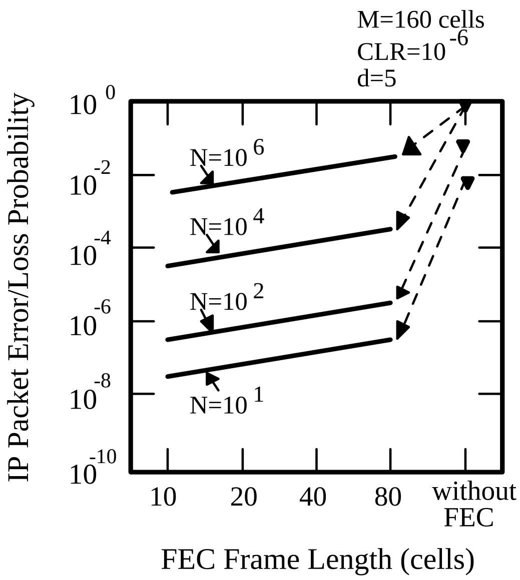


Fig. 12 IP Packet Error or Loss Prob. vs. FEC Frame Length

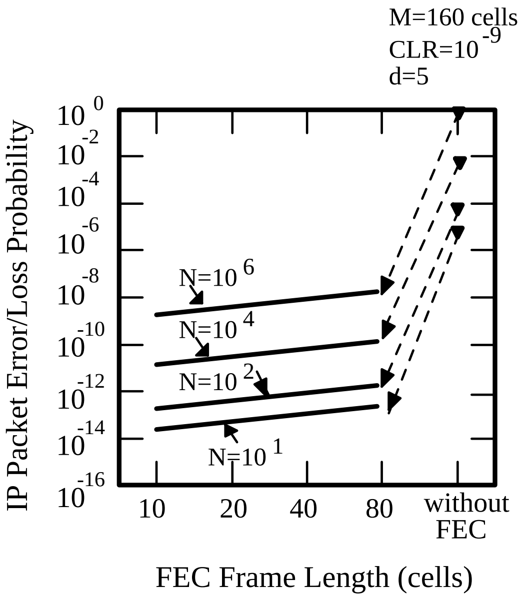


Fig. 13 IP Packet Error or Loss Prob. vs. FEC Frame Length

Transmission overhead is the following two factors.

- FEC code
  In order to perform the FEC policy, the FEC code is additionally transferred from the sender. The transmission overhead due to the FEC policy is given by $1/f$. For example, the transmission overhead is 10% when the FEC frame size is of 10 cells.
- Retransmission of errored or lost IP packet
  The errored or lost IP packet is subject to retransmission. Since TCP applies the go-back-N policy, the IP packets, which are correctly delivered to the receivers, would be subject to retransmission. However, in the evaluation, the retransmission of the IP packets, which are correctly delivered but are retransmitted due to the go-back-N policy, has not been taken into account. This means that the evaluated retransmission overhead corresponds to the case where the selective retransmission policy for the errored or lost IP packet(s) is applied (e.g. the IP packet retransmission policy of MTP in RFC1301). The detailed analysis is described in appendix B.3.

Figures 14 and 15 show the transmission overhead in order to provide an error-free packet delivery versus the number of receivers N. The diameter of data-link segments d is 5 and the FEC frame size f is 10 cells. Figure 14 is the case where the cell error or loss probability (β) at data-link segment is 10^{-3}, and figure 15 is the case where β is 10^{-6}.

When the cell error or loss probability (β) in data-link segment is large (i.e. 10^{-3}), the transmission overhead without the FEC policy is about 100% even for

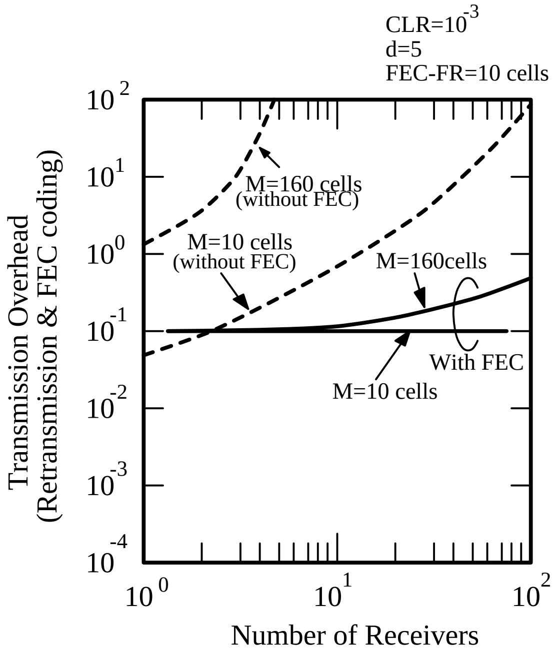


Fig. 14 Transmission Overhead vs. Number of Receivers

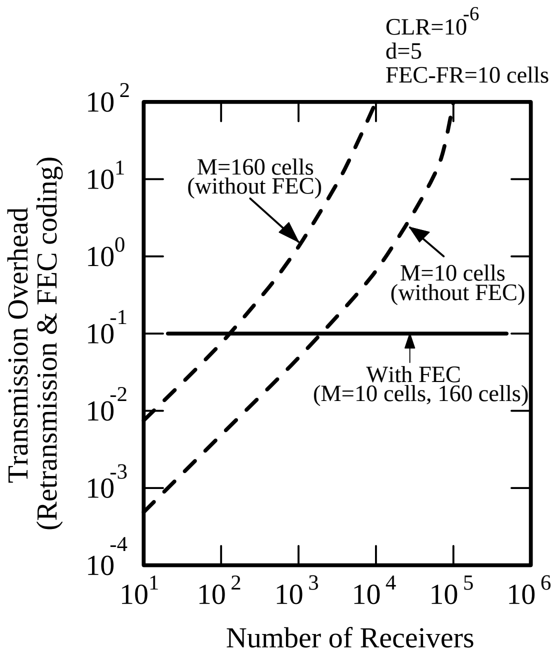


Fig. 15 Transmission Overhead vs. Number of Receivers

only 10 receivers with 10 cells sized IP packet. On the contrary, the transmission overhead with the FEC policy can be always about 10%, which is almost same as the overhead due to the FEC code, *even for 100 receivers* with 10 cells sized IP packet. That means that the transmission overhead due to the IP packet retransmission can be sufficiently small. When the cell error or loss probability in data-link segment β is 10^{-6}, the retransmission overhead without FEC policy is about 100% for 10^4 receivers with 10 cells sized IP packet. This means that throughput of this case will be less than 50%. On the contrary, the transmission overhead with the FEC policy can be again always about 10%. This means that the transmission overhead due to the IP packet retransmission can be sufficiently small, even for 10^6 receivers.

Since the probability of IP packet retransmission is sufficiently small with the FEC policy, a high throughput with small latent IP multicast packet delivery service can be provided by applying the FEC policy.

4.3 Discussion

4.3.1 Impact of Data-Link Sharing at Intermediate Links

The analytical evaluation above corresponds to the case where every IP packet delivery from the sender to receivers uses the separate paths. But, in the actual network, some links are shared, i.e. one down-stream IP packet flow will duplicated at the branching points (i.e. intermediate router). Also, some destinations (i.e. intermediate routers or end-receivers) could share the same physical transmission channel (e.g. pt-mpt connection without any server, e.g. Ethernet). Here, the number of destinations that shares the physical multicast channel could be usually less than 100.

Regarding the former factor, when $d = 1$, the expected packet error or loss probability could be the smallest. Let analyze the following case shown in figure 16. Here, the detailed analysis is described in appendix B.4. The i-th ($1 \leqq i \leqq B$) intermediate router multicasts the received IP packet to n_i of destinations (i.e. receivers or intermediate routers). And the diameter of data-links between i-th branching point and $(i+1)$-th branching point is d_i. Here, the first branching point corresponds to the root of multicast tree.

Then, the packet error or loss probability (P_{share}) for sender is given by the following equations.

- Without FEC policy

$$P_{wofec} \simeq d \times N \times M \times \beta \tag{1}$$

$$P_{share} \simeq \sum_{k=1}^{k=B} [d_k \prod_{i=1}^{i=k} n_i] \times M\beta \tag{2}$$

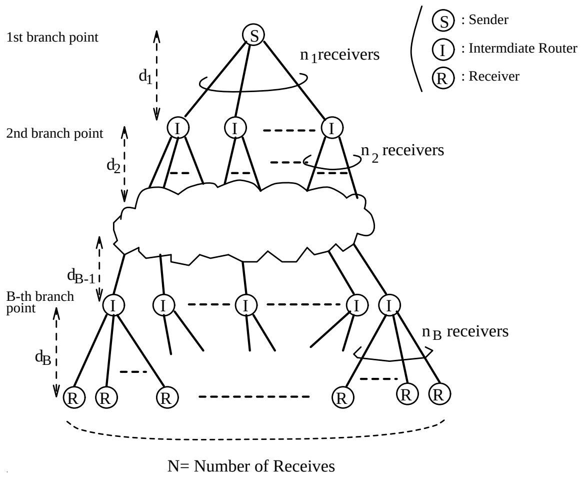


Fig. 16 Evaluation Model of Link Sharing Effect

$$< d_B \times N \times M \times \beta \tag{3}$$

- With FEC policy

$$P_{wfec} \simeq d \times N \times m' \times (f-1)^2\beta^2 \tag{4}$$

$$P_{share} \simeq \sum_{k=1}^{k=B} [d_k \prod_{i=1}^{i=k} n_i] \times (f-1)^2\beta^2 \tag{5}$$

$$< d_B \times N \times m' \times (f-1)^2\beta^2 \tag{6}$$

The difference is "$(\sum_{k=1}^{k=B} [d_k \prod_{i=1}^{i=k} n_i] < N \times d_B)$" of P_{share} and "$(N \times d)$" of P_{wodec} and P_{wfec}. Here, $\sum_{k=1}^{k=B} d_k = d$ and $\prod_{i=1}^{k=B} n_i = N$.

As shown in the previous subsection, the contribution of d to the IP packet error or loss probability is small. Therefore, the contribution of data-link sharing in the multicast tree could be also small as the evaluation results in the previous subsection.

Regarding the later factor, it can be modified the number of receivers in the evaluation results based on the number of receivers sharing the physical multicast channel, which could be from 10 to 100. But, regarding ATM networks, many ATM networks may provide a multicast service using a multicast server. In this case, the multicast service is provided by multiple point-to-point connections between the server and every client. Therefore, when a multicast service is provided by a multicast server in the (ATM-based) data-link segment, the later factor is not effective, i.e. the evaluation in 4.2 could be true for the actual network.

4.3.2 Impact of FEC Policy for Point-to-Point Communication

This paper focuses on the multicast communication (pt-mpt) to realize that the cell level FEC policy can provide an error-free IP multicast service with a simple architecture. The cell level FEC policy could also achieve the benefits for the point-to-point (pt-pt) communication, as well as the multicast communication.

Either(both) when the cell loss/error probability is not sufficiently small or(and) when the IP packet size is large, the FEC policy will yield a quite higher throughput compared to the case without FEC policy.

The IP packet loss/error probabilities are given by the following equations.

- Without FEC policy

$$P_{wofec} = 1 - (1-q)^d \tag{7}$$

where,

$$q = 1 - (1-\beta)^M \tag{8}$$

When β is sufficiently small (i.e. $\beta \ll 1.0$), P_{wofec} is almost equivalent to the following equation.

$$P_{wofec} \simeq d \times M \times \beta \tag{9}$$

- With FEC policy

$$P_{wfec} = 1 - (1-q)^d \tag{10}$$

where,

$$\begin{aligned} q &= 1-(1-s)^{m'} & (11)\\ s &= 1-(1-\beta)^{(f-1)} \times [1+(f-1)\beta] & (12)\\ m' &= \lceil M/f \rceil & (13) \end{aligned}$$

When β is sufficiently small (i.e. $\beta \ll 1.0$), P_{wfec} is almost equivalent to the following equation.

$$P_{wfec} \simeq d \times m' \times (f-1)^2 \beta^2 \tag{14}$$

Figure 17 shows the IP packet loss/error probability versus the cell error/loss probability of the each data-link segment for the point-to-point communication. Here, the FEC Frame size (f) is 10 cells, and the diameter of data-link segment (d) is 5.

When the IP packet size is large (e.g. 1330cells for 64Kbyte), the IP packet loss/error probability without FEC policy will be large, that is larger than 0.6% at $\beta = 10^{-6}$. Since the throughput of TCP will be suddenly degraded around $10^{-2} - 10^{-3}$, it could not be expected to obtain a sufficient throughput at TCP level without FEC policy, when IP packet size is large. On the contrary, the IP packet error/loss probability is 2.99×10^{-8} at $\beta = 10^{-6}$ with the FEC policy. Therefore, it is expected to obtain a sufficient throughput with the FEC policy, even for the large sized IP packet.

When β, the cell loss/error probability in data-link segment, is not small (e.g. 10^{-3}), it could not be expected to obtain a sufficient throughput without FEC

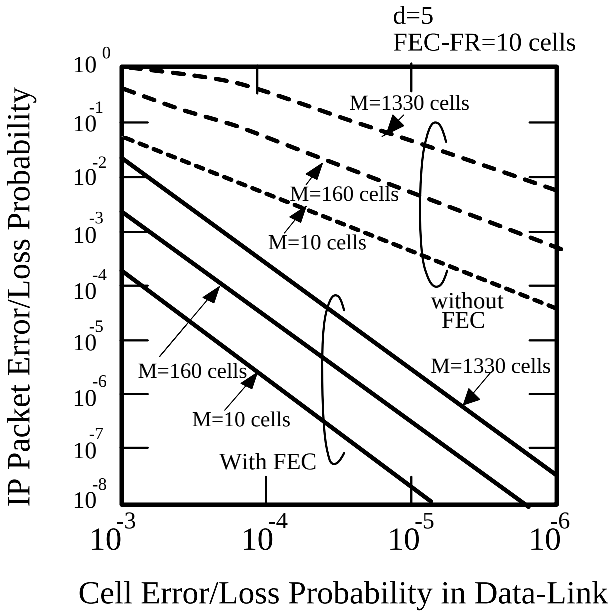


Fig. 17 IP Packet Loss/Error Ratio for Point-to-Point Communication

policy. In general, for the large error/loss probability in data-link, the IP packet size should be small. However, even when the IP packet size is 10 cells (for 500byte packet), the IP packet loss/error probability will be 4.8%. With 4.8% of the IP packet loss/error probability, it is expected that the throughput at TCP level will be poor. On the contrary, with the FEC policy, the IP packet error/loss probability will be 2.24×10^{-4}. As a result, it is expected to obtain a sufficient throughput with the FEC policy, even when the cell loss/error probability in data-link segment is not small.

5. Conclusion

This paper discusses and evaluates the effect of the cell level FEC policy on the error-free IP multicast service over ATM networks. By the analytical evaluations, the observed IP packet error/loss probability can be kept small enough by using the FEC policy, even when the number of multicast receiver is large, e.g. 10^5. And, therefore, with the cell level FEC policy, the end-to-end IP packet transmission latency will be small, and the end-to-end IP packet transmission throughput will be large enough. Though the FEC control requires some transmission overhead due to the additional FEC code information, a high throughput with small latent IP multicast service can be provided and, sometimes, the expected transmission overhead with the FEC policy is less than the transmission overhead without the FEC policy. This is because the probability of retransmission for the errored or lost IP packet can be sufficiently small, when the FEC policy is applied.

Acknowledgement

The authors thank to Prof.Daniel Duchamp, who works for CS (Computer Science) and CTR (Center for Telecommunications Research) of Columbia University in New York, for his valuable comments and suggestions, associated with this work.

References

[1] ATM Forum, "ATM User Network Interface Specification", Sept, 1994

[2] ITU-T Recommendation I.150, "B-ISDN Asynchronous Transfer Mode", 1990

[3] ITU-T Recommendation I.363, "B-ISDN ATM Adaptation Layer (AAL) Specification", July, 1993

[4] D.E.Comer, *Internetworking with TCP/IP*, Prentice Hall, 1991.

[5] ISI-USC, "Internet Protocol : DARPA Internet Program Protocol Specification", *IETF RFC*, RFC791, Sept., 1981.

[6] R.Atkinson, "Default IP MTU for use over ATM AAL5", *IETF RFC*, RFC1626, May, 1994.

[7] A.J.McAuley, "Reliable Broadband Communication Using a Burst Erasure Correcting Code", *ACM SIGCOMM90*, Sept., 1990.

[8] H.Esaki, Y.Tsuda, K.Kanai, "Evaluation of High Speed Multimedia Communication Architecture in ATM Networks", *IEICE Trans. on Communications*, Vol.E77-B, No.11, Nov. 1994

[9] S.Deering, "Host Extensions for IP Multicasting", *IETF RFC*, RFC1112, Aug., 1989.

[10] L.Zhang, "Resource ReSerVation Protocol (RSVP)", *IETF Internet-Draft*, draft-ietf-rsvp-spec-02.ps, May, 1994

[11] S.Armstrong, A.Freier, K.Maezullo, "Multicast Transport Protocol", *IETF RFC*, RFC 1301, Feb., 1992.

[12] L.Lamport, "Time, clocks, and the ordering of events in a distributed system", *ACM Trans, Commun.*, vol.21, No.7, July, 1978.

[13] F.B.Schneider, "Synchronization in distributed programs", *ACM Trans. Program. Lang. Syst.*, vol.4, No.2, April 1982

[14] K.P.Beirman, T.A.Joseph, "Reliable communication in the presence of failures", *ACM Trans. Comput. Syst.*, vol.5, No.1, Feb.,1987.

[15] J.Chang, N.F.Maxemchuk, "Reliable broadcast protocols", *ACM Trans. Comput. Syst.*, vol.2, No.1, Feb., 1987.

[16] H.Garcia-Molina, A.Spauster, "Ordered and reliable multicast communication", *ACM Trans. Comput. Syst.*, vol.9, No.3, August, 1991.

Appendix A: FEC-SSCS Spcification

Appendix A:1 FEC Frame Format

The format of FEC frame, that is actually transmitted from the source entity to the destination entity is described in Figure 18.

The FEC frame has user-data-part and FEC-code-part. Each part has the FEC-frame-header field. The user-data-part contains the part (or whole) of FEC-SSCS-SDU with the attached FEC-frame header filed. The FEC-code-part contains the FEC redundant information field with the attached header field.

- user-data filed
  The user-data field is used to carry the part (or whole) of FEC-SSCS-SDU. The "n", the length in vertical direction, is determined so that the length of the vertical line in the FEC frame, {(t,1),(t,2),(t,3),...(t,n)}, is $46/i$ octet. Here, i is a integer more than zero and it is usually one. In other words, the length of the vertical line in the FEC frame is usually 48 octet, that is the payload length of AAL5 SAR-PDU. The maximum length of horizontal line in the user-data must be less than $[2^p - q]$ symbols. Here, p is the symbol size in bit, and q is the horizontal length of FEC-code-part (s).

- FEC redundant data field
  The FEC redundant data field is used to carry the appended data to protect the user-data field information against bit error and cell loss. The data in FEC redundant data field is calculated based on the defined FEC algorithm with the negotiated parameter (i.e. p and q). The calculation algorithm of the FEC redundant data vector, {(m+1,t),(m+2,t),...(m+s,t)}, is based on the Reed-Solomon error recovery algorithm.

- FEC-frame-header filed
  The FEC-frame-header field is used to identify the position of missing or errored vertical line associated x-axis and to detect the bit error in each vertical line. The length of the FEC-frame-header is 2 octets in each vertical line. The detailed format and coding rule of the FEC-frame-header field is specified in figure 19.

The writing order and the reading order have the same order, to avoid the re-ordering procedure at the destination FEC-SSCS entity that causes some processing latency.

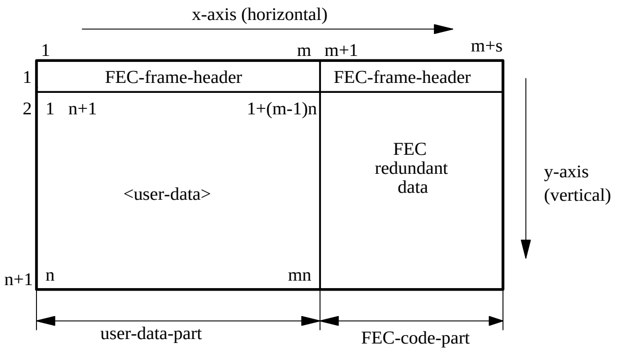
Each horizontal line (i.e. the horizontal vector) in the user-data part corresponds the user data protected by the FEC redundant information, that located at the same vertical position in the FEC-code part. "m" of symbols {(k,1),(k,2),....(k,m)} in user-data part are protected by the "s" of symbols {(k,m+1),(k,m+2),...(k,m+s)} in FEC-code-part.

Appendix B: IP Packet Error or Loss Probability

Appendix B:1 IP Packet Error or Loss Probability without FEC

The following is the parameters to evaluate the IP packet error loss probability without FEC policy.

- L2 data-unit error or loss probability in data-link segment : β



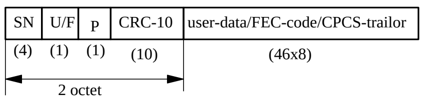
[Writing Order]

(1,2), (1,3), (1.4), ... (1,n+1),
(2,2), (2,3), (2,4), ... (2,n+1),
(3,2), (3,3), (3,4), ... (3,n+1),
: : :
(m,2), (m,3), (m,4), ... (m,n+1),
(m+1,2), (m+1,3), ... (m+1,n+1),
(m+2,2), (m+2,3), ... (m+2,n+1),
: : :
(m+s,2), (m+s,3), ... (m+s,n+1).

[Reading Order]

(1,1), (1,2), (1,3), (1.4), ... (1,n+1),
(2,1), (2,2), (2,3), (2,4), ... (2,n+1),
(3,1), (3,2), (3,3), (3,4), ... (3,n+1),
: : :
(m,1), (m,2), (m,3), (m,4), ... (m,n+1),
(m+1,1), (m+1,2), (m+1,3), ... (m+1,n+1),
(m+2,1), (m+2,2), (m+2,3), ... (m+2,n+1),
: : :
(m+s,1), (m+s,2), (m+s,3), ... (m+s,n+1).

Fig. 18 Frame Format of FEC Frame



CPCS-SDU	FEC	user-data-part	FEC	data-part
SN	1 2 3 4	5 6 7 - - 15 1 2- - 0 1	2 3 4 5	0 1 2 3 - - - -
U/F	F F F F	U U U- - U U U- - U U	F F F F	U U U U- - - -

Fig. 19 FEC-frame-header Field

(corresponding to Cell Loss Ratio in ATM network)

- Diameter of data-link segments : d
- Number of L2 data-unit's in one IP packet : M units
  (in ATM networks, it is M cells)
- Number of receivers : N
- Packet error or loss probability in data-link : q
- Packet error or loss probability for receiver : p
- Packet error or loss probability for sender : P_{wofec}

The packet error or loss probability in data-link (q) is given by the following equation, when the cell error or loss is occurred in random with the probability β.

$$q = 1 - (1-\beta)^M \tag{15}$$

Packet error or loss probability for receiver (p) is given by the following equation.

$$p = 1 - (1-q)^d \tag{16}$$

Then, the packet error or loss probability for sender (P_{wofec}) is given by the following equation.

$$P_{wofec} = 1 - (1-p)^N \tag{17}$$

When β is sufficiently small (i.e. $\beta \ll 1.0$), the equations are almost equivalent to the followings.

$$q \simeq M \times \beta \tag{18}$$
$$p \simeq d \times M \times \beta \tag{19}$$
$$P_{wofec} \simeq N \times d \times M \times \beta \tag{20}$$

Appendix B:2 IP Packet Error or Loss Probability with FEC

The FEC capability is indicated by "f". This means that when the error or loss L2-data-units within f L2-data-units (called as FEC-Frame) is less than two, the IP packet is correctly transferred.

Therefore, the error or loss probability of FEC-Frame, "s", is given by the following equation.

$$s = 1 - [(1-\beta)^f + f \times \beta(1-\beta)^{(f-1)}] \tag{21}$$
$$= 1 - (1-\beta)^{(f-1)} \times (1 - \beta + f\beta) \tag{22}$$
$$= 1 - (1-\beta)^{(f-1)} \times [1 + (f-1)\beta] \tag{23}$$

Let m' is $\lceil M/f \rceil$, which corresponds to the number of FEC-Frames in one IP packet. Then, q, p and P_{wfec} are given by the following equations. Here, P_{wfec} is the packet error or loss probability for sender, when FEC policy is applied to.

$$q = 1-(1-s)^{m'} \quad (24)$$
$$p = 1-(1-q)^{d} \quad (25)$$
$$P_{wfec} = 1-(1-p)^{N} \quad (26)$$

When β is sufficiently small (i.e. $\beta \ll 1.0$), the equations are almost equivalent to the followings.

$$s \simeq (f-1)^2 \times \beta^2 \quad (27)$$
$$q \simeq m' \times (f-1)^2 \times \beta^2 \quad (28)$$
$$p \simeq d \times m' \times (f-1)^2\beta^2 \quad (29)$$
$$P_{wfec} \simeq N \times d \times m' \times (f-1)^2\beta^2 \quad (30)$$

Therefore, the packet error or loss probability (P_{wfec}) will be about $(f-1) \times \beta$ times compared to P_{wofec}, that is for without FEC policy.

$$P_{wfec} \simeq (f-1) \times \beta \times P_{wofec} \quad (31)$$

Appendix B:3 Re-transmission Overhead to Provide Error-Free Delivery

When the packet error or loss probability at sender is P, the probability (α_k), that IP packet is successfully transferred by the k-th packet transmission (excluding first IP packet transmission), is given by the following equation.

$$\alpha_k = P \times P^{(k-1)}(1-P) \quad (32)$$
$$= (1-P)P^k \quad (33)$$

Then, the expected re-transmission number (R) is given by the following equation.

$$R = \sum_{k=1}^{\infty} k \times \alpha_k \quad (34)$$
$$= \sum_{k=1}^{\infty} k \times (1-P)P^k \quad (35)$$
$$= (1-P)\sum_{k=1}^{\infty} kP^k \quad (36)$$
$$= (1-P) \times P/(1-P)^2 \quad (37)$$
$$= P/(1-P) \quad (38)$$

Appendix B:4 Impact of Data-Link Sharing at Intermediate Links

The evaluation model is shown in figure 16. The packet error or loss probability (P_{share}) for the sender is given by the following equation. Here, Q_i is the packet error or loss probability for the d_i diameter's data-links, and r is the packet error or loss probability for a single data-link segment.

$$P_{share} = \prod_{k=1}^{k=B}\{1-(1-Q_k)^{\prod_{i=1}^{i=k} n_i}\} \quad (39)$$

$$Q_k = 1-(1-r)^{d_k} \quad (40)$$

Here,

$$r = 1-(1-\beta)^M \qquad \text{(for without FEC)} \quad (41)$$
$$r = 1-(1-s)^{m'} \qquad \text{(for with FEC)} \quad (42)$$
$$s = 1-(1-\beta)^{(f-1)} \times [1+(f-1)\beta] \quad (43)$$

When, β is sufficiently small ($\beta \ll 1.0$), the above equations can be almost equivalent to the followings.

$$P_{share} \simeq \sum_{k=1}^{k=B}\{Q_k \prod_{i=1}^{i=k} n_i\} \quad (44)$$
$$\simeq \sum_{k=1}^{k=B}\{r \times d_k \times \prod_{i=1}^{i=k} n_i\} \quad (45)$$

Here,

$$Q_k \simeq d_k \times r \quad (46)$$
$$r \simeq M\beta \qquad \text{(for without FEC)} \quad (47)$$
$$r \simeq m' \times s \qquad \text{(for with FEC)} \quad (48)$$
$$s \simeq (f-1)^2 \times \beta^2 \quad (49)$$

Then,

- Without FEC Policy

$$P_{share} \simeq \sum_{k=1}^{k=B}\{d_k \prod_{i=1}^{i=k} n_i\}M\beta \quad (50)$$

- With FEC Policy

$$P_{share} \simeq \sum_{k=1}^{k=B}\{d_k \prod_{i=1}^{i=k} n_i\}m'(f-1)^2\beta^2 \quad (51)$$

Now, let compare with the evaluation in B.1 and B.2.

- Without FEC policy

$$P_{wofec} \simeq d \times N \times M \times \beta \quad (52)$$

$$P_{share} \simeq \sum_{k=1}^{k=B}\{d_k \prod_{i=1}^{i=k} n_i\} \times M\beta \quad (53)$$
$$< d_B \times N \times M \times \beta \quad (54)$$

- With FEC policy

$$P_{wfec} \simeq d \times N \times m' \times (f-1)^2\beta^2 \tag{55}$$

$$P_{share} \simeq \sum_{k=1}^{k=B}[d_k \prod_{i=1}^{i=k} n_i] \times m' \times (f-1)^2\beta^2 \tag{56}$$

$$< d_B \times N \times m' \times (f-1)^2\beta^2 \tag{57}$$

Hiroshi Esaki received the B.S. and M.S. degrees from Kyushu University, Fukuoka, Japan, in 1985 and 1987, respectively. In 1987, he joined Research and Development Center, Toshiba Corporeation, where he engaged in the research of ATM systems. He has been at Bellcore in New Jersey (USA) as a residential researcher from 1990 to 1991, and has engaged in the research on high speed computer communications. From 1994, he has been at CTR (Center for Telecommunications Reserch) of Columbia University in New York (USA) as a visiting scholar. He is currently interested in an internet architecture using ATM technology and a mobile computing.

Takeko Fukuda received the B.S. and M.S. degrees from Kyushu University, Fukuoka, Japan, in 1963 and 1965 respectively, and received M.S. degree from M.I.T., U.S., in 1974. In 1965, he joined Research and Development Center, Toshiba Corporeation, where he engaged in the research and development of communication systems and equipment. He is currently Chief Technology Executive, Electronics & Telecommunication Systems Group, Tosiba Corporation.