Performance Evaluation and Optimization of Reliable Multicast

Zuo Wen Wan

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy at
Concordia University
Montreal, Quebec, Canada

2005

Your file Votre référence
ISBN: 978-0-494-27236-7
Our file Notre référence
ISBN: 978-0-494-27236-7

ABSTRACT

Performance Evaluation and Optimization of Reliable Multicast

Zuo Wen Wan, Ph.D

Concordia University, 2005

Many multicast applications require reliable delivery of data packets to multiple receivers. Scalability is one of the key challenges in the design of reliable multicast. The major obstacles of the scalability are feedback implosion and retransmissions. Furthermore, a real network changes with time. A reliable multicast protocol must adapt to such dynamic change of multicast sessions. Thus, it is necessary to design an efficient and adaptive loss recovery scheme for reliable multicast.

In this thesis, we present an efficient and adaptive loss recovery scheme, which is based on the performance evaluation of reliable multicast. The multicast performance depends on the loss recovery mechanism, the underlying tree topology, the loss characteristics, and the locations of repair servers. We present an efficient performance evaluation of these basic performance parameters, which is useful for adequate determination of the locations of repair servers.

ACKNOWLEDGEMENTS

My foremost thank goes to my thesis supervisor Dr. Ahmed Elhakeem. Without him, this thesis would not have been possible. I would like to thank him for his patience and encouragement that carried me on through difficult times, and for his insights and suggestions that helped to shape my research skills. His valuable feedback contributed greatly to this thesis. His visionary thoughts and energetic working style have influenced me greatly.

I would like to take this opportunity to express my great thanks to the rest of my thesis committee members: Dr. Eugene I. Plotkin, Dr. Abdel R. Sebak, Dr. Walaa Hamouda, Dr. Michel Kadoch, Dr. Malleswara Talla. Their constructive comments helped me to improve the thesis in many ways.

This work was supported by a CRD grant from NSERC and Bell Canada. The completion of this research was made possible thanks to the support of Bell Canada through its Bell University Research Program.

Finally, I thank my wife and my lovely children for supporting me through all these years. My wife contributes significantly to the completion of this thesis. I am in a real debit to my family.

To My Wife Xiao Dong Zhou

TABLE OF CONTENTS (TOC)

LIST OF FIGURES (LOF)

LIST OF TABLES (LOT)

LIST OF SYMBOLS (LOS)

ID	Meaning
S_i	The set of nodes covered by RR i.
S_0	The set of nodes not covered by any RR.
N	The total number of links within the group, also the set of these links.
N_i	The number of links within subgroup S_i, also the set of these links.
p	The packet loss probability of each link
$i, i_1, i_2, \ldots$	The level of RRs
$j, j_1, j_2 \ldots$	The number of independent losses for one packet in a transmission trial.
M, m	Total number of packet transmissions per source packet (due to losses)
P(M=m)	The probability that one packet multicast is successful in the m^{th} transmission (all nodes get the same packet).
$P(m\|j)$	The probability that multicast is successful in the m^{th} transmission after j losses take place.
$P(j_0 j_1 .. j_r)$	The joint probability of S_0 having j_0 losses, S_1 having j_1 losses,..., S_r having j_r losses in the first multicast transmission of a given source packet.
Q(j)	The probability that j independent losses take place in one transmission over different links.
C	The total bandwidth consumed for the multicast group (successful delivery

of one source packet to all nodes including all retransmissions.)

B	Average bandwidth consumption per link for the multicast group. B=C/N.
C_i	The bandwidth consumption for subgroup S_i, same description as C.
B_i	Average bandwidth consumption per link for subgroup S_i, $B_i = C_i / N_I$
$E[M_i]$	The expected number of transmission times for all nodes of subgroup S_i to receive correct source packet.
$E[C_i]$	The total expected bandwidth consumed for subgroup S_i member over all transmissions.
$E[B_i]$	The expected bandwidth consumed per link for subgroup S_i
$E[C]i_{1...}i_r$	The total expected bandwidth for the whole multicast group S when r RRs are used.
$E[B]\ i_{1...}i_r$	The expected bandwidth per link for the whole multicast group S when r RRs are used.
L	The depth of a tree
α_0	The probability that the loss falls into the domain of the sender
α_i	The probability that the loss falls into the domain of RR i.
d_k	The number of links from the sender to node k. It is also the set of these links.
$d_{k_1 k_2 \cdots k_i}$	The total number of links from the sender to node $k_1, k_2, \ldots, k_i$, i=1,2,.... It is also the set of these links.

n_k	The total number of links under node k. It is also the set of these links.
D_k	The total number of links involved in retransmissions if link k loses a packet. It is also the set of these links.
$D_{k_1 k_2 \cdots k_i}$	The total number of links involved in retransmissions if link $k_1, k_2, \ldots, k_i$ lose a packet (i=1,2,....). It is also the set of these links.
P(M=m)	The probability of the m transmission and retransmissions for one packet.
$P(M=m\|k_1k_2\ldots)$	The conditional probability of m retransmissions after link k_1, k_2, ..., lose the packet in a previous transmission.
G_t	The total number of links in level t.
λ	The pruning probability of receivers.
x	The average number of links involved in retransmissions if one independent loss takes place.
y	Topology variable
u, v	Topology parameters
λ_M	An arrival rate of multicast data packets
λ_B	An arrival rate of background traffic
G_M	The number of arrived multicast packets during the period of Δt
G_B	The number of arrived background packets during the period of Δt

θ	The number of transmitted packets
Y	The mean (an estimate of E[M]) of a random sample
α	The (1-α)% confidence interval.
$z_{\alpha/2}$	The z-value leaving an area of $\alpha/2$ to the right.
σ	The sample variance.
A_B	Maximum buffer size of routers
A_{ij}	The number of packets in the queue of node i at iteration j
A_i	The average number of packets in the buffer of router i
$\overline{A}$	The average number of packets in the buffer over all routers and iterations.
R	The number of routers
S	The number of iterations.
a_{ij}^M	The number of multicast packets arriving in router i at iteration j
a_{ij}^B	The number of background packets arriving in router i at iteration j
O_{ij}^M	The number of multicast packets lost in buffer i at iteration j.
O_{ij}^B	The number of unicast background packets lost in buffer i at iteration j.
O_i^M	The multicast packet loss percent in buffer i
$\overline{O^M}$	The average multicast packet loss percent over all routers

O_i^B	The background packet loss percent in buffer i
$\overline{O^B}$	The average background packet loss percent over all routers
$\rho_M = \lambda_M/\mu$	The multicast traffic intensity
$\rho_B = \lambda_B/\mu$	The background traffic intensity
T_j	Delay of packet j
$\overline{T}$	The average packet delay
σ_T^2	Variance value of delay
k	The number of data packets in FEC coding
n	The total number of packets in a FEC codeword
λ_P	The probability that a receiver prunes from the multicast session
λ_J	The rate that new receivers join the session
λ_s	The rate that new routers join the session
δ	The average service time of packets
μ	Service rate of routers
p_k	Packet loss probability over link k.
H_k	The set of the links from the sender to node k.
$\mathcal{M}_k$	The set of the links under node k.
$\mathcal{M}(\Omega)$	The set of the links under the links of the multiple loss scenarios Ω.

Ω	Link loss scenario, i.e. the set of independent link losses. For example, {1,2} means the losses from links 1 and 2, {0} means no loss.
$D(\Omega)$	The number of links involved in retransmissions for multiple loss scenarios Ω.
$n(\Omega)$	The number of links under the links of the multiple loss scenarios Ω.
$P(m\|\Omega)$	The probability that multicast is successful in the m^{th} retransmission after the loss scenarios Ω take place.
$Q_N(\Omega)$	The probability that multiple loss scenarios Ω take place in one transmission over N links.
$\Lambda(\Omega)$	Bandwidth consumed by NAKs for multiple loss scenarios Ω.
Λ_k	Bandwidth consumed by NAKs if link k loses a packet.
β	The ratio of a NAK packet to a data packet.
$Q_N(\Omega_0,\Omega_1)$	The probability that the loss scenarios Ω_0 and Ω_1 take place over N links

LIST OF ABBREVIATIONS (LOA)

ACK	Acknowledgement
ARQ	Automatic Repeat reQuest
AWGN	Additive White Gaussian Noise
CDF	Cumulative Distribution Function
DLR	Designated Local Retransmitters
DM	Domain Manager
DR	Designated Receiver
FEC	Forward Error Control
FIFO	First In First Out
IETF	The Internet Engineering Task Force
IP	Internet Protocol
IRMA	Illinois Reliable Multicast Architecture
LAN	Local Area Network
LBRM	Log-Based Receiver-Reliable Multicast
LMS	Lightweight Multicast Services
MAN	Metropolitan Area Network
MTP	Multicast Transport Protocol

NAK	Negative Acknowledgement
NCF	NAK Confirmation
ODATA	Original Content Data
OTERS	On-Tree Efficient Recovery using Subcasting
PGM	Pragmatic General Multicast
QoS	Quality of Service
RDATA	Retransmission Repair
RFC	The Request for Comments
RM	Reliable Multicast
RMTP	Reliable Multicast Transport Protocol
RR	Repair Router
RRMP	Randomized Reliable Multicast Protocol
RSE	Reed-Solomon Erasure
SPM	Source Path Message
SRM	Scalable Reliable Multicast
TCP	Transmission Control Protocol
TMTP	The Tree-based Multicast Transport Protocol
TTL	Time To Live
WAN	Wide Area Network

Chapter 1 INTRODUCTION

1.1 Background

With the rapid development of the Internet, more and more attentions to some applications such as videoconferencing, the distribution of financial and billing data, multimedia, network gaming, corporate communications, distance learning, and the distribution of softwares, stock quotes, and news, data communications, have been paid [1], [2]. These applications involve many-to-many or one-to-many transmissions to a lot of receivers over the Internet. It is possible to establish multiple unicast connections and accomplish such data transmissions between each sender-receiver pair. However, unicast using point-to-point protocols is far inefficient to transmit data to a large number of receivers because a lot of bandwidth is wasted in the transmission of the same data copy.

Multicasting provides an efficient means of one-to-many or many-to-many communications. The sender transmits only one copy of the data packet to all members in the multicast session, instead of one copy to each receiver in the Unicast protocol. Thus, the processing cost of the sender is reduced greatly. Multicast can save a lot of bandwidth consumption for large-scale receiver populations. It is very significant to save bandwidth in a bandwidth-starved environment.

IP (Internet protocol) Multicast is an extension to the standard IP network-level protocol. RFC1112 (The Request for Comments) in The Internet Engineering Task Force (IETF), Host Extensions for IP Multicasting [3], describes IP Multicasting as: "the transmission of an IP datagram to a 'host group', a set of zero or more hosts identified by a single IP destination address. A multicast datagram is delivered to all members of its destination host group with the same 'best-efforts' reliability as regular unicast IP datagrams. The membership of a host group is dynamic; i.e., hosts may join and leave groups at any time. There is no restriction on the location or number of members in a host group."

Multicast IP packet forwarding is best effort, just as it is with unicast packet forwarding. However, many of the above applications require reliable multicast communications, in which every member in a multicast session must correctly receive data packets from the sender. Due to packet losses during the transmissions, the retransmissions of the lost data packets are essential for reliable delivery. The most basic way that a receiver can recover from packet loss is by sending a negative acknowledgements (NAK) or acknowledgements (ACK) to the sender, which in turn retransmits the lost packet. There are, however, some challenges in the design of reliable multicast. One important factor is scalability. A major obstacle of scalability is that a sender has to handle a lot of feedback information such as NAKs or ACKs. If the same packet is seen as missing by several receivers simultaneously, the sender will become congested handling NAKs or ACKs. This is NAK or ACK

implosion [1]. Another important factor is retransmissions. Bandwidth consumption strongly depends on the method of retransmissions. Retransmission schemes decide the performance of reliable multicast.

Many loss recovery schemes have been proposed to overcome these design challenges of reliable multicast [3]-[6]. Efficient loss recovery is essential for reliable multicast and is a key issue in the design of reliable multicast protocols.

1.2 Motivation

Loss recovery is a key technique for the design of a large-scale reliable multicast protocol. The retransmission of the lost packets is required to guarantee the reliability of multicast. Whatever loss recovery schemes are used, repair packets need to be retransmitted by some special routers or receivers. Different reliable multicast protocols use different terminology to represent such repair routers or receivers, e.g., designated receivers (DR) in RMTP (Reliable Multicast Transport Protocol) [4]-[6], link repliers [7], [8], Designated Local Retransmitters (DLRs) [9], group controllers [10], domain managers [11], proxies [12], [13], log servers [14], active routers [15]-[17], repair servers [18]-[20], representative member [21], and so on. They may be members of the group or special servers; they may be collocated with the routers or not. They all share similar fundamental functionality. They report the state of subgroup to the sender on behalf of their domains in order to avoid

the feedback implosions. They request retransmissions from the sender if they do not receive the correct packets. On the other hand, they retransmit the repair packets to the receivers that lost data upon receiving the retransmission request. They work as both senders and receivers. Overall, they play an important role in reliable multicast. Their placement is clearly an important topic in reliable multicast because multicast performance depends greatly on their placement.

In this thesis, we will use repair routers (RR) to represent these special routers, receivers, or repair servers. Such repair routers have the function of repairing lost/damaged packets and will request a retransmission from their parent repair routers or the sender if they miss a packet. RRs cache the received packets and retransmit them upon receiving the request for retransmission. Repair services are enabled only at some special receivers or routers to support the reliable delivery of data packets. It is unrealistic to have repair services enabled at all routers. The impact of these special receiver or router locations on multicast performance is significant. How to find their best locations is a challenging task.

Furthermore, a real network changes with time due to routing changes, as any receivers join and leave the multicast session at any time. Due to the dynamics of networks, the performance of reliable multicast is another challenging work to be taken into consideration. Adaptive transport protocols need simple but effective mechanisms to decide multicast performance. This requires a fast evaluation of some network parameters

for a dynamic network. Current performance analysis includes the number of transmissions, bandwidth consumption and delay. The computation of these parameters is very intensive and thus is not applicable to an adaptive multicast protocol. It is necessary to find an efficient way for estimating network performance, so efficient and adaptive allocation of network resources in a real time network would be possible.

Although a large number of reliable multicast protocols have been proposed, these issues are still important for practical applications. Optimization of the performance of multicast networks can save bandwidth consumption and reduce delay greatly. This is the main objective of this thesis. We will focus primarily on the discussion of wired multicast networks in this thesis. We will not consider the effect of AWGN (Additive White Gaussian Noise) channel and interferences.

1.3 Contributions

The performance of reliable multicast depends on many factors such as the loss recovery schemes, the underlying multicast tree topologies, the loss characteristics, and the locations of repair servers. This thesis will address these issues and discuss their effects on multicast performance. The following lists the objectives and contributions of this thesis.

- **The performance evaluation of reliable multicast.** We analyze and simulate the

basic performance parameter, such as the number of transmissions, bandwidth consumption and delay, based on two typical loss recovery schemes, i.e. application-level and router-assisted loss recovery.

- **The effects of topologies on reliable multicast.** A new topology parameter is introduced to explain the effects of topology on the performance of reliable multicast.

- **Performance analysis of dynamic networks.** We analyze the effects of member pruning on multicast performance.

- **The optimal locations of repair servers for tree-based reliable multicast.** The performance of reliable multicast depends greatly on the RR locations, thus, it is significant to find the optimal RR placements.

- **A 3-phase algorithm for finding the optimal locations of repair servers adaptive to the change of dynamic networks.** Due to the change of dynamic networks, efficient adaptation of repair services to the resulting dynamic network is significant to loss recovery design.

- **The loss characteristics of multicast networks with heterogeneous link loss**

probability. The loss characteristics of various tree topologies and their effects on the optimal RR locations are analyzed.

1.4 Thesis Structure

The rest of this thesis is organized as follows. Chapter 2 gives related work of reliable multicast.

In chapter 3, we present a comprehensive performance comparison of two loss recovery schemes for hierarchical reliable multicast. End-to-end and router-assisted loss recovery schemes are analyzed.

Chapter 4 studies the effects of topology on multicast performance. We first evaluate analytically the number of transmissions and bandwidth consumption for multicast networks. Then we propose a general method to locate RRs. Finally, we discuss the effects of topology on multicast performance and optimal partitioning.

In chapter 5, we first discuss the performance of dynamic networks, which focus on the pruning of receivers and subnets. Then, we propose a 3-phase algorithm to adapt the optimal placements of RRs to the change of dynamic networks.

Chapter 6 simulates a reliable multicast protocol. The delay performance of FEC and

ARQ/FEC are compared based on multicast delivery of a best effort network. Buffer overflow is addressed. A lot of simulation results are given and compared with analytical results.

Chapter 7 analyzes the performance of multicast networks with heterogeneous link loss probabilities and discusses the effects of the loss characteristics of trees and control traffic.

Finally, we conclude in Chapter 8 with a discussion and proposals for future work.

Chapter 2 RELATED WORK

2.1 Introduction

A larger number of reliable multicast protocols have been proposed over the past few years [3]-[50]. They can fall into ARQ-based (Automatic Repeat reQuest) [4], [6], [9], [14], [22]-[35] or FEC-based (forward error correction) [36]-[46] protocols depending on whether or not redundant packets are transmitted along with original data packets. They can also be classified into application-level [4]-[6], [14], [22]-[35] or router-assisted [7]-[9], [47], [50] protocols depending on whether routers are utilized to recover loss or not. Whatever loss recovery is used, it is always required to isolate losses from these receivers that have received the packet. Therefore, loss recovery is often involved in local recovery. In the following, one will briefly introduce some important reliable multicast protocols: tree-based, router-assisted, application-level, and FEC-based reliable multicast, which will be addressed and analyzed in this thesis.

2.2 Tree-based Reliable Multicast

The receivers are organized in a tree such that the sender is at the root of the tree, the receivers are at the leaves and domain representatives are at the intermediate points of the tree. Domain representatives or designated receivers (DR) represent a group of receivers or

a domain and are also organized in a hierarchical manner. Therefore, tree-based reliable multicast protocols often use hierarchical loss recovery. There are a number of tree-based protocols. Some famous tree-based protocols are RMTP (Reliable Multicast Transport Protocol) [4]-[6], TMTP (Tree-based Multicast Transport Protocol) [11], RRMP (A randomized error recovery algorithm for reliable multicast) [23], The LBRM (Log-Based Receiver-Reliable Multicast) [14], and so on.

RMTP is one of the best known examples of tree-based protocols. RMTP is an early static hierarchical structure in which receivers are grouped into local domains and in each domain there is a special receiver called a Designated Receiver (DR). DR is responsible for sending ACK to the sender, for processing ACK from receivers in its domain and for retransmitting lost data to the corresponding receivers. The DR is like a combination of a sender and a receiver, but DR never transmits new packets. A DR maintains a cache to store received packets that may be needed for retransmissions later on.

The sender multicasts data to all receivers and DRs return acknowledgements to the sender. DR is a representative of the local domain. All the members in a local domain only can send their status to the corresponding DR. DR uses these status messages to perform local retransmissions to the receivers, reducing the end-to-end delay. Only the DRs in a multicast group can send their status to the sender. Thus, the sender sees only the DRs and a DR sees

only the receivers in its local region. The processing of status messages is distributed among the sender and the DRs, thereby, avoiding the ACK-implosion problem. RMTP uses manual configuration to construct and maintain the hierarchy, thus, DR in each domain is assigned beforehand.

Unlike RMTP, TMTP is a dynamically hierarchical structure [11]. In TMTP, every region has a Domain Manager (DM) that is responsible for retransmission. A DM uses an expanding ring search to look for a parent so as to join a multicast group. The new DM always selects the closest DM as its parent. Each endpoint maintains the hop distance to its DM, and each DM maintains the hop distance to its farthest child. These values are used to set the TTL (time to live) field on requests and replies to limit their scope. TMTP also uses hierarchy to recover losses.

A randomized error recovery algorithm for reliable multicast, called RRMP in [23], is used to recover losses. RRMP does not use any repair server or the fixed DR to retransmit data, but any receivers having the packet to recover losses. The responsibility of loss recovery is randomly distributed among all members in the group because the lost packet is retransmitted by a randomly chosen receiver. Therefore, the robustness is improved.

When a receiver p detects a loss in RRMP, the receiver randomly selects another neighbor receiver q to retransmit data rather than multicast its request to the group. p sets a timer

according to its estimated round trip time to q. Upon receiving the request from receiver p, the receiver q checks whether it has the data. If so, it sends the data to p. Otherwise, it ignores the request. If p does not receive a repair packet when its timer expires, it randomly selects another receiver in its region and repeats the above process. As long as at least one local receiver has the data, p is eventually able to recover the lost packet.

If a local region misses a packet, the packet loss cannot be repaired within the local region using the above loss recovery. In this scenario, the receiver, e.g. p, randomly chooses a remote receiver r in its parent region not in the local region and sends a request to r. p also sets a timer according to its estimated round trip time to r. Upon receiving a request from a remote receiver p, r checks whether it has the requested packet. If so, it sends the packet to p. Otherwise, r ignores the request as well. If p does not receive the repair packet when its timer expires, it randomly selects another remote receiver in its parent region and repeats the above process until it receives the lost packet.

2.3 Application-level Reliable Multicast

Application-level reliable multicast protocols do not have any knowledge about the underlying topology. For example, SRM (Scalable Reliable Multicast) [25] and RMTP do not know the multicast topology. Routers are not used to recover loss in many of reliable multicast protocols, which include SRM [25], RMTP [6], TMTP [11], LBRM [14], RRMP

[23], MTP [22], APES (Active Parity Encoding Services) [44], HRM (hierarchical reliable multicast) [12], [13], and so on [31], [32], [48], [80]. The following introduces the loss recovery mechanism of SRM.

The SRM protocol [25] uses the receiver that has the data to recover lost packets. A basic idea of SRM is that lost data need be retransmitted not by the sender but by any receiver that has the data packet. The nearer to the lost receiver the repair receiver is, the better. It is expected that the nearest receiver multicasts the retransmitted data.

When a receiver misses a data packet, it waits for a random time determined by its distance from the original source of the data, before it sends a repair request. Repair requests are multicast to the whole group just as regular data packets are. Thus, although many receivers may lose the same data, a receiver close to the point of failure is likely to timeout first and multicast the request. Upon receiving the request, the other receivers that are also missing the same data cancel their own requests. This prevents request implosion.

Any receiver that has a copy of the requested data can answer a request. However, it will set a repair timer to a random value depending on its distance from the receiver that sends the request message. When the timer goes off, the repair packet is multicast. Other receivers that have the data and scheduled repairs will cancel their repair timer when they receive the repair packet from the first receiver. This prevents a response implosion.

For the basic SRM protocol, the repair requests are always multicast to the entire group. If there is a single lost receiver, it will flood the network with repair requests. An obvious way to solve the problem is to restrict the propagation of the repair request to a small region around the receiver that lost the packet. This mechanism of limiting the repair request within a local region and repairing the losses using retransmission within the local region is called local recovery. In SRM, it is essential to use local recovery in order to confine the region of retransmission and the repair request [66].

2.4 Router-assisted Reliable Multicast

Because routers have some underlying information about topology, routers are often used to recover losses in reliable multicast. Router-assisted loss recovery schemes include pragmatic general multicast (PGM) [9], lightweight multicast services (LMS) [7], [8], [51], search party [47], OTERS (On-Tree Efficient Recovery using Subcasting) [50], and so on [52].

Pragmatic General Multicast (PGM) [9] is an approach to using the network infrastructure, i.e., router. PGM defines only a few data packets: ODATA (original content data), NAK (selective negative acknowledgment), NCF (NAK confirmation), RDATA (retransmission repair), SPM (source path message). ODATA, NCF, RDATA, and SPM packets flow downstream in the distribution tree, and NAK packet flows upstream toward the sender.

When a receiver detects a lost packet, it sends a NAK to the nearest PGM-aware router to request retransmissions. Each PGM-aware router keeps forwarding NAKs until it sees an NCF (NAK confirmation) or RDATA (retransmission data). All PGM-aware routers eliminate duplicate NAKs all the way upstream to the sender. Thus, routers provide NAK suppression through constrained NAK forwarding. PGM-aware routers also keep state on where the NAKs come from in the distribution tree. Therefore, they may confine the forwarding of RDATA repairs to only those receivers listed in the repair states for the respective data. Routers that do not have a repair state for the data should not forward RDATA data units. Thus, retransmissions are constrained to those subnetworks whose host receivers lost the data. This protocol feature reduces bandwidth consumption.

PGM is a second generation of reliable multicast protocols. PGM minimizes both the probability of NAK implosion and the loading of the multicast network due to the retransmissions of lost packets. With this combination of characteristics, PGM provides a highly reliable IP Multicast transport mechanism that should be able to scale to networks as large as the Internet.

Lightweight Multicast Services (LMS) [7], [8], [51] extends the routers with a new but simple forwarding service. For each router, a replier link that is one of router children links is assigned to deal with data retransmission and communicate with its parents. A request

from a non-replier link is forwarded to its replier link, while a request from replier link is forwarded to its parent replier link. The routers guide requests to repliers, and in the process discover the root of the loss subtree. This is called the turning point. The turning point is the point in the topology at which a request moving upstream is moved downward towards the replier. The router at the turning point performs a directed multicast to multicast the retransmission to the subtree defined by the turning point. When the loss occurs on a non-replier link, recovery is only confined to the loss subtree. When the replier link loses a packet, recovery is confined to a parent subtree. Upon receiving a retransmission request, the replier unicasts a multicast packet to the turning point if it has data. The turning point uses subtree multicast to send the repair packet to all receivers in the subtree. The advantages are that it has lower recovery latency and excellent response instantly to group changes.

Search party proposed in [47] follows the structure of LMS: requests for retransmissions are sent up the tree toward the root and bounced back down with inserted information about the turning point. However, it modifies LMS from simple deterministic forwarding service to randomized forwarding service with randomcast that forwards packets randomly inside a multicast tree. No replier links are assigned to routers to deal with retransmission. Routers randomly choose another child link except the lost link to retransmit data. Robustness can be improved through randomization and responsibility is diffused.

Although routers are used to recover loss in a reliable multicast, this does not mean that router-assisted multicast outperforms greatly application-level schemes [48][49]. Contrary to intuitive expectations, the qualitative performance of application-level hierarchy is comparable to router-assisted hierarchy when a good loss recovery technique is used to build the hierarchy. This is a very interesting result.

2.5 FEC-based Reliable Multicast

FEC (forward error control) is another efficient way to recover lost data for multicast communication [36]-[46], [54]-[58]. It transmits redundant data, called parity data, along with the original data. Reed-Solomon erasure (RSE) correcting code is often used to generate redundant packets for reliable multicast. If the number of lost original packets is not more than the number of parity packets, the parity packets can be used to reconstruct the lost original data packets. Thus, it can reduce greatly the number of retransmissions and latency [40]. Because Tornado codes have a much lower computational complexity (much faster encoding-decoding), they are often used for reliable multicast distribution of bulk data [54][58]. FEC-based loss recovery is very attractive for some applications such as real-time video and audio transmissions.

The RSE encoder takes k data packets and produces n-k parity packets. Thus, one has n packets. The RSE decoder at the receiver side can reconstruct the data packets whenever it

has received any k out of the n packets. There are multiple benefits using parity packets for loss recovery rather than retransmitting the lost original data packets, especially in transmission efficiency, scalability, and unnecessary receptions.

FEC by itself cannot provide full reliability. It must be combined with ARQ. Thus, there are two approaches for FEC-based reliable multicasts: layered FEC and integrated FEC. Integrated FEC/ARQ schemes are also called as hybrid ARQ.

2.5.1.1 Layered FEC

FEC and reliable multicast (RM) layer are separated for layered FEC. At the sender, the RM layer passes the data packets to the FEC layer. After receiving k original packets, the FEC layer produces n-k parity packets and sends all of the packets to the receiving FEC layer. Whenever the receiving FEC layer receives at least k out of n packets, all of the original data packets can be reconstructed and sent to the receiving RM layer. If fewer than k packets are received, the lost original packets cannot be reconstructed and the FEC layer discards the received parity packets. The sending RM layer then puts the lost original packets into a new FEC block and repeats the above process.

2.5.1.2 Integrated FEC

There are many ways FEC can be included within the RM layer to improve its performance.

A generic integrated FEC has the following characteristics:

- The sender multicasts k original data packets and some parity packets (e.g. q<n-k parity packets) from the associated FEC block.

- If no more than q packets are lost among the k+q sent packets, all the original packets can be recovered. If a receiver misses more than q packets, it requests a new parity packet from the sender until it receives k packets out of the n packets in the FEC block to recover all k data packets.

- The sender multicasts parity packets. If all parity packets (n-k parity packets) have been used up, the lost data packets are placed into the next data block.

By simulations, it was found that integrated FEC dramatically reduces the mean number of transmissions as compared to the case of no FEC and has better performance than layered FEC [40].

The performance analysis of FEC-based loss recovery was examined in [71]. Performance comparison is based on two approaches for loss recovery in reliable multicast: centralized and distributed recovery. Their results are obvious that a distributed protocol based on integrated FEC/ARQ reaches best performance in terms of bandwidth and latency.

2.6 Performance Analysis

The above protocols have been suggested for different applications with different performance requirements. Performance parameters used in reliable multicast include the number of transmissions, bandwidth consumption, delay, and throughput [62]-[81]. The number of transmissions is often used to estimate bandwidth consumption and delay in the performance analysis of reliable multicast. The number of transmissions is an important measure, which first appeared in [62] and has been used in most analytical works on reliable multicast [13], [18], [20], [36]-[41],[44], [45], [64], [65], [71], [75].

The number of transmissions can be computed numerically by a set of equations [62]. Due to the intensive computation for general topologies, one often ignores the effects of topology [37]-[41],[71], [75], or investigates the multicast performance only for some simple topologies such as linear, binary tree and Fanout topologies [36]. The average throughput is derived from the cumulative loss probability. Even when the buffer overflow probability at routers and receivers is low, the cumulative loss probability seen by a sender may be quite high [62].

A.P. Markopoulou and F.A. Tobagi discussed the performance analysis of hierarchical reliable multicast [12]. They suggested a reduction technique that a tree can be equivalent to one link in order to calculate the number of transmissions. However, the computation

can be still exponential in the number of links N in the general case, e.g. for a star topology with N links, one has to consider the union of error events on links 1, 2, … , N so that 2^N subsets must be calculated [12], [13]. The computation of the number of transmissions is clearly intensive for general topologies. They emphasized the importance of location problems for hierarchical reliable multicast. An algorithm was suggested to find the placements of proxies for the multicast and unicast cases [13]. Due to the huge amount of calculation, they had to upper bound the cost of multicast trees in their dynamic program approach. It is difficult for this algorithm to adapt to the change of dynamic networks due to the lack of the knowledge of topology.

Bandwidth, delay and throughput analysis have been documented for some reliable multicast algorithms [6], [9], [40], [59], [71]-[73], [80]. In their analysis, the number of transmissions is used to estimate these basic performance criteria. They do not deal with the estimation of the number of transmissions. Thus, because of the intensive computation of the number of transmissions, it is hard to efficiently evaluate bandwidth, delay and throughput for dynamic topologies, especially for real time loss recovery techniques as in chapter 5. Sender-initiated and receiver-initiated reliable multicast protocols were compared in [73]. The performance analysis of centralized and distributed loss recoveries was examined in [40]. Their results show that a distributed protocol based on integrated FEC/ARQ reaches best performance in terms of bandwidth and latency [71].

Some references also discuss the optimization of some factors [91]-[98]. Gang Feng et al [15], [17] discussed the optimal cache-partitioning and optimal cache allocation for active reliable multicast. For active reliable multicast, each router can transmit the repair packet. Therefore, it is unnecessary to suggest the optimal placements of repair servers. However, it is important to optimize the cache allocation of active routers for different multicast sessions. The cache location problem has been addressed in [91]. C.Mailhofer et al [92] discussed the optimal branching factor for tree-based reliable multicast protocols. S.K.Karera et al [18] analyzed buffer requirements and replacement policies for multicast repair service. One can also find the importance of local recovery from their numerical results.

For unicast transmissions over the Internet, the optimal placements of web proxies have been discussed in [68], [84], [91]-[95]. Web proxies, like DRs in reliable multicast, are often used to handle the ever-increasing demand for information retrieval over the Internet. Dynamic programming is used to determine where to place web proxies for different numbers of web proxies based on the cost of network [68]. The cost depends only on distance from each node to the proxy in unicast transmissions. However, a multicast connection has different characteristics because the cost depends not only on the path from a node to the proxy, but also on the loss behavior of the rest members.

In [19], three methods for the placement of repair servers, namely k-maximum path count, k-maximum degree and k-minimum average distance, were proposed. For each method, one needs to calculate the corresponding parameter of each node, i.e., path count (the number of the shortest paths passing through the node), degree (the number of links connected to the node), and distance (the average number of hops along the shortest paths from the node to all other nodes). The first k nodes with the maximal path counts, the maximal degree or the minimal distance are selected to place the repair servers. Although they did not aim at finding optimal placements, they could yield better multicast performance than the random placement method of repair servers.

Chapter 3 A PERFORMANCE COMPARISON OF TWO LOSS RECOVERY POLICIES

3.1 Introduction

Much attention has been paid to reliable multicast in recent years [3]-[50]. Scalable loss recovery is a key factor in reliable multicast design. Two major loss recovery schemes, i.e, end-to-end schemes [4], [14], [22]-[24], [30], [35] and router-assisted schemes [7]-[9], [47]-[53], have been proposed to provide scalability. Whatever loss recovery is used, it is always required to isolate the losses from these receivers that have received the packet. Hierarchical loss control is a popular approach to limiting the scope of recovery of data and control messages. In these hierarchical schemes, a multicast group is partitioned into subgroups [4], [12], [23], [69]. Within each subgroup, a special receiver or router is selected to be responsible for the retransmission of the packet and to collect the feedback from the receivers in the subgroup. Only such special receivers report the state of the subgroup to the sender. Designated receivers in reliable multicast transport protocol (RMTP) [4] or link repliers in [7] have such function of repair. The scope of loss recovery is limited to a small subgroup. Thus feedback implosion and bandwidth consumption can be substantially reduced.

End-to-end and router assisted schemes have been employed in hierarchical loss recovery [4]-[9] and compared in [49]. Due to the lack of information about the underlying network,

end-to-end schemes often multicast or unicast repair packets to all members of a subgroup, e.g., reliable multicast transport protocol (RMTP) [4] uses designated receivers (DR) to remulticast repair packets for each subgroup. Router-assisted schemes further reduce the scope of recovery, e.g., Pragmatic General Multicast (PGM) [9] conducts retransmissions to the receivers that requested them because routers can remember where NAKs (negative acknowledgement) come from. If desired, suitable receivers are used to act as Designated Local Retransmitters (DLRs). Naturally, the placement of these DRs or DLRs has a noticeable influence on the bandwidth consumption of a network. Their optimal placement in a hierarchy, where a repair packet is always multicast to all members of each subgroup, has been obtained for a few special cases. These include uniform packet loss and linear topologies in [12] and for the general topologies and uniform packet loss in [69], where bandwidth consumption is minimized to obtain the optimal locations. Furthermore, the effects of topology on optimal locations have been analyzed in [62]. Research on the location problems can also be found for server-based reliable multicast [19] that presents only simulation results not related to any topology and for independent unicast transmissions from the sender to receivers [84]. Although the approaches proposed in [19] yield better multicast performance than the random placement method of repair servers, they do not aim to find the optimal placements. These placements are determined based on the number of the shortest paths passing through the node, the number of links connected to the node and the average number of hops from the node to all other nodes. Different loss

recovery schemes employed in different multicast protocols yield different values of bandwidth consumption. In this chapter, we address the problem of finding the effects of loss recovery on the locations of repair services. Here we use RR (repair router) to represent special routers having repair functions. An analytical comparison of two typical loss recovery schemes will be investigated in this chapter.

The rest of this chapter is organized as follows. In section 3.2, we present our model for hierarchical reliable multicast. In section 3.3, we study the performance of reliable multicast and give an evaluation on the number of transmissions and bandwidth consumption. In section 3.4, we use actual bandwidth consumption to optimize the placements of RRs for the binary tree. In section 3.5, we present the results of the optimal placements of RRs for binary tree. Section 3.6 concludes this chapter.

3.2 Model

The performance analysis herein is based on a single-source multicast tree. Here we use a RR to retransmit the repair packet to a subgroup for end-to-end and router assisted loss recovery schemes. We only consider the transmission of data packets by the sender or RRs. Each RR is located at the root of its subgroup. When one or more receivers do not receive the packet, i.e. the detection of out-of-order packets or the use of time out, the RR or the sender retransmits the missing packet.

3.2.1 The Functions of RRs

- Store-and-forward. For a new data packet from the sender, RR will always store and forward to its members. Duplicate packets that the RR has already received are not forwarded to its members.

- The retransmission of data. Only if one or more members require the retransmission of a missing data packet, will the RR retransmit it. Similarly, the sender retransmits packets to the requesting members in the sender subgroup not covered by any RR.

- The feedback of the domain. If the RR does not receive a data packet, i.e. the detection of out-of-order packets or the use of time out, it will ask for retransmission from the sender. The RR is an ordinary member in the domain of the sender.

3.2.2 The Partitioning of the Group:

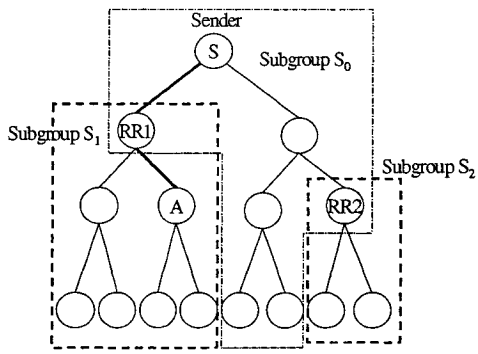


Fig. 3.1 The partition of a multicast group

Each RR manages a group of receivers and is responsible for retransmissions of lost

packets. Denote by S_r the subgroup covered by RR_r. Placing r RRs results in r+1 subgroups that are covered by r RRs and the sender, respectively. Fig. 3.1 is an example of the partition of a multicast group having 2 RRs where subgroup S_0 belonging to the sender is the subgroup not covered by any RRs.

3.2.3 Loss Recovery Policies

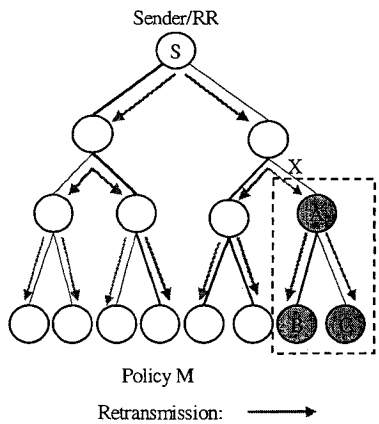


Fig. 3.2 The retransmission mechanism of policy M. The loss occurs at node A.

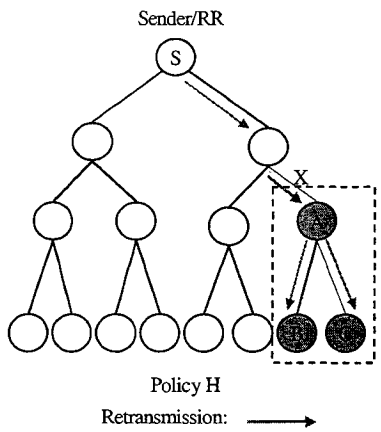


Fig. 3.3 The retransmission mechanism of policy H. The loss occurs at node A.

We focus mainly on the transmissions of data packets for two loss recovery policies

discussed in this chapter.

Policy M – Whole subgroup retransmissions

Receivers missing packets send NAK upstream to the sender or RR to request retransmissions. The sender and/or RRs multicast the lost packet to all members in their domain once they receive the retransmission requests (NAK) from receivers below [4]. Fig. 3.2 shows the loss recovery procedure of policy M where nodes B and C have not received a certain packet. This is a typical end-to-end scheme easily employed in reliable multicast.

Policy H – Selective retransmissions

When one receiver encounters a loss, it sends NAK to a sender or RR depending on which is closer. The sender and/or RR then retransmit the packet only to these receivers that did not receive data in previous transmissions. This protocol needs the assistance of routers to recover the packet loss. Some simple functions are added, so intermediate routers remember which link downstream loses the packet and needs retransmissions. Intermediate routers will retransmit packets only to some members of their subgroups. This policy adds a small extra capability to ordinary routers. They should be able to route repair packets only to these nodes in their subgroup that have sent a NAK upstream through this router. But they would send the repair packets to all members of this subgroup if they themselves have lost the original packet. Fig. 3.3 shows the loss recovery of policy H.

3.3 Performance Analysis

In this section, we will explain the two metrics that will be used to optimize the location of RRs, i.e. the expected number of transmissions E[M] and effective bandwidth consumption. E[M], which is the average number of transmissions of a given packet by the sender until all group members receive it correctly, is one of the most important parameters for reliable multicast. It depends on the topology of trees and loss probability [36][40][62][69][76].

E[M] gives the cost of the sender in the transmission of a packet to all receivers in the tree. The actual bandwidth consumption of networks depends not only on the number of transmissions E[M] but also on the number of links traversed by each transmission. We define E[B] as the expected bandwidth consumption averaged over all links in one successful multicast. We define E[C] as the total expected bandwidth of a group consumed by one successful source packet multicast. Thus,

$$E[B]=E[C]/N \tag{3.1}$$

where N is the total number of links for a multicast group. E[M], E[B] and E[C] reflect the bandwidth performance of the multicast protocol. In the following, we first evaluate their values for full binary trees. Please refer to Table 3.1 for the various notations used in this chapter.

3.3.1 E[M] of Binary Trees – the Exact Solution

The expected number of transmissions E[M] that a packet should be multicast by the

source using the selective reject ARQ until all group members receive it correctly can be recursively calculated . However, the recursive computation of E[M] may be very intensive for a general topology [36], [40], [62]. A reduction technique, which finds the equivalent link of any subtree, is used to compute E[M] [12]. The equivalent link has the same loss behavior in terms of E[M] on average as the subtree it substitutes for, i.e. the sender transmits E[M] times on average for the equivalent link or the subtree replaced by the equivalent link. The equivalent link has the equivalent loss probability p_e on the link or equivalently the number of transmissions, i.e. $E[M]=1/(1-p_e)$.

Star and linear topologies are the two basic units in a general tree. A tree can be decomposed into few star and linear topologies. In the sequel, any tree can be reduced step by step to one equivalent link. The equivalent link of star and linear topologies has been derived in [12].

- **A linear topology with 2 links**

  In Fig. 3.4, if receivers 0, 1, 2 are in the same subgroup, they can be made equivalent to a link with the equivalent packet loss probability p_e. The following gives the average number of transmissions to success [12].

$$p_e = 1-(1-p_1)(1-p_2) \qquad (3.2)$$
$$E[M] = \frac{1}{1-p_e} = \frac{1}{(1-p_1)(1-p_2)}$$

  where p_1 and p_2 are packet loss probability for link 1 and 2, respectively.

- **A star topology with 2 links**

For 2 links in a star topology as shown in Fig. 3.5, the number of transmissions is given by [12].

$$E[M]=\frac{1}{1-p_e}=\frac{1}{1-p_1}+\frac{1}{1-p_2}-\frac{1}{1-p_1p_2} \tag{3.3}$$

where p_1 and p_2 are packet loss probability for link 1 and 2, respectively.

The equivalent loss probability of general star topology with N links is complex. Its computation is also intensive. It can be 2^N where N is the total number of links [12], [13].

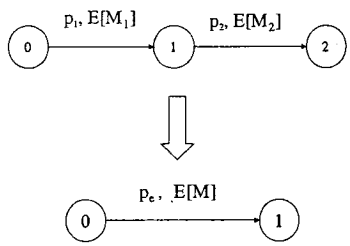


Fig. 3.4 The equivalent link of a linear tree with 2 links.

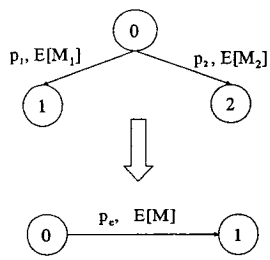


Fig. 3.5 The equivalent link of a star tree with 2 links.

Starting with the above formulas, one may recursively calculate the number of transmissions for a binary tree. Assume that we consider a binary tree with the homogeneous loss probability p. The binary tree of depth h is equivalent to one link with loss probability p_h. According to the reduction techniques in [12], one can successively reduce the number of levels from h to 1. As an example, Fig. 3.6 (b) is obtained from Fig. 3.6 (a) by defining p_{h-1} as the equivalent loss probability of the subtree of depth h-1. Finally, the binary tree is equivalent to a 2-star topology of loss probability q_h in Fig. 3.6 (c). One can then obtain the equivalent loss probability of 2-star and linear topologies from [12].

$$\frac{1}{1-p_h} = \frac{2}{1-q_h} - \frac{1}{1-q_h^2} \tag{3.4}$$

$$q_h = 1-(1-p)(1-p_{h-1}) \tag{3.5}$$

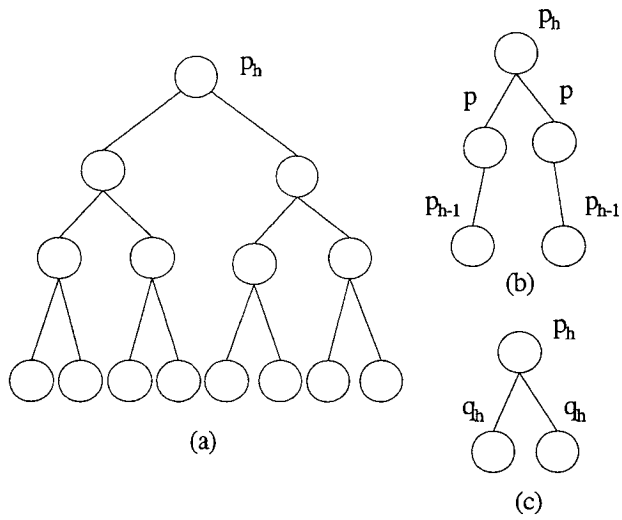


Fig. 3.6 The reduction technique for a binary tree

For a binary tree of depth h, one can calculate the expected number of transmission $E[M_h]$ from the equivalent link.

$$E[M_h] = \sum_{m=1}^{\infty} m \cdot p_h (1 - p_h)^{m-1} = \frac{1}{1 - p_h} \tag{3.6}$$

Therefore, we can obtain $E[M_h]$ for the whole binary tree by substituting (3.6) and (3.5) into (3.4).

$$E[M_h] = \frac{E[M_{h-1}]}{1-p}\left(2 + \frac{1}{\frac{1-p}{E[M_{h-1}]} - 2}\right) \tag{3.7}$$

Table 3.1 Notations for Reliable Multicast

ID	Meaning
S_i	The set of nodes covered by RR i.
S_0	The set of nodes not covered by any RR.
N	The total number of links within the group
N_i	The number of links within subgroup S_i ,i=0,1,...
$i, i_1, i_2, ...$	The level of placement of RR
$j, j_1, j_2 ...$	The number of independent losses for one packet in a transmission trial.
M, m	The total number of packet transmissions per source packet (due to losses)
$P(M=m)$	The probability that one packet multicast is successful in the m^{th} transmission (all nodes get the same packet).
$P(m\|j)$	The probability that multicast is successful in the m^{th} transmission after j losses take place.
$Q(j)$	The probability that j losses take place in one transmission over different

	links.
$P(j_0 j_1 .. j_r)$	The joint probability of S_0 having j_0 losses,S_1 having j_1 losses,..., S_r having j_r losses in the first multicast transmission of a given source packet.
C	The total bandwidth consumed for the multicast group (successful delivery of one source packet to all nodes including all retransmission.)
B	Average bandwidth consumption per link for the multicast group. B=C/N.
C_i	The Bandwidth consumption for subgroup S_i, same description as C.
B_i	Average bandwidth consumption per link for subgroup S_i, $B_i = C_i / N_I$
$E[M_i]$	The expected number of transmission times for all the nodes of subgroup S_i to receive correct source packet.
$E[C_i]$	The total expected bandwidth consumed for subgroup S_i member over all transmissions.
$E[B_i]$	The expected bandwidth consumed per link for subgroup S_i, $E[B_i]=E[C_i]/N_I$
$E[C]_{i_1 \ldots i_r}$	The total expected bandwidth for the whole multicast group S when r RRs are used.
$E[B]_{i_1 \ldots i_r}$	The expected bandwidth per link for the whole multicast group S when r RRs are used.
p	The packet loss probability per link.
d_k	The number of links from the sender to node k.
n_k	The total number of links under node k.

x, (x_i)	The average number of links that need retransmissions if one loss takes place within a multicast group S (or S_i), i.e. $x = \sum_{k=1}^{N} (d_k + n_k)/N$.
k	The sequence number of a node.
l	The level of a node.

3.3.2 E[M] of Binary Trees – An Approximate Evaluation

Although one may recursively calculate the expected number of transmissions for a binary tree using (3.7), the computation of E[M] for a general topology is intensive [36]-[40],[62]. Even if a reduction technique is used, the computation of E[M] can be exponential in the number of links N [12], thus defeating real time allocation of RRs as will follow in chapter 5. Thus, we try herein to find an approximate solution for the evaluation of the expected number of transmissions for general topologies. Because the loss sharing in multicast trees is modeled well by a full binary tree [40], we first derive the approximate solution for a full binary tree in the following.

Hierarchical reliable multicast is based on a tree topology where the loss of an intermediate link leads to the losses of all links below that link. Even if only one intermediate node loss takes place in a multicast tree, many links need retransmissions from the sender. In one multicast transmission, many losses may take place. Therefore, it is necessary to know how many losses take place and how many links are involved in retransmissions because only these nodes want to receive the repair packet in the next transmission. It is difficult to

obtain the exact number of nodes that actually need retransmissions. One may evaluate it according to the following steps.

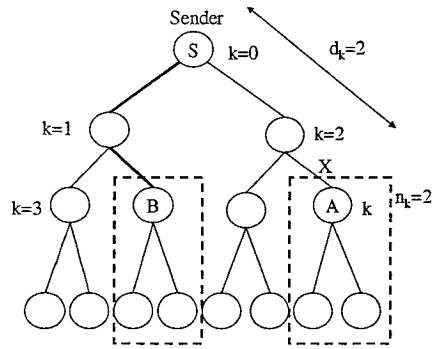


Fig.3.7 Many nodes need retransmissions if one link loss takes place

We assume that the probability of *j* independent losses for one transmission is binomially distributed for a full binary tree [40], [55]. In chapter 7, we will elaborate on an exact probability density for general trees over different loss scenarios. For the star topology, this assumption is exact because these losses are independent. For the other topology like a binary tree, multicast packet losses are not independent. In Fig.3.7, the loss of node A will result in the losses of all links below node A. These losses are dependent because they involve the nodes of subtree rooted at node A. However, if one node is not under the subtree of another node, the losses such as node A and B are independent. They do not affect each other. One may consider the effect of one loss on the other nodes of its subtree by introducing a parameter x as will follow while taking approximate binomially distributed probability of losses for other independent losses occurring outside the subtree mentioned above. Here *x* refer to as the number of links involved in retransmission for a

loss.

Suppose that each node has a sequence number and the sender has the sequence number 0, as shown in Fig.3.7. For a given packet transmission, if node k (e.g., node A) does not receive the packet, n_k links also lose the packet where n_k is the total number of links below node k. Thus, one needs to retransmit the packet to these n_k nodes. The repair packet must pass through the path from the sender to node k to recover the loss. The number of nodes to be retransmitted depends not only on the number of nodes that did not receive the packet but also on the number of nodes within the path from the sender to the link. Therefore, we only consider if these nodes receive the packet correctly in the next retransmission. For example, if node k loses a packet in Fig.3.7, the sender needs to retransmit the repair packet to at least d_k and n_k links where n_k is the total number of links under node k and d_k is the number of links from the sender to node k.

We define x to be the average number of links that need retransmissions for a multicast group if one loss takes place. From the above discussion, if node k loses a packet, the number of links involved in retransmission is d_k+n_k. Thus one may obtain x by averaging all links over the multicast group.

$$x = \sum_{k=1}^{N} (d_k + n_k) / N \tag{3.8}$$

where N is the total number of links, k is the index for nodes, n_k is the total number of the links below node k, d_k is the number of links from the sender to node k shown in Fig.3.7.

Here we use a simple mechanism, i.e. the use of x, to reflect on effect of topology.

When one link encounters a loss, x links need retransmission from the sender and/or RR. $\sum_{k=0}^{N} d_k$ is the summation of the number of links from the sender to all nodes. In order to calculate this, one can consider it in a different way. Suppose that each node has a counter whose value is 0 at the beginning. Every time one adds the value d_k for node k to $\sum_{k=0}^{N} d_k$, then the counters of these nodes (except node k) belonging to the path from the sender to node k will be incremented by 1. For example, $d_3=2$ in Fig.3.7, is equivalent to increment the counters of nodes $k=0$ and $k=1$ by 1. Finally, $\sum_{k=0}^{N} d_k$ is the summation of counters of all nodes. For a specific node (node j), its counter is determined by the number of downstream links n_j, i.e, its counter will be increased by n_j times. Thus, $\sum_{k=0}^{N} d_k$ is the summation of n_j for all nodes including the sender (node 0), i.e,

$$\sum_{k=0}^{N} d_k = \sum_{k=0}^{N} n_k \quad \text{or} \quad \sum_{k=1}^{N} d_k = \sum_{k=1}^{N} n_k + N \tag{3.9}$$

As an example for $N=14$ in Fig.3.7, $\sum_{k=0}^{N} d_k = 1*2+2*4+3*8=34$, $\sum_{k=0}^{N} n_k = 14+2*6+2*4=34$. Therefore, one may obtain the value of x by substituting (3.9) into (3.8).

$$x = 2\sum_{k=1}^{N} d_k / N - 1 = 2\sum_{k=1}^{N} n_k / N + 1 \tag{3.10}$$

n_k of a binary tree only depends on the level of node k, i.e., $n_k=2^{L-t+1}-2$ where t is the level of node k, so one may obtain the x value of a binary tree.

$$x=2L-3+4L/N \tag{3.11}$$

where L is the total depth of the tree.

The probability density function $P(M=m)$ representing the number of the final successful transmission trials can be obtained recursively, i.e.

$$P(M=1)=(1-p)^N \tag{3.12}$$

$$P(M=m)=\sum_{j\neq 0}^{N} Q(j)P(M=m-1|j), \qquad m=2,3,... \tag{3.13}$$

where, for analysis convenience, $Q(j)$, i.e, the probability that j losses take place for one multicast transmission, is assumed to be approximately binomially distributed.

$$Q(j)=\binom{N}{j}p^j(1-p)^{N-j} \tag{3.14}$$

where N is the total number of links in the whole network, p is the link packet loss probability. Taking this analysis assumption does not mean we will not consider the subsequent dependent losses. Actually starting with this independent assumption and then evaluating x for the randomizing j failures amount to a worst case analysis as has been verified in [69]. The conditional probability $P(M=m-1|j)$ is the probability that $m-1$ retransmissions are required to recover j losses. It can be recursively calculated similar to (3.12) and (3.13), but only jx not N nodes are involved in retransmissions if j losses take place.

Thus, one can easily obtain the expected number of total transmissions, which can be recursively calculated.

$$
\begin{aligned}
E[M] &= \sum_{m=1}^{\infty} mP(M=m) \\
&= P(M=1) + \sum_{m=2}^{\infty} m \sum_{j\neq 0}^{N} Q(j)P(M=m-1 \mid j) \\
&= P(M=1) + \sum_{j\neq 0}^{N} Q(j)[1 + \sum_{m=2}^{\infty} (m-1)P(M=m-1 \mid j)] \\
&= 1 + \sum_{j\neq 0}^{N} Q(j)E[M \mid j]
\end{aligned}
\tag{3.15}
$$

where E[M|j] is the expected number of transmissions to correct j losses. Only these nodes that did not receive the packet are involved in retransmissions, i.e. jx links are involved in retransmissions in the worst case due to j independent losses. So, E[M|j] can also be similarly obtained.

$$
E[M \mid j] = 1 + \sum_{j_1\neq 0}^{jx} Q(j_1)E[M \mid j_1] \tag{3.16}
$$

One may use (3.15) to recursively calculate E[M]. E[M] of a real multicast network should have small value, e.g, $E[M]<2$, so, one may only consider the impact of a couple of retransmissions. Substituting (3.16) into (3.15), one may obtain the following approximation for E[M].

$$
E[M] = 2 - (1-p)^N + \sum_{j\neq 0}^{N} Q(j) \sum_{j_1\neq 0}^{jx} Q(j_1)E[M \mid j_1] \tag{3.17}
$$

Applying E[M|j_1] again to (3.17), i.e., $E[M \mid j_1] = 1 + \sum_{j_2\neq 0}^{j_1 x} Q(j_2)E[M \mid j_2]$ and taking $(1-p)^{jx} \approx 1 - jxp$, we obtain

$$
E[M] \approx 2 - (1-p)^N + xNp^2 + \sum_{j\neq 0}^{N} Q(j) \sum_{j_1\neq 0}^{jx} Q(j_1) \sum_{j_2\neq 0}^{j_1 x} Q(j_2)E[M \mid j_2] \tag{3.18}
$$

If the packet loss probability p is low, one obtains further approximation from (3.18) by

ignoring the higher order terms of p.

$$E[M] \approx 1 + Np + [xN - \binom{N}{2}]p^2 + \binom{N}{3}p^3 \tag{3.19}$$

E[M] depends on N, p and x. x is the quantity related to topology in (3.8) or (3.10). Different topologies have different x values. The x value for binary tree can be found in (3.11). The x values for linear and star topologies are their depth, i.e., N and 1.

3.3.3 The Evaluation of Bandwidth Consumption

Although E[M] gives the expected number of transmissions, the number of links affected by each transmission is different. Thus, it is necessary to evaluate the bandwidth consumed on all links till final success. By final success, we mean to include all link transmissions used till all nodes receive the correct packet. A real multicast network has small E[M] value, so the sender or RR may transmit the packet only at most a couple of times. In the first transmission, if node k does not receive the packet, n_k links will lose the packet as shown in Fig.3.7. The average number of links not affected by the first transmission is $\sum_{k=1}^{N} n_k / N = (x-1)/2$ (see (3.9)). Then the actual bandwidth consumption is $N-(x-1)/2$ for 1 independent loss. The actual bandwidth of j losses is $N-j(x-1)/2$. Thus, we may estimate the expected bandwidth as follows.

$$\begin{aligned} E[C] &= \sum_{j=0}^{N} Q(j)\{N - j(x-1)/2 + E[C \mid j]\} \\ &= N - \frac{1}{2}N(x-1)p + \sum_{j=0}^{N} Q(j)E[C \mid j] \end{aligned} \tag{3.20}$$

where $Q(j)$ is the probability of having j losses in a transmission trial as defined in Table 3.1, $E[C|j]$ is the extra bandwidth consumed in all retransmission trials to correct the j losses that take place in the first transmission. As a worst case, one may take N as the number of nodes that received the packet correctly. In a real multicast network that requires only a few transmission trials to recover loss, this is the close approximation for many topologies such as binary trees and star trees, in which $(x-1)p/2<<1$. Thus, we can upper bound the above equation as follows.

$$E[C] \le N + \sum_j Q(j) E[C \mid j] \tag{3.21}$$

In order to evaluate the expected conditional bandwidth consumption, i.e. E[C|j], we encounter two cases depending on the recovery methodology, i.e., policy M and H to follow.

Policy M

In this loss recovery scheme, the packet is always multicast to the whole node population by the sender or RR, i.e., each transmission or retransmission affects the same multicast group. The actual bandwidth of each retransmission is $N[1-(x-1)p/2]$. Thus, one obtains

$$E[C|j] = E[M|j]N[1-(x-1)p/2] \tag{3.22}$$

where E[M|j] is the expected number of retransmissions after j losses take place.

By substituting (3.22) into (3.20), one obtains E[C] and E[B].

$$E[C] = N[1-(x-1)p/2]\cdot\{1+\sum_{j\neq 0}^{N} Q(j)E[M \mid j]\} = E[M]\cdot N[1-(x-1)p/2] \tag{3.23}$$

$$E[B] = E[M]\cdot[1-(x-1)p/2] \tag{3.24}$$

One can also upper bound the bandwidth consumption as follows,

$$E[B] \leq E[M] \tag{3.25}$$

Policy H

For each retransmission, the sender and/or RRs will retransmit the packet to only these receivers that did not receive data in previous transmissions. Thus, only the receivers who did not receive the packet in the previous transmissions will obtain the retransmission from the sender or RR. Each retransmission affects different nodes. Although it requires routers to recover losses, it offers a good method to estimate the minimal bandwidth consumed in loss recovery. For a successful multicast, the bandwidth consumption is at least N per source packet. Each retransmission will lead to extra bandwidth consumption. For example, if node k (node A) does not receive the packet as shown in Fig.3.7, it will ask the retransmission from the sender or the RR. Then, the bandwidth consumed in the first transmission is N-n_k, where n_k is total number of links under node k. The number of links involved in next transmissions is d_k+n_k for the loss from link k. The average number of links involved in the loss recovery is $\sum_{k=1}^{N}(d_k + n_k)/N = x$ for one loss and jx for j losses. Thus, the conditional bandwidth for the next transmission can be recursively obtained similar to (3.20).

$$E[C \mid j] = jx - \frac{1}{2} jxp(x-1) + \sum_{j_1}^{jx} Q(j_1) E[C \mid j_1] \tag{3.26}$$

From the above discussion, the conditional bandwidth consumed in retransmissions for policy H depends on the number of losses while the bandwidth consumed in policy M only depends on whether losses take place or not.

One may estimate the bandwidth consumed in protocol H by substituting (3.26) into (3.20).

$$E[C] = N + \frac{1}{2} Np(x+1) - \frac{1}{2} xNp^2(x-1) + \sum_{j \neq 0}^{N} Q(j) \sum_{j_1 \neq 0}^{jx} Q(j_1) E[C \mid j_1] \tag{3.27}$$

where N is the number of links, p is the packet loss probability, and x can be obtained from (3.10). If only a couple of transmissions are considered, the losses are assumed to be recovered by the third transmissions in the expected sense, i.e. we can assume $E[C|j_1]=j_1x$. Thus, substituting $E[C|j_1]=j_1x$ into (3.27), one can obtain the following approximation.

$$E[C] \approx N + \frac{1}{2} Np(x+1) + \frac{1}{2} xNp^2(x+1) \tag{3.28}$$

Substituting (3.27) into (3.1), one can obtain the expected bandwidth E[B] consumed per link for policy H.

$$E[B] \approx 1 + \frac{1}{2} p(x+1) + \frac{1}{2} x(x-1)p^2 \tag{3.29}$$

3.4 Effects of RRs on Bandwidth

Suppose that the binary tree is a uniform tree, i.e., all the links have the same packet loss probability p. If N and p are large for this topology, the sender needs to retransmit many times to have a successful multicast. In order to reduce the cost and the number of

retransmissions, we may put RRs to provide the FEC/ARQ capacity. Suppose that we prepare to put *r* RRs. We will show how bandwidth will change with the locations of these RRs. First we use *1* RR to derive the estimation of bandwidth consumption.

3.4.1 1 RR

Assume that this RR is in level i of a binary tree with the height of *L*. N_i is the number of links beneath the RR node, that is, $N_i=2^{L-i+1}-2$. For the first multicast transmission, there may be no error (with the probability $(1-p)^N$). Bandwidth consumption is easy to compute, which is only *N* links.

However, some receivers may not receive the packet in the first transmission. These receivers may be beneath the RR node down the multicast tree, or they may be located outside the RR domain. Different users that do not receive the packet have different locations and retransmission sources. This means that retransmissions depend on the location of receivers that do not receive the packet. The RR is only responsible for the retransmission of the lost packet within its domain. The probability α_1 that the loss falls into the domain of RRs will be

$$\alpha_1 = \frac{N_1}{N} \tag{3.30}$$

Thus, the probability α_0 that the loss falls into the domain of the sender is

$$\alpha_0 = \frac{N - N_1}{N} = 1 - \alpha_1 \tag{3.31}$$

Assume that *j* losses take place in the first transmission. Out of these *j* losses, j_0 losses fall

within the domain of the sender and j_1 losses fall within the domain of RR, i.e. $j=j_0+j_1$. Therefore, the probability that j_0 losses are located within the domain S_0 and j_1 losses are located within the domain S_1 in the first multicast is

$$P(j_0, j_1) = \binom{N}{j_0 + j_1} p^{j_0+j_1}(1-p)^{N-j_0-j_1} \binom{j_0 + j_1}{j_1} \alpha_1^{j_1} \alpha_0^{j_0} \tag{3.32}$$
$$= \frac{N!}{j_0! j_1! (N - j_0 - j_1)!} (\alpha_0 p)^{j_0} (\alpha_1 p)^{j_1} (1-p)^{N-j_0-j_1}$$

One can obviously see that $P(j_0,j_1)$ is multinomially distributed. The total bandwidth consumed in the whole group is obtained recursively

$$E[C]_i = \sum_{j_0}^{N_0} \sum_{j_1}^{N_i} P(j_0, j_1)\{N(j_0) + N(j_1) + E[C \mid (j_0, j_1)]\} \tag{3.33}$$

where $N(j_0)$ and $N(j_1)$ are the actual bandwidth consumed in subgroup S_0 and S_1 in the first multicast transmission. $E[C|(j_0,j_1)]$ is the expected total bandwidth when j_0 losses take place within domain S_0 and j_1 losses take place within domain S_1. These losses are recovered by either the RR or the sender, depending on which subgroup includes these losses. If the lost nodes are located under the RR in the multicast tree, the RR will recover these losses. Otherwise, if the lost nodes are located outside the RR, the sender will recover these losses. The RR is not responsible for the loss recovery of j_0 losses located outside the RR. Similarly, the sender will not recover losses from j_1 losses located under the RR. The sender and the RR work independently in the next retransmissions. Thus the total bandwidth $E[C|(j_0,j_1)]$ can be divided into the summation of bandwidth $E[C| j_0]$ and $E[C| j_1]$ for two subgroups:

$$E[C|(j_0,j_1)]=E[C|j_0]+E[C|j_1] \tag{3.34}$$

$E[C|j_1]$ is the total conditional bandwidth consumed by subgroup S_1 when j_1 losses located under the RR. $E[C|j_0]$ is the total conditional bandwidth consumed by subgroup S_0 when j_0 losses located outside the RR.

One may obtain the following results for the total bandwidth consumption by substituting equation (3.34) into (3.33).

$$\begin{aligned} E[C]_i &= \sum_{j_0}^{N_0}\sum_{j_1}^{N_1} P(j_0,j_1)\{N(j_0)+E[C_0 \mid j_0]\} \\ &+ \sum_{j_0}^{N_0}\sum_{j_1}^{N_1} P(j_0,j_1)\{N(j_1)+E[C_1 \mid j_1]\} = E[C_0]+E[C_1] \end{aligned} \tag{3.35}$$

$$E[B]_i = \frac{E[C]_i}{N} = \alpha_0 E[B_0]+\alpha_1 E[B_1] \tag{3.36}$$

The above equation shows that the total bandwidth consumed in the whole group equal the summation of bandwidth consumed in every subgroup.

Based on the discussion above, the total bandwidth equal the summation of bandwidth consumed in two subgroups, i.e., $E[C_0]$ and $E[C_1]$. As discussed previously, the evaluation of bandwidth $E[C]$ depends on the policy of loss recovery. Similarly, we will obtain the corresponding bandwidth consumption per link $E[B]$ as follows.

3.4.1.1 Policy M

We multicast repair packets to the subgroup after losses take place in one subgroup. We assume that each subgroup is independent of each other. Therefore, the losses within the

RR_1 influence only the subgroup S_1. Because the RR does not forward the same data from the sender twice, the losses outside the RR_1 do not influence the bandwidth consumption of subgroup S_1, but they affect S_0. Their maximal affected links are N_0 and N_1 respectively for each multicast. Therefore, one may obtain the following bandwidth consumption per link from (3.36).

$$E[B]_i = \alpha_0 E[M_0][1-(x_0-1)p/2] + \alpha_1 E[M_1][1-(x_1-1)p/2] \tag{3.37}$$

where $E[M_0]$ and $E[M_1]$ are the expected number of transmission times for the subgroup S_0 and S_1, respectively. x_0 and x_1 are the topology parameters of S_0 and S_1, respectively. $E[M_0]$ and $E[M_1]$ can be easily found from equation (3.7) or evaluated from (3.19). α_i (i=0 or 1) can be calculated by (3.30) and (3.31).

3.4.1.2 Policy H

This case is available for the scenario with less loss. We retransmit the repair packet to these receivers that did not receive the packet. We use upper bound to estimate the bandwidth consumption per link.

$$\begin{aligned} E[B]_i &= \alpha_0 E[B_0] + \alpha_1 E[B_1] \\ &\approx \alpha_0[1+\frac{1}{2}p(x_0+1)+\frac{1}{2}x_0(x_0-1)p^2] + \alpha_0[1+\frac{1}{2}p(x_1+1)+\frac{1}{2}x_1(x_1-1)p^2] \end{aligned} \tag{3.38}$$

where $E[B_0]$ and $E[B_1]$ are the expected bandwidth consumed per link for the subgroup S_0 and S_1, respectively. i is the level of the RR. x_0 and x_1 are obtained from (3.10).

3.4.2 r RRs

When r RRs having FEC/ARQ capability are placed in the multicast group, the probability $P(j_0 j_1,...,j_r)$ that j_0, j_1 ,...., and j_r losses are located respectively within domain S_0 ,S_1 ,...., and S_r in the first multicast is multinomially distributed. The total bandwidth consumed in the multicast group is

$$E[C]_{i_1 i_2 \cdots i_r} = \sum_{j_0=0}^{N_0} \cdots \sum_{j_r=0}^{N_r} P(j_0, j_1, \cdots, j_r)\{N(j_0, j_1, \cdots, j_r) + E[C \mid (j_0, j_1, \cdots, j_r)]\} \tag{3.39}$$

where

$$P(j_0, j_1, \cdots, j_r) = \frac{N!}{j_0! j_1! j_2! \cdots j_{r+1}!} p_0^{j_0} p_1^{j_1} \cdots p_{r+1}^{j_r} \tag{3.40}$$

$$\begin{aligned} & j_0 + j_1 + \cdots + j_{r+1} = N \\ & p_0 = N_0 p / N, \quad p_1 = N_1 p / N, \quad \cdots \quad p_r = N_r p / N \\ & p_{r+1} = 1 - p \end{aligned} \tag{3.41}$$

N_k is the number of links within subgroup S_k, $k=0, 1, ..., r$. The total bandwidth consumed for the whole multicast group is the summation of bandwidth consumed by all subgroups in the same way we developed the case of one RR.

$$E[C]_{i_1 i_2 \cdots i_r} = \sum_{i=0}^{r} E[C_i] \tag{3.42}$$

where $E[C_k]$ is the expected bandwidth consumed in subgroup S_k, i_k is the levels of the k^{th} RR, $k=0,1,...,r$. Therefore, one can obtain the following bandwidth consumption for two cases.

3.4.2.1 Policy M

For each retransmission, the packet from the sender or RR is always multicast to the whole subgroup. Thus, one may obtain

$$E[B]_{i_1 i_2 \cdots i_r} = \sum_{i=0}^{r} \alpha_i E[M_i]][1-(x_i-1)p/2] \tag{3.43}$$

where $E[M_i]$ is the expected number of transmissions for subgroup S_i, x_i is the topology parameter of subgroup S_i, $\alpha_i=N_i/N$, $i=0,1,...,r$.

3.4.2.2 Policy H

In this case, repair is retransmitted only to these receivers that do not receive the packet. Therefore, total bandwidth can be obtained.

$$E[B]_{i_1 i_2 \cdots i_r} = \frac{1}{N}(\sum_{i=0}^{r} N_i E[B_i]) = \sum_{i=0}^{r} \alpha_i E[B_i] \tag{3.44}$$

where $E[B_i]$ is the expected bandwidth consumed per link for subgroup S_i, $\alpha_i=N_i/N$, $i=0,1,...,r$. It can be calculated according to (3.29).

3.5 Analysis Results

In a large multicast tree, if the background traffic (unicast traffic and other multicast sessions) is randomized over all links, and if link channel error effects are ignored, then the same link loss probability will prevail over all links. There will be little or no loss changes from link to link, and, since the traffic over all links is almost the same, multicast retransmissions from one sender and no RR will lead to the same uniform traffic detailed

above. Even in the case of using RRs for error correction, these RRs will handle the retransmissions in their subgroup while the sender handles the rest of the retransmissions, thus leading to the same uniform traffic detailed above, in that the total original and retransmissions traffic is uniformly distributed over all the links. However, if the background traffic (unicast plus other multicast sessions), and/or link errors are not uniform over all links, then one must assume a different loss probability for each link. This section will compute the optimal DR locations for the network with homogeneous (same) link loss probability.

As an example of optimizing the location of DRs, we use a binary tree to illustrate the performance of uniform loss probability. Suppose that the binary tree has depth L and the homogeneous link loss probability p. Thus the total number of links is $N=2^{L+1}-2$. For hierarchical reliable multicast, the senders and/or DRs must transmit the packet several times in order to achieve a successful multicast transmission.

Though we compute the results for few networks with depth up to 10 and for different topologies, the analysis results are more general and are easily computable and extendable to other cases.

From (3.7) or (3.19), one can easily calculate the number of transmissions. E[M] changes with the loss probability p, the topology parameter x and the total number of links N. Fig. 3.8 shows that the impact of topology on the number of transmissions E[M] is significant even if they have the same number of links N and the loss probability p. Different

topologies have different x parameters. With the increase of the total number of links N, E[M] for linear topology increases rapidly while E[M] for star and binary tree increases smoothly. One may find that (3.19) is a good approximation to (3.7) for linear, star and binary tree. The comparisons of E[M] for two approaches (3.19) and (3.7) are given in the case of binary tree proving that (3.19) is a very good approximate solution to (3.7), in Fig. 3.9. In general, (3.19) also applies to other topologies because x reflects the impact of topology. As long as E[M] is smaller than 2, (3.19) is a good approximation of E[M] for different topology [69].

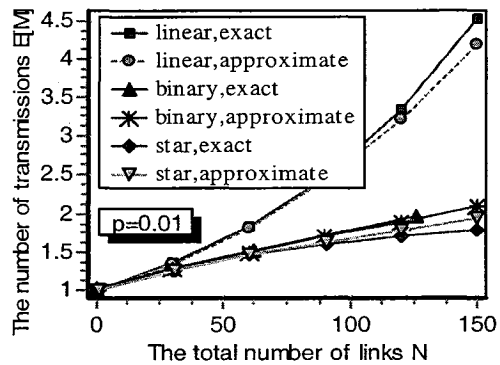


Fig. 3.8 A comparison of E[M] for different topologies.

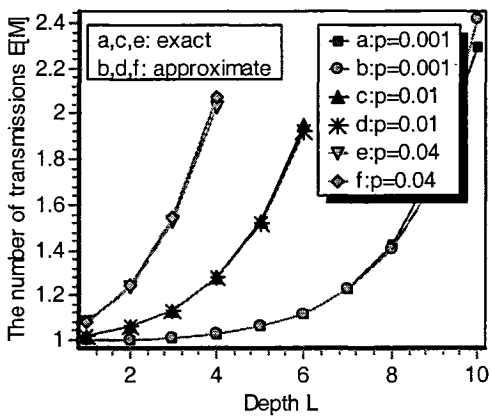


Fig. 3.9 A comparison of E[M] for binary topologies.

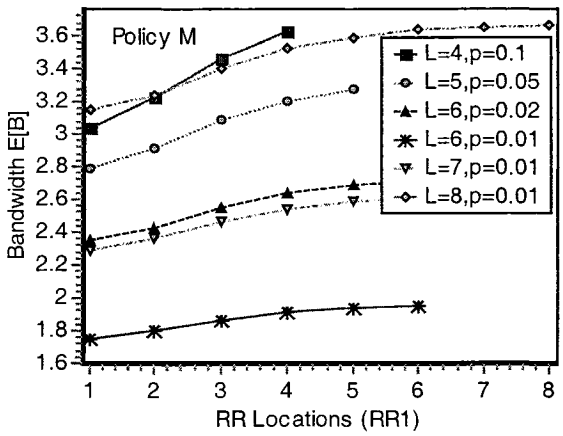


Fig. 3.10 The bandwidth E[B] of 1 RR for policy M.

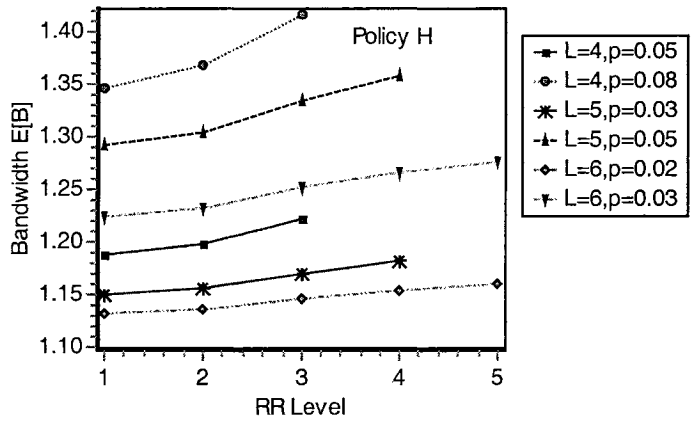


Fig.3.11 The bandwidth E[B] of 1 RRs for policy H

Based on the estimate of bandwidth consumption, the optimal placements of RRs are obtained. Fig. 3.10 - Fig.3.15 give the optimal placements of RRs for two policies. Fig. 3.10 and Fig.3.11 show that the best location of 1 RR is in level 1 for the two policies. When 2 RRs are used, the optimal locations are: one RR in level 1, another RR in level 2 for policy M, as shown in Fig.3.12. For policy H, the best locations of 2 RRs are in level 1. When 3 RRs are used, all 3 RRs are placed in level 2 for policy M to obtain the minimal bandwidth consumption, as shown in Fig.3.14. The best locations of 3 RRs for policy H are 1, 2, 2. The best locations of two policies are very close. These results are exactly the same

for the two methods of computing E[M], i.e. the exact and approximate solutions.

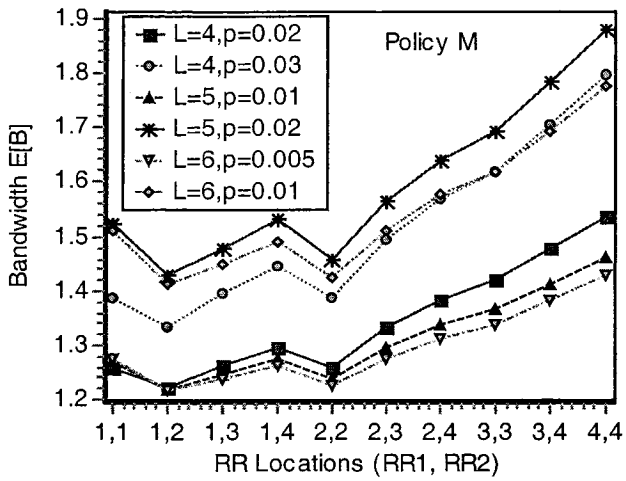


Fig.3.12 The bandwidth E[B] of 2 RRs for policy M

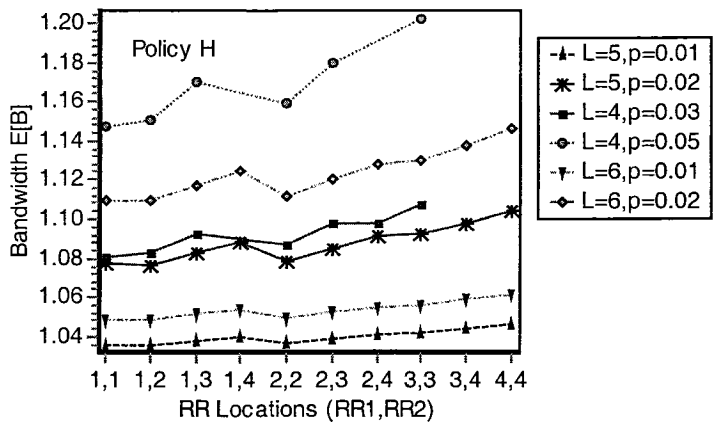


Fig.3.13 The bandwidth E[B] of 2 RRs for policy H

Some results with different tree depth and different loss probability are also given in Fig.3.12 - Fig.3.15. It is obvious that one can greatly reduce the bandwidth consumption by placing RRs in appropriate levels. If these RRs are not placed in appropriate locations, bandwidth consumption is large. From these curves, one can easily see that the bandwidth consumption increases with the increasing depth L and loss probability p, because the number of retransmissions steadily increases in tandem with N and p. However, the

optimal locations do not change with depth L and/or loss probability p, although they change with the number of RRs. As has been proven in the following chapter, this is true only for homogeneous binary topologies.

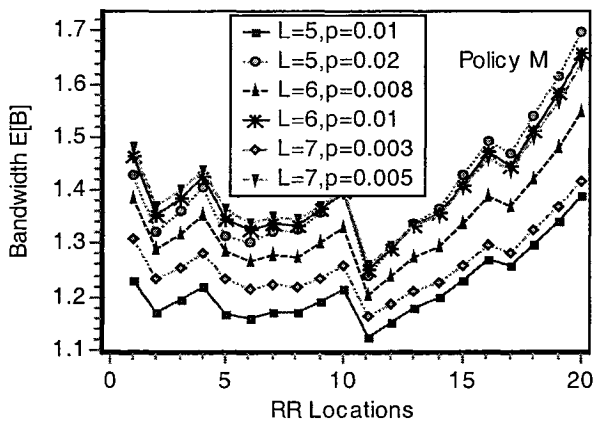


Fig.3.14 The bandwidth E[B] of 3 RRs for policy M. x axis is assumed levels of the 3 RRs as shown in Table 3.2.

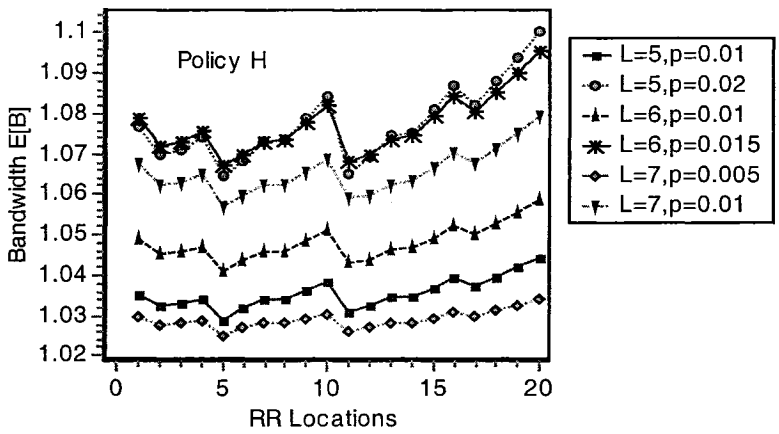


Fig.3.15 The bandwidth E[B] of 3 RRs for policy H. x axis is assumed levels of the 3 RRs as shown in Table 3.2.

This reinforces that the best RR allocation policy is one that divides the total network into r+1 equal parts, where r is the number of RRs such that the duty of error correction is

equally shared in homogeneous topologies. This equalized distribution is clearly independent of the number of links.

Table 3.2 The RR levels corresponding to the x axis of Fig.3.14 and Fig.3.15

X axis	1	2	3	4	5	6	7
RR levels	1,1,1	1,1,2	1,1,3	1,1,4	1,2,2	1,2,3	1,2,4
X axis	8	9	10	11	12	13	14
RR levels	1,3,3	1,3,4	1,4,4	2,2,2	2,2,3	2,2,4	2,3,3
X axis	15	16	17	18	19	20	
RR levels	2,3,4	2,4,4	3,3,3	3,3,4	3,4,4	4,4,4	

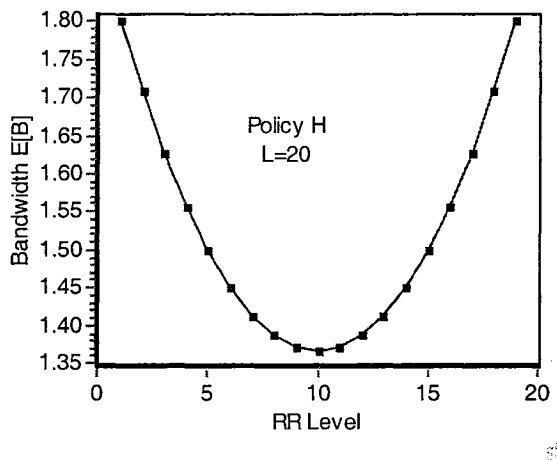


Fig.3.16 The bandwidth E[B] of 1RR for policy H in the linear topology.

The optimal RR locations for the two loss recovery policies give comparable results although policy H has less bandwidth consumption than policy M in the case of binary trees. The optimal partitioning for policy M also provides bandwidth consumption savings up to 40%. Policy H works well for binary trees with small packet loss probability even if

no RR is placed. Although the optimal placements of RRs can still be found, the advantages due to the retransmissions from RRs are not obvious. In this case, we do not place RRs to retransmit repair packets. However, policy H does not always work very well, and it still requires the use of RRs to recover losses for other topologies such as linear-like topologies. For example, policy H cannot recover efficiently loss of linear topology without the use of RRs, which will be discussed in chapter 7. Fig.3.16 shows the importance of RR location in Policy H for a linear topology where the bandwidth savings reach 30%.

Also evident is the fact that the closer the RR is to the receivers, the more bandwidth is consumed. If bandwidth consumption is to be minimized, RRs should not be placed too close to the receivers in binary trees because, although RRs close to the receivers consume less bandwidth, the sender will consume more, thus leading to a total bandwidth consumption that remains large. The optimal locations are therefore a balancing of bandwidth consumption between the sender and the RRs.

3.6 Conclusions

Based on the estimation of bandwidth consumed by the two policies, the optimal RR placements have been investigated in this chapter. The results based on the binary tree of uniform loss probability show that the best placements of RRs are closer to the sender rather than the receivers. For policy M, the best location is in level 1 when only 1 RR is used to recover loss. When 2 RRs are used, the optimal locations are: 1 RR in level 1, and

another RR in level 2. When 3 RRs are used, 3 RRs are placed in level 2 to obtain the minimal consumption of bandwidth. For policy H, the best location is in level 1 for 1 RR. When 2 RRs are used, the optimal locations are: both RRs in level 1. When 3 RRs are used, 2 RRs are placed in level 2 and one RR is placed in level 1 to obtain the minimal bandwidth consumption. The optimal placements for the two policies are very close though slightly different.

In this chapter, we have compared two loss recovery polices and their bandwidth consumption, i.e., policy M and H. The bandwidth consumption of policy H is much smaller than that of policy M, however, the optimal RR locations for the two cases are close. We thus coupled the choice of which policy to the locations and number of routers having the FEC/ARQ capability.

Finally, we have also derived a good approximation for the number of transmissions in the case of small E[M]. E[M] changes with N, p and x where x reflects the impact of topology on E[M].

Chapter 4 EFFECTS OF TOPOLOGY

4.1 Introduction

Loss recovery is crucial for reliable multicast [4]-[6], [25], [29], [52]. For tree-based multicast protocols, receivers are organized in a tree in which the sender is at the root, the receivers are at the leaves, and the domain representatives are at intermediate nodes of the tree. Some special receivers or routers, Designated Receivers (DR) in RMTP [4] for example, manage a group of receivers or a domain and are also organized in a hierarchical manner. In this chapter, we will use repair routers (RR), which have repair functions and handle retransmissions, to represent these special receivers or routers. These RRs will have Automatic Repeat Request (ARQ) retransmissions in case of packet loss.

In tree-based multicast protocols, RRs (similar in repair function to domain receivers or designated receivers) are placed at intermediate nodes of the tree in order to retransmit lost packets. In fact, the multicast group is partitioned into different subgroups according to these repair routers. Each subgroup resembles a small multicast group, and its repair router is the transmitter of each subgroup. The placement of these repair routers determines the partitioning of the multicast group; therefore, the multicast performance depends upon the placements of these repair routers.

Intermediate links are considered to be almost loss-free [18], [44], [71]; thus, because repair routers are typically placed close to receivers, RRs are assumed to always have the

correct packets. This mode of repair is considered to be a local recovery mechanism; however, these intermediate routers may lose multicast packets, and losses may still exist in intermediate nodes. For example, packets may be lost due to the buffer overflow of intermediate routers. Loss from an intermediate link will result in frequent retransmissions, a phenomenon whose impact on the multicast network cannot be ignored. In the old model of loss-free intermediate links, the effects of topology cannot be estimated. If such intermediate losses are considered, simulations in [9] proved that the active repair routers in WAN (Wide Area Network) will improve the performance when compared to situating the repair routers in MAN (Metropolitan Area Network) or LAN (Local Area Network). In fact, this can be considered to be a global loss recovery that is dependent on topology.

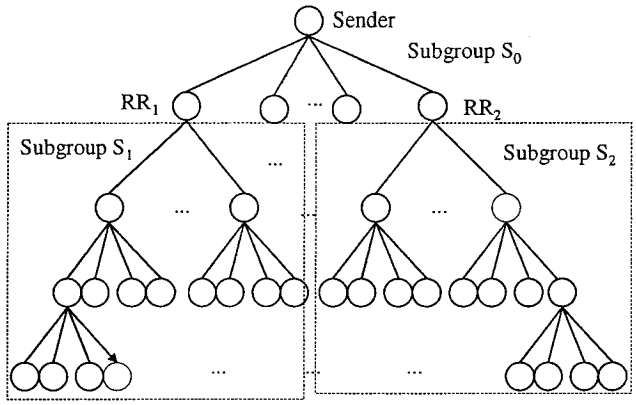
While the literature on general multicast [3]-[50] and FEC for multicast networks [37]-[46], [54], [58] is rich, no trial to find the effect of topology on the optimal RR locations has been undertaken. This, then, is the subject of this chapter. There have been some research results on the placement of repair routers or servers. Reference [19] suggests a few placement methods that improve multicast performance, compared to the random placement of repair routers; however, these placements are not optimal. References [69], [12] and [75] use performance analysis to optimize the placement of repair routers for multicast. In particular, [12] estimates the bandwidth consumption for linear topology, but does not consider the impact of topology on bandwidth consumption and the optimal placement of RRs. [62] evaluated the effect of topology type on multicast performance, but it applied no RR

location and did not consider the effects of RR. The optimal placements of repair routers clearly depend on network topology. In this chapter, we will focus on the effects of topology on multicast performance. We will use analytical models to compute the number of packet transmissions. The sender or repair servers will then apply the findings to the selection of the best locations for repair routers, based on the loss models of the intermediate links, in order to minimize bandwidth consumption.

This chapter is organized as follows: in section 4.2, we present our model for hierarchical reliable multicast; in section 4.3, we study the multicast performance of some simple topologies; in section 4.4, we give an analytical estimate of multicast performance for general topologies; in section 4.5, we minimize bandwidth consumption by choosing placement of RRs; and in section 4.6, we present the results of optimal RR placements. Section 4.7 concludes the chapter.

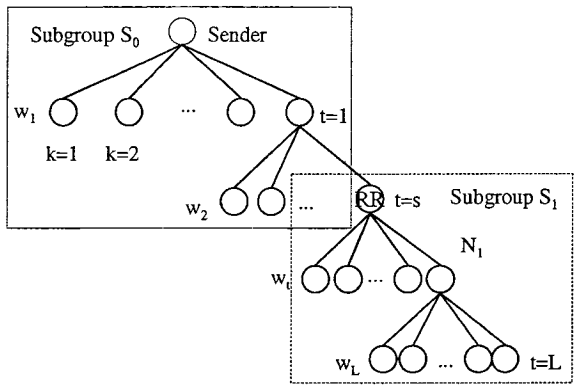
4.2 Model

We will investigate the performance of multicast networks for several types of topologies, as shown in Fig. 4.1, Fig. 4.2, Fig. 4.3, Fig. 4.4, and Fig. 4.5. For type A, each intermediate node has the same number of children nodes, i.e. w children nodes. For type B, only one node in each level has children links, and the number of links for each level is different. If the width of type B is small, it has a more clearly linear topology. Type C is one example of general topologies. Type D is a linear topology. Type E is a star topology. For these types of topologies, each link is assumed to have the same packet loss probability p.



(a) w: the number of branches for each node
L: depth of tree

Fig. 4.1 Topology of type A -- a k-ary tree. Each intermediate node has the same number of children links.



w_t: the number of branches for level t
L: depth of tree

Fig. 4.2 Topology of type B. Only one node in each level has children nodes.

RRs are placed in or attached to intermediate nodes in order to recover losses from the nodes below; we therefore obtain different subgroups according to these RRs. Fig. 4.1 gives an example in which two RRs are responsible for two subgroups (S_1 and S_2) and the sender is responsible for the group that is not covered by the two RRs (S_0). These RRs work

independently. As long as RR gets the packet, it functions like a sender for its subgroup; it may multicast the repair packets to the whole subgroup after it receives requests for packet retransmissions (NAK). If RRs also lose the packet, they will ask for retransmissions from the sender. Thus, RRs function both as receivers and as senders. This chapter investigates reliable multicast for the backbone network that connects all routers. The analysis of the final hop (from the routers to the receivers) is straight and is an assumed ideal; therefore, it will not be discussed in this chapter.

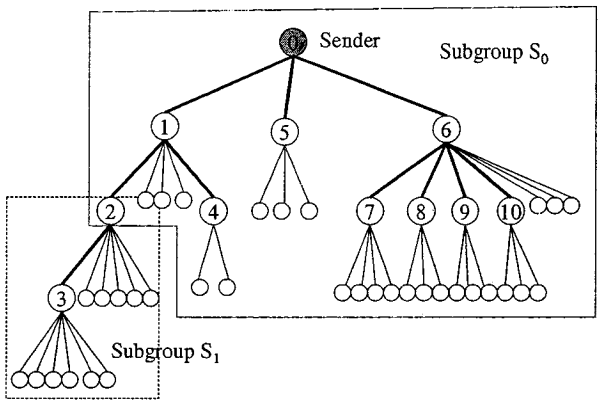


Fig. 4.3 Topology of type C -- an example of a general multicast network

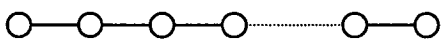


Fig. 4.4 Topology of type D – a linear topology with N links.

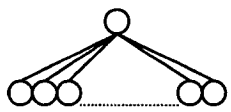


Fig. 4.5 Topology of type E – a star topology with N links.

4.3 Performance Analysis

Bandwidth consumption is a very important parameter that has been defined in [40], [71]-[73]. Although the actual bandwidth consumption is difficult to determine, calculating the number of transmissions and retransmissions provides a very efficient estimate. For reliable multicast, bandwidth consumption depends on many factors, such as loss recovery and network topology, the impact of which must be considered in order to find the optimal placements of RRs. In the following, we will discuss the effect of network topology on the optimal placements of RRs in reliable multicast communications.

For a multicast network without RR-based loss recovery, one can calculate the CDF (cumulative distribution function) of the total number of transmissions and retransmissions from the sender. Let $M(n)$ be the total number of the transmission and retransmissions of a packet until it is received by all receivers under node n. Then, the CDF for node n is $F_n(m)=Pr[M(n)<=m]$, i.e. $F_n(m)$=Prob (all nodes from n and below got the packet in, at most, m trials). We denote $F_r(m)$, $F_n(m)$, and $F_s(m)$ to be the CDF of the total number of transmissions for leaf receiver (r), nodes (n), and sender (s), respectively. One can then obtain the number of transmissions E[M] [12], [62] for the sender as follows.

$$E[M(S)] = \sum_{m=1}^{\infty} m P_S(M=m) = \sum_{m=1}^{\infty} (1 - F_s(m)) \tag{4.1}$$

The CDF of sender S can be obtained in terms of the CDF of its children nodes. For example, w_1 nodes are the children nodes of the sender in Fig. 4.2.

$$F_S(m) = \prod_{c \in child(S)} F_c(m) \tag{4.2}$$

The CDF $F_n(m)$ of an intermediate node n can also be computed in terms of the CDF $F_c(m)$ of children nodes. If m trials are conducted at node n, the probability that node n lost a specific packet exactly u times out of m trials is $\binom{m}{u} p_n^u (1-p_n)^{m-u}$ where p_n is the packet loss probability on the link leading to node n. Node n would forward the m-u correct copies of the same specific packet to its children node. The conditional probability that the packet would be received successfully by all receivers under node n is $\prod_{c \in child(n)} F_c(m-u)$. Thus, the exact expression for $F_n(m)$ may now be written as

$$F_n(m) = \sum_{u=0}^{m-1} \binom{m}{u} p_n^u (1-p_n)^{m-u} \prod_{c \in child(n)} F_c(m-u) \tag{4.3}$$

where p_n is the packet loss probability on the link leading to node n.

For each bottom leaf node that has no children node, one can obtain the CDF of a leaf receiver r from (4.3).

$$F_r(m) = \sum_{u=0}^{m-1} \binom{m}{u} p_r^u (1-p_r)^{m-u} = 1 - p_r^m, \tag{4.4}$$

where p_r is the packet loss probability on the link leading to leaf receiver r.

Using the above equations (4.1)-(4.4) that apply to general tree topologies, one can recursively calculate the expected number of transmissions and retransmissions. Needless to say, one should use recursion starting from the bottom of the nodes, and should use the numerical evaluation of equations (4.1) - (4.4) to evaluate bandwidth consumption E[M], even if they look like analytical expressions. One may follow such an approach for a

simple topology (e.g. topologies of type A and type B), and, even in this case, it will not be possible to investigate the effects of RR location on such E[M], since equations (4.1)--(4.4) are void of such RR locations. In this chapter, we rather derive a recursive equation that solves E[M] in the case that RRs are not used. However, for the cases that RRs are utilized we follow a completely analytical approach for the evaluation of E[M] without the need for recursion. For the sake of convenience, the following assumes a homogeneous packet loss probability, i.e. all links have the same packet loss probability p.

4.3.1 Linear Topology

For a linear chain in Fig. 4.4, all nodes are aligned in a row. A loss from any link will lead to retransmissions to all nodes. The equivalent loss probability of a linear chain for N links is p_{equ}=1-Prob (all nodes receive the correct packet)=$1-(1-p)^N$. Thus, the probability density function of transmission trials m follows the geometric distribution.

$$P_S(m) = p_{equ}^{m-1}(1 - p_{equ}) \tag{4.5}$$

Therefore, the expected number of transmissions, including retransmissions, is

$$E[M] = \sum_{m=o}^{\infty} mP_S(m) = \frac{1}{1 - p_{equ}} = \frac{1}{(1-p)^N}, \tag{4.6}$$

where N is the total number of links for linear topology, and p is the packet loss probability.

4.3.2 Star Topology

For a star topology in Fig. 4.5, we have only one level of children links. Finding the

distribution function of transmission trials m for the sender from (4.2) and (4.4) is easy:

$$F_S(m) = \prod_{c \in child(S)} F_c(m) = (1-p^m)^N = \sum_{k=0}^{N} \binom{N}{k} (-1)^k p^{mk}, \tag{4.7}$$

where N is the total number of links for a star topology and p is the packet loss probability of each link. Substituting the above equation into (4.1), one obtains the following result.

$$E[M] = \sum_{m=0}^{\infty} (1 - F_S(m)) = \sum_{k=1}^{N} \binom{N}{k} (-1)^{k+1} \frac{1}{1-p^k}. \tag{4.8}$$

4.3.3 Type A: k-ary Tree

For a k-ary tree with no RRs, one can successively reduce the tree to a single link, as in Fig. 4.6. For example, Fig. 4.6(b) is obtained from Fig. 4.6(a) by defining p_{h-1} as the equivalent packet loss probability of the subtree of depth $h-1$. Finally, the tree is equivalent to a star topology of packet loss probability q_h in Fig. 4.6(d), where q_h is the equivalent loss probability for the two linear links of p and p_{h-1}.

For the star topology in Fig. 4.6(c), one can easily find the expected number of transmissions $E[M]_h$ for depth h from (4.8).

$$E[M]_h = \sum_{k=1}^{w} \binom{w}{k} (-1)^{k+1} \frac{1}{1-q_h^k}, \tag{4.9}$$

where w is the number of children nodes, and q_h is equivalent to the packet loss probability of the child link for the tree of depth h. From Fig. 4.6(c) and Fig. 4.6(d), we obtain

$$E[M]_h = \frac{1}{1-p_h} = \sum_{k=1}^{w} \binom{w}{k} (-1)^{k+1} \frac{1}{1-q_h^k}, \tag{4.10}$$

where p_h is the equivalent packet loss probability for a k-ary tree of depth h. From Fig.

4.6(b) and Fig. 4.6(c), one link with p_{h-1} and one link with p constitute a simple linear topology; thus, one may easily obtain q_h in terms of p_{h-1}.

$$q_h = 1 - (1-p)(1-p_{h-1}), \tag{4.11}$$

where p_{h-1} is the equivalent packet loss probability for a k-ary tree of depth *h-1* (in rectangular of Fig. 4.6), and p is the packet loss probability of each link.

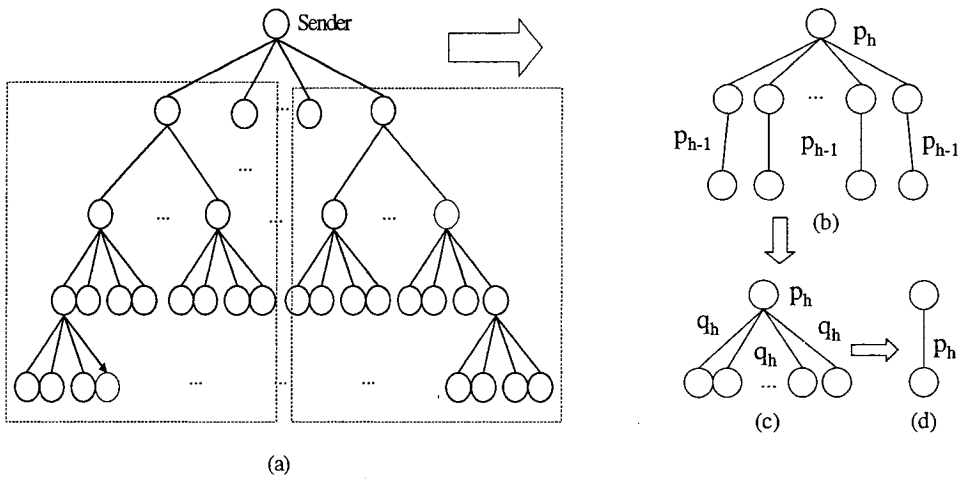


Fig. 4.6 Reduction techniques for type A

Substituting (4.11) into (4.10), we obtain the equivalent loss probability of depth h in terms of the loss probability of depth *h-1*.

$$\frac{1}{1-p_h} = \sum_{k=1}^{w} \binom{w}{k} (-1)^{k+1} \frac{1}{1-[1-(1-p)(1-p_{h-1})]^k} \tag{4.12}$$

In the same way, we have the following equation for a tree of depth *h-1*.

$$E[M]_{h-1} = \frac{1}{1-p_{h-1}} \tag{4.13}$$

Substituting (4.13) into (4.10), we obtain the expected number of transmissions for a tree of depth *h* in terms of the same at depth *h-1*, i.e., recursive equations in $E[M_h]$.

$$E[M]_h = \sum_{k=1}^{w} \binom{w}{k} (-1)^{k+1} \frac{1}{1-[1-\frac{1-p}{E[M]_{h-1}}]^k}, \tag{4.14}$$

where $E[M]_h$ is the expected number of transmissions for a k-ary tree of depth h. $E[M]_{h-1}$ is the expected number of transmissions for a k-ary tree of depth *h-1*, *p* is the packet loss probability, and w is the number of children nodes. A tree of depth *h=1* is a star topology in which each children link has the packet loss probability p; thus, one can easily obtain the following equation from (4.10):

$$E[M]_{h=1} = \sum_{k=1}^{w} \binom{w}{k} (-1)^{k+1} \frac{1}{1-p^k}. \tag{4.15}$$

Thus, we can recursively calculate the number of transmissions using (4.14) and the initial value of (4.15). However, this analysis does not consider where the RR location might be, or if there is RR at all.

4.3.4 Type B

For a topology of type B, we can reduce the tree to one single link in a manner similar to that of type A, shown in Fig. 4.7. A type B typology with a depth of h can be made equivalent to a star topology where only one link has a different loss probability q_h from other links. For this star topology, one may obtain the CDF of the number of trials for a topology of depth h.

$$\begin{aligned} F_S(m) &= \prod_{c \in child(S)} F_c(m) = (1-p^m)^{w_1-1}(1-q_h^m) \\ &= \sum_{u=0}^{w_1-1} \binom{w_1-1}{u} (-1)^u [p^{mu} - (p^u q_h)^m] \end{aligned} \tag{4.16}$$

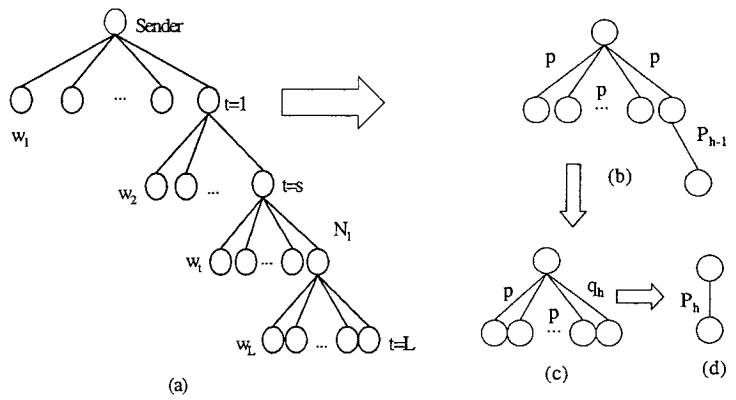


Fig. 4.7 Reduction techniques for type B

where p is the packet loss probability, q_h is the equivalent packet loss probability for the two links of p and p_{h-1}, and w_1 is the number of children links for level 1. Substituting the above equation into (4.1), one may predict the expected number of transmissions and retransmissions for depth h, i.e. $E[M]_h$.

$$E[M]_h = \sum_{m=o}^{\infty}(1 - F_S(m)) \tag{4.17}$$
$$= \frac{1}{1-q_h} + \sum_{u=1}^{w_1-1}\binom{w_1 - 1}{u}(-1)^{u+1}\left(\frac{1}{1-p^u} - \frac{1}{1-p^u q_h}\right)$$

where q_h is the equivalent loss probability of p and p_{h-1} shown in Fig. 4.7(b) , i.e.,

$$q_h = 1-(1-p)(1-p_{h-1}) = 1-\frac{1-p}{E[M]_{h-1}}\ , \tag{4.18}$$

where $E[M]_{h-1}$ is the total number of transmissions for depth h-1. Substituting (4.18) into (4.17), one can recursively obtain the total number of transmissions at depth h:

$$E[M]_h = \frac{E[M]_{h-1}}{1-p} + \sum_{u=1}^{w_1-1}\binom{w_1 - 1}{u}(-1)^{u+1}\left(\frac{1}{1-p^u} - \frac{1}{1-p^u(1-(1-p)/E[M]_{h-1})}\right) \tag{4.19}$$

where w_1 is the number of links in level 1 for a tree of depth h and recursion may be applied

using (4.19), starting with $E[M]_{h=1}$ of the leaves in (4.15).

4.4 Performance Evaluations of General Tree Topologies

One can use equations (4.1)-(4.4) to recursively calculate E[M]; however, the computation of E[M] may be very intensive for a general topology [40]. Even if the reduction technique is used, the computation of E[M] can be exponential with the number of nodes N [12], [13]. Based on intensive computations of E[M], it is difficult to calculate other parameters, such as bandwidth and delay, for a large multicast network, especially for a real time dynamic network. The efficient estimation of E[M] is necessary. In this section, we follow a completely analytical approach, evaluating the E[M] of a general topology without the need for recursion. Table 4.1 lists the notations used in this chapter.

4.4.1 Some Lemmas

Before deriving the number of transmissions, we first obtain some useful characteristics of general trees. Assume that d_k is the number of links from the sender to node k and n_k is the total number of links under node k. The root node is denoted as $k=0$. A tree topology with N links has the following characteristics.

Lemma 1: $$\sum_{k=1}^{N} n_k = \sum_{k=1}^{N} (d_k - 1) \tag{4.20}$$

Proof: Suppose that each node of the multicast tree has a counter whose value is 0 at the beginning. Each time one adds the value of d_k for node k to $\sum_{k=0}^{N} d_k$, the counters of these

nodes (except node k) that belong to the path from the sender to node k will be incremented by *1*. For example, $d_A=2$ in Fig. 4.8, is equivalent to the counters of nodes $k=0$ and $k=1$ incremented by 1. Finally, $\sum_{k=0}^{N} d_k$ is the summation of the counters of all nodes. The counter of a specific node (node j) is determined by the number of downstream nodes n_j. Its counter will be increased n_j times. Thus, $\sum_{k=0}^{N} d_k$ is the summation of n_j for all nodes including the sender (node 0), i.e.

$$\sum_{k=0}^{N} d_k = \sum_{k=0}^{N} n_k \quad \text{or} \quad \sum_{k=1}^{N} n_k = \sum_{k=1}^{N} (d_k - 1) \tag{4.21}$$

Lemma 2: $$\sum_{k_1=1}^{N} \sum_{k_2 \in d_{k_1}} n_{k_2} = \sum_{k_1=1}^{N} (n_{k_1} + 1) n_{k_1} \tag{4.22}$$

$$\sum_{k_1=1}^{N} \sum_{k_2 \in d_{k_1}} d_{k_2} = \sum_{k_1=1}^{N} (n_{k_1} + 1) d_{k_1} \tag{4.23}$$

Proof: Suppose that each node has a counter whose beginning value is 0. For each node from the sender to the specific node k_1, $\sum_{k_2 \in d_{k_1}} n_{k_2}$ will add to its counter a value equal to the number of downstream links. The counter of a specific node k will be increased n_k+1 times, determined by the number of nodes downstream including node k itself. The final counter value of node k is $n_k(n_k+1)$. Thus, $\sum_{k_1=1}^{N} \sum_{k_2 \in d_{k_1}} n_{k_2}$ is the summation of all node counters: i.e.

$\sum_{k_1=1}^{N} \sum_{k_2 \in d_{k_1}} n_{k_2} = \sum_{k_1=1}^{N} (n_{k_1} + 1) n_{k_1}$.

We can obtain (4.23) in a similar manner. Due to $\sum_{k_2 \in d_{k_1}} d_{k_2} = \frac{1}{2}(d_{k_1} + 1) d_{k_1}$, we have the

following equation:

$$\sum_{k_1=1}^{N} \sum_{k_2 \in d_{k_1}} d_{k_2} = \frac{1}{2} \sum_{k_1=1}^{N} (d_{k_1} + 1) d_{k_1} \tag{4.24}$$

From (4.23) and (4.24), we have the following equation:

$$\sum_{k_1=1}^{N} n_{k_1} d_{k_1} = \frac{1}{2} \sum_{k_1=1}^{N} (d_{k_1} - 1) d_{k_1} . \tag{4.25}$$

Lemma 3 : $$\sum_{k_1=1}^{N} \sum_{k_2 \in n_{k_1}} n_{k_2} = \sum_{k_1=1}^{N} (d_{k_1} - 1) n_{k_1} \tag{4.26}$$

$$\sum_{k_1=1}^{N} \sum_{k_2 \in n_{k_1}} d_{k_2} = \sum_{k_1=1}^{N} (d_{k_1} - 1) d_{k_1} \tag{4.27}$$

Proof: Suppose that each node has a counter whose starting value is 0. For each node under the specific node k_1, $\sum_{k_2 \in n_{k_1}} n_{k_2}$ will add to its counter a value equal to the number of links downstream. The counter of a specific node k will be increased d_k-1 times, as determined by the number of links not including node k itself. The final counter value of node k is $n_k(d_k$-1). Thus, $\sum_{k_1=1}^{N} \sum_{k_2 \in n_{k_1}} n_{k_2}$ is the summation of all node counters, i.e. $\sum_{k_1=1}^{N} \sum_{k_2 \in n_{k_1}} n_{k_2} = \sum_{k_1=1}^{N} (d_{k_1} - 1) n_{k_1}$. Proof of (4.27) is omitted here.

4.4.2 Probability of the Number of Transmissions

For a tree-based reliable multicast, a packet loss of one intermediate node is involved in the

retransmissions of many nodes. Once one intermediate node loses a packet, all nodes under the node will lose it. For example, if node k_1 loses a packet in Fig. 4.8, n_{k_1} nodes will not receive the packet where n_{k_1} is the number of links under the node k_1. The purpose of the next retransmission is to have the requested packet received at the n_{k_1} nodes. The requested packet must pass through the path from the sender to node k_1 to recover the losses. The number of nodes involved in retransmissions depends not only on the nodes that did not receive the packet, but also on the nodes along the path from the sender to node k_1. Therefore, we only concentrate on whether these nodes receive the packet correctly in the next retransmission.

If losses out of the n_{k_1} nodes recur in the next retransmission, we need to repeat the above process until all nodes receive the packet correctly. We do not concentrate on whether the nodes that have received the packet correctly do so again on further retransmissions. Therefore, each multicast retransmission is aimed at recovering losses from the previous transmission or retransmissions.

After understanding the process of loss recovery, we can evaluate the multicast performance. Packet losses may take place at different nodes in each transmission or retransmission. The loss recovery of different nodes needs different retransmission times; therefore, the probability density of the number of transmissions M depends on which node loses the packet.

Table 4.1 Notations for a Reliable Multicast

N is the total number of links.
p is the packet loss probability of each link.
M, m is the number of transmissions (including retransmissions) for one packet.
d_k is the number of links from the sender to node k. It is also the set of these links.
n_k is the total number of links under node k. It is also the set of these links.
$d_{k_1 k_2 \cdots k_i}$ is the total number of links from the sender to node $k_1,k_2,...,k_i$, $i=1,2,....$ It is also the set of these links.
$P(M=m)$ is the probability of the m transmission and retransmissions for one packet.
$P(M=m \mid k_1,k_2...)$ is the conditional probability of m retransmissions after node k_1, node k_2, ..., lose the packet in a previous transmission.

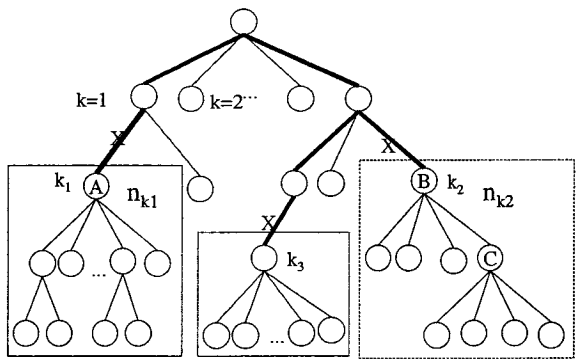


Fig. 4.8 One example of a general tree

Here, we first clarify two kinds of losses. If the loss of one link is not under the subtree of another link loss, two losses do not affect each other; for example, the losses of node k_1 (A)

and k_2 (B) are independent. However, if one link is under the subtree of another link loss, the losses are dependent; for example, the loss of node B clearly results in the loss of node C. In the following, losses refer to independent losses that do not affect each other, and dependent loss is considered through the factor n_k.

1) No loss occurs

The probability that no loss occurs is

$$\text{Prob.(no loss)}=(1-p)^N \tag{4.28}$$

2) Only one loss occurs

Assume that only one loss occurs; e.g. node k_1 loses the packet. The probability that only node k_1 loses data is

$$\text{Prob.(only node } k_1 \text{ loses)}= p(1-p)^{N-n_{k_1}-1}, \tag{4.29}$$

where n_{k1} is the number of links under node k_1. Readers are urged to read Table 4.1 for multicast notations.

3) j losses occur

The probability that node k_1, k_2,..., k_j lost the packet in one transmission is

$$\text{Prob.(node } k_1, k_2,..., k_j \text{ lose)} = \tag{4.30}$$

$$\begin{cases} p^j(1-p)^{N-\sum_{s=1}^{j} n_{k_s}-j}, & k_2 \notin D_{k_1},\cdots,k_j \notin D_{k_1k_2\cdots k_{j-1}} \\ 0, & otherwise \end{cases}$$

where n_{k_s} is the number of links under node k_s ($s=1,2,...,N$), and is also the set of these links. $d_{k_1k_2\cdots k_i}$ is the total number of links from the sender to node $k_1,k_2,...,k_i$, (and is also the set of these links). For example, $d_{k_1k_2}$ =4 (in bold), $d_{k_2k_3}$ =4, and $d_{k_1k_2k_3}$ =6 in Fig. 4.8. $D_{k_1k_2\cdots k_i} = d_{k_1k_2\cdots k_i} \cup n_{k_1} \cup n_{k_2} \cup \cdots \cup n_{k_i}$ (i=1,2,...,j) is set of these links, including $d_{k_1k_2\cdots k_i}$ and n_{k_s} (s=1,2…,i).

After we know the probability at which nodes lose the packet in one transmission, we can recursively calculate the probability density of the number of transmissions M. The probability of only one transmission is obvious:

$$P(M=1)=(1-p)^N \tag{4.31}$$

If some nodes lose the packet in the first transmission and the sender can recover these losses in the first retransmission, one can obtain the probability of two transmissions.

$$P(M=2)=\sum_{k_1=1}^{N} p(1-p)^{N-n_{k_1}-1} P(M=1|k_1)+\frac{1}{2!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}} p^2(1-p)^{N-n_{k_1}-n_{k_2}-2} P(M=1|k_1k_2)+\cdots \tag{4.32}$$

where $P(M=1|k_1k_2...k_s)$ is the conditional probability that one retransmission is needed to make multicast successful if nodes k_1, $k_2,...,k_s$ lose the packet in the first transmission.

Similarly, we can obtain the probability of m transmissions.

$$\begin{aligned} P(M=m) &= \sum_{k_1=1}^{N} p(1-p)^{N-n_{k_1}-1} P(M=m-1|k_1)+\cdots \\ &+\frac{1}{j!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}\cdots\sum_{k_j\notin D_{k_1\cdots k_{j-1}}} p^j (1-p)^{N-\sum_{i=1}^{j} n_{k_i}-j} P(M=m-1|k_1k_2\cdots k_j)+\cdots \end{aligned} \tag{4.33}$$

where $P(M=m-1|k_1k_2...k_j)$ is the conditional probability of $m-1$ retransmissions after node k_1, node k_2, ..., k_j lose the packet in previous transmission; for example, $P(M=m-1|k_1)$ is the conditional probability of $m-1$ retransmissions if node k_1 loses the packet.

4.4.3 The Analytical Approximation of E[M]

Each multicast subgroup meets the condition of $Np<1$. If $Np>1$, the sender needs to multicast the packet so many times that the protocol cannot work properly. Thus, we need to design loss recovery to reduce the number of retransmissions. The local recovery and partitioning of a group can be efficiently used to reduce the scope of loss recovery and to further reduce the number of retransmissions. Here, we consider the case of $Np<1$.

For a multicast subgroup of $Np<1$, we may expand the above probability density function of M according to the order of loss probability p. In the following, we will consider the function until the 3rd order approximation of p or Np. From (4.31), one may have

$$P(M=1)=(1-p)^N=1-Np+\binom{N}{2}p^2-\binom{N}{3}p^3+\cdots \tag{4.34}$$

In order to calculate P(M=2), one needs to retransmit only once after the losses take place. For example, if only node k_1 loses the packet in the first transmission, the probability that the first retransmission will be successful is $P(M=1|k_1)=(1-p)^{d_{k_1}+n_{k_1}}$, where d_{k_1} is the number of links from the sender to node k_1 and n_{k1} is the number of links under node k_1. Similarly, one can obtain

$$P(M=1|k_1k_2\cdots k_j)=(1-p)^{d_{k_1k_2\cdots k_j}+\sum_{s=1}^{j} n_{k_s}}, \qquad k_2\notin D_{k_1},\cdots,k_j\notin D_{k_1k_2\cdots k_{j-1}} \tag{4.35}$$

where $d_{k_1k_2\cdots k_j}$ is the number of links from the sender to node $k_1,k_2,...,k_j$. Substituting (4.35) into (4.32), one has

$$P(M=2)=\sum_{k_1=1}^{N}p(1-p)^{N+d_k-1}+\frac{1}{2!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}p^2(1-p)^{N+d_{k_1k_2}-2}$$
$$+\frac{1}{3!}\sum_{k_1=1k_2\notin D_{k_1}}^{N}\sum\sum_{k_3\notin D_{k_1k_2}}p^3(1-p)^{N+d_{k_1k_2k_3}-3}+\cdots \tag{4.36}$$

After expanding the equation above, one may obtain the following approximation:

$$P(M=2)=Np-[\binom{N}{2}+\sum_{k_1=1}^{N}(n_{k_1}+d_{k_1})]p^2+a_{23}p^3+\cdots, \tag{4.37}$$

where

$$a_{23}=\frac{1}{3!}\sum_{k_1=1k_2\notin D_{k_1}}^{N}\sum\sum_{k_3\notin D_{k_1k_2}}1+\sum_{k_1=1}^{N}\binom{N+d_k-1}{2}-\frac{1}{2!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}(N+d_{k_1k_2}-2) \tag{4.38}$$

Similarly, one may use (4.37) to evaluate $P(M=2|k_1,k_2,...,k_j)$. If one only considers the 3rd order approximation, one acheives the following result:

$$P(M=3)=\sum_{k_1=1}^{N}(d_{k_1}+n_{k_1})p^2+a_{33}p^3+\cdots, \tag{4.39}$$

where

$$a_{33}=-\sum_{k_1=1}^{N}[(N-\frac{3}{2}+\frac{d_{k_1}-n_{k_1}}{2})(d_{k_1}+n_{k_1})+\sum_{k_2\in D_{k_1}}(d'_{k_2}+n'_{k_2})]$$
$$+\frac{1}{2!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}(d_{k_1k_2}+n_{k_1}+n_{k_2}) \tag{4.40}$$

Using (4.39), one can obtain the 3rd approximation of $P(M=4)$.

$$P(M=4)=a_{43}p^3+\cdots, \tag{4.41}$$

where $a_{43} = \sum_{k_1=1}^{N} \sum_{k_2 \in D_{k_1}} (d'_{k_2} + n'_{k_2})$. (4.42)

So far, we derive until the 3rd order approximation for the probability density of the number of transmissions and retransmissions. One may find the expected number of transmissions easily using (4.34), (4.37), (4.39), and (4.41):

$$E[M] = \sum_{m=1}^{\infty} mP(M=m) \approx 1 + Np + [xN - \binom{N}{2}]p^2 + [\binom{N}{3} + y]p^3, \quad (4.43)$$

where $x = \frac{1}{N}\sum_{k=1}^{N}(n_k + d_k)$ (4.44)

$$y = -2\binom{N}{3} + 2a_{23} + 3a_{33} + 4a_{43}. \quad (4.45)$$

After some calculations, y may be simplified to the following formula (see Appendix):

$$y = \frac{5}{6}\sum_{k=1}^{N}[(d_k)^2 - (n_k)^2] + \frac{xN}{6}. \quad (4.46)$$

Assume that $d_k+n_k \sim x$. We can estimate the y value as

$$y \sim \frac{5x}{6}\sum_{k=1}^{N}(d_k - n_k) + \frac{xN}{6} = xN \quad (4.47)$$

We may ignore the effect of y on E[M] due to $yp^3 \sim xNp^3 << xNp^2$ for the small p value and to $y << \binom{N}{3}$ for the large N value.

4.4.4 Discussions

From (4.43), E[M] depends on the number of links N, the packet loss probability p, and the topology. Different topologies have different values of x and y. x and y reflect the effects of

the 2nd and 3rd approximation of the topology, respectively.

From lemma 1, we have $\sum_{k=1}^{N} n_k = \sum_{k=1}^{N} (d_k - 1)$; therefore, x is written as follows.

$$x = \frac{1}{N}\sum_{k=1}^{N}(n_k + d_k) = \frac{2}{N}\sum_{k=1}^{N} n_k + 1 = \frac{2}{N}\sum_{k=1}^{N} d_k - 1 \tag{4.48}$$

Defining g_t as the probability that a node is in level t, i.e. $g_t = G_t/N$ where G_t is the total number of links for level t, x can be written as

$$x = \frac{2}{N}\sum_{k=1}^{N} d_k - 1 = \frac{2}{N}\sum_{t=1}^{L} tG_t - 1 = 2\sum_{t=1}^{L} tg_t - 1 = 2E[t] - 1, \tag{4.49}$$

where $E[t]$ is the expected value of the node levels over the whole network for a general topology, and L is the depth of the tree.

The x value reaches the biggest value for the linear topology and the smallest value for the star topology if they have the same number of links. For a linear topology, $n_k + d_k = N$, so it is easy to get x and y value from (4.48) and (4.46): $x=N$ and $y=N^2$. For a star topology, $d_k=1$ and $n_k=0$, so $x=1$ and $y=N$ from (4.48) and (4.46). This approximation (4.43) for the star and linear topologies coincides perfectly with the exact solution until the 3rd order result.

For the topology of type A in Fig. 4.1, each intermediate node has the same number of children nodes, i.e. w children nodes. Thus, we obtain the x value of a k-ary tree from (4.48).

$$x = \frac{2}{N}\sum_{k=1}^{N} n_k + 1 = 2[L - \frac{w+1}{2(w-1)}] + \frac{2Lw}{N(w-1)}, \tag{4.50}$$

where L is the depth of a k-ary tree and w is the number of children nodes for each intermediate node.

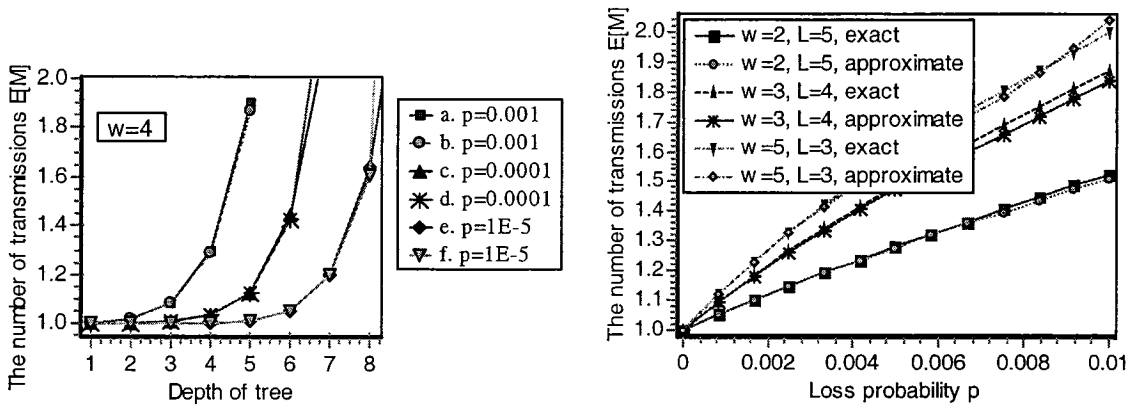


Fig. 4.9 A comparison of E[M] for type A. a,c,e: exact solution, b,d,f: approximate solution.

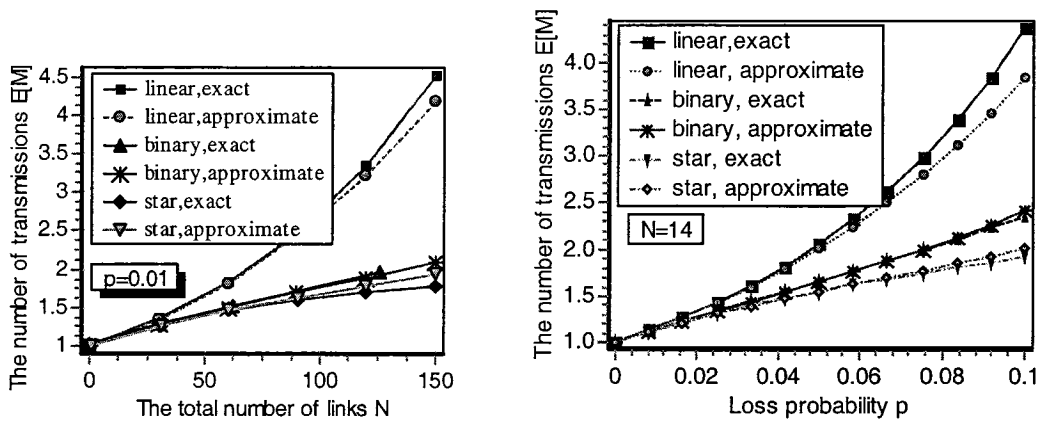


Fig. 4.10 A comparison of E[M] for different topologies.

For the topology of type B in Fig. 4.2, only one node in each level has children links; thus, we obtain the x value from (4.48):

$$x = \frac{2}{N}\sum_{k=1}^{N} d_k - 1 = \frac{2}{N}\sum_{t=1}^{L} t w_t - 1, \tag{4.51}$$

where w_t is the total number of links for level t, N is the total number of links. For a special

case of $w_1=w_2=...=w_L=w$, we have $N=Lw$ and $x=L$ from (4.51). The x value of a general tree (e.g., type C) can be obtained by (4.48)

By now, we have used different approaches to obtaining the expected number of transmissions. The following compares two such approaches. Fig. 4.9, Fig. 4.10, Fig. 4.11, and Fig. 4.12 each show a comparison of several topologies for the exact (i.e. the earlier recursion solution (4.6), (4.8), (4.14), and (4.19)) and the approximate solutions (4.43) for E[M]. For the small E[M] value, (4.43) gives a simple approximation to the exact solution.

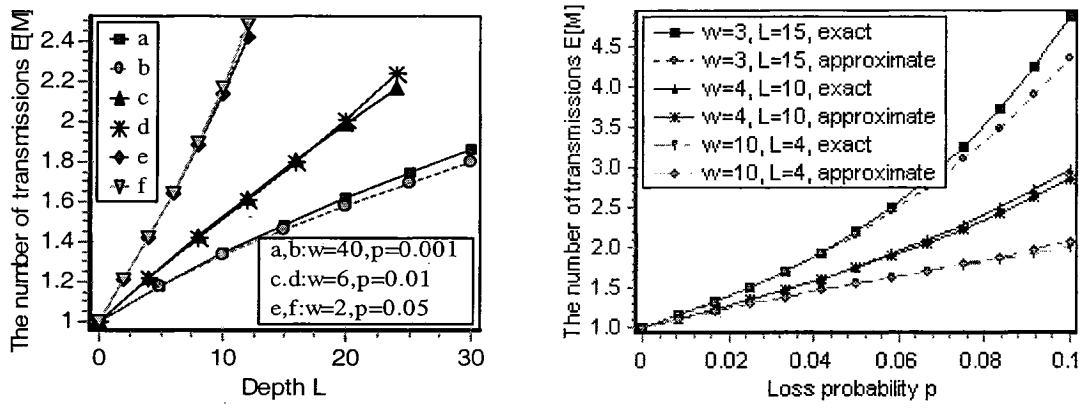


Fig. 4.11 A comparison of E[M] for type B. $w_1=w_2=w_3=...=w_L=w$. L is the depth of tree. a,c,e: exact solutions; b,d,f: approximate solution

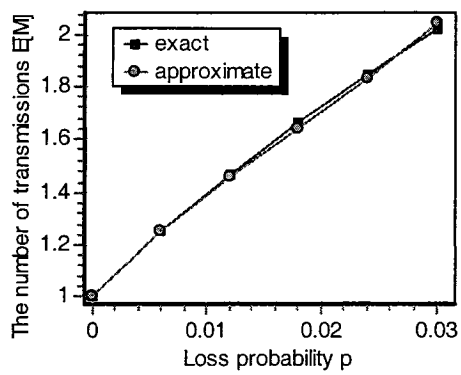


Fig. 4.12 A comparison of E[M] for type C.

4.5 Optimal RR Placements

The E[M] of earlier equations gives the expected number of transmission trials and it roughly reflects the bandwidth consumption; however, it applies only to one subgroup. For a hierarchical reliable multicast using one or more RRs, one multicast group is partitioned to several subgroups whose retransmission is handled by one RR. Placing r RRs that can send repairs to requesting nodes, one has r+1 subgroups. For each retransmission in one subgroup, the packet is multicast to the whole node population of the subgroup by the parent RR. Most nodes will be affected by retransmissions. As a worst case estimate of bandwidth consumption, one can assume that all links in a subgroup will be affected by each retransmission. We define C_i as the total bandwidth consumed by one source multicast packet over all links in subgroup i whose retransmissions are handled by one RR. Thus, one may obtain the expected bandwidth of each subgroup.

$$E[C_i] \approx E[M_i] N_i \ , \tag{4.52}$$

where $E[C_i]$ and $E[M_i]$ are the expected bandwidth consumption and the expected number of transmissions and retransmissions for subgroup i, respectively, and N_i is the number of links for subgroup i.

The bandwidth consumption of the whole multicast group is the summation of the bandwidth consumed by all subgroups. Therefore, one may find the total expected bandwidth consumption E[C] of r RRs.

$$E[C] = \sum_{i=0}^{r} E[C_i] = \sum_{i=0}^{r} N_i E[M_i] \tag{4.53}$$

From the approximation (4.43), E[M] should be a function of N for the given p. E[M] has different functions for different topologies or subgroups. We can assume that $f_i(N_i)=E[M_i]$ for subgroup S_i, $i=0,1,...,r$. If B is defined as the bandwidth consumed per link by a multicast packet averaged over all links in a multicast group, one may obtain the following equation:

$$E[B]=\frac{E[C]}{N}=\frac{1}{N}\sum_{i=0}^{r}N_i f_i(N_i) \quad . \tag{4.54}$$

In this thesis, we will minimize the bandwidth consumption (4.54) to obtain the optimal RR locations.

4.5.1 1 RR

The bandwidth consumption of one RR is

$$E[B]_i=\frac{N_0}{N}f_0(N_0)+\frac{N_1}{N}f_1(N_1) \quad . \tag{4.55}$$

The total number of links remains constant; i.e.

$$N_0+N_1=N \quad . \tag{4.56}$$

In order to find the optimal RR placements, we need to differentiate (4.55) and (4.56) with respect to N_0 (or to N_1, since they are related by (4.56)).

$$\frac{\partial[N_0 f_0(N_0)]}{\partial N_0}+\frac{\partial[N_1 f_1(N_1)]}{\partial N_1}\frac{\partial N_1}{\partial N_0}=0 \tag{4.57}$$

$$\frac{\partial N_1}{\partial N_0}+1=0 \tag{4.58}$$

It is clear that we can obtain the following solution to the above equations, i.e. the optimal

condition of one RR:

$$\frac{\partial[N_0 f_0(N_0)]}{\partial N_0} = \frac{\partial[N_1 f_1(N_1)]}{\partial N_1} \tag{4.59}$$

$$\text{or} \quad f_0(N_0) + N_0 \frac{\partial f_0(N_0)}{\partial N_0} = f_1(N_1) + N_1 \frac{\partial f_1(N_1)}{\partial N_1} \tag{4.60}$$

We can obtain the function $f_i(N_i)$ of each subgroup from (4.43); i.e.

$$f_i(N_i) = x_i N_i p^2 + g(N_i), \tag{4.61}$$

$$\text{where} \quad g(N_i) = 1 + N_i p - \binom{N_i}{2} p^2 + \binom{N_i}{3} p^3. \tag{4.62}$$

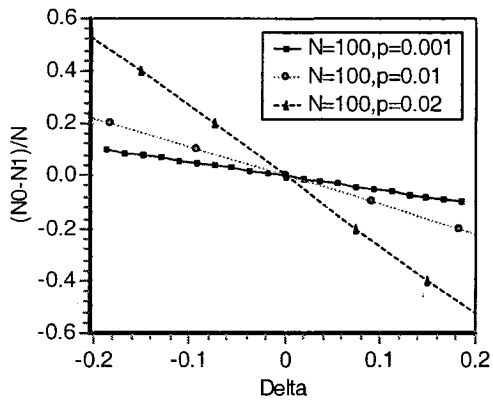


Fig. 4.13 The effects of topology on subgroup size.

Substituting (4.61) into (4.59), one gets

$$\Delta = \frac{\partial[N_1 g(N_1)]}{\partial N_1} - \frac{\partial[N_0 g(N_0)]}{\partial N_0}, \tag{4.63}$$

$$\text{where} \quad \Delta = [\frac{\partial(N_0^2 x_0)}{\partial N_0} - \frac{\partial(N_1^2 x_1)}{\partial N_1}] p^2. \tag{4.64}$$

Δ reflects the effect of the topology on the optimal partitioning of multicast trees, shown in Fig. 4.13. The subgroup with the larger topology parameter should have smaller size. In the case of a larger value of Np, the topology effect is significant and the subgroup size is

different.

4.5.1.1 Homogeneous Topology Structure of Different Subgroups

If each subgroup has the same topology, then $E[M_0]$ and $E[M_1]$ should have the same functional dependence on the number of links. Many topologies fit into this case: in the linear topology, each subgroup always has the same topology, no matter where RRs are placed; in the type A topology, all subgroups have a similar topology; in type B, with $w_1=w_2=...=w_L$, the subgroups also have the same topologies when RRs are placed in the intermediate nodes. Thus, they have the same functional dependence with N, i.e. $f_0(N)=f_1(N)=f(N)$. From analytical approximation (4.43) in the case that y is neglected, if the x of each subgroup has the same functional dependence with N, each x has the same f(N) function. From (4.43) and (4.49), the subgroups have the same function x and the same functional dependence on N, as long as each level has the same number of links for two subgroups. In the case of the same functional dependence with N, i.e. $f_0(N_0)=f_1(N_1)=f(N)$ or $x_0(N_0)=x_1(N_1)=x(N)$, the optimal conditions of RRs are obvious from (4.60):

$$N_0=N_1 . \tag{4.65}$$

When two subgroups have the same number of links, the whole network consumes the least bandwidth and has the best performance. Therefore, RR should be placed optimally to partition the group into subgroups that have the same number of links. This applies to homogeneous topology structures.

4.5.1.2 Different Topology Structures for Different Subgroups

If each subgroup has a different topology, then each of $E[M_0]$ and $E[M_1]$ will have a different functional dependence on the number of links; for example, for type B, if $w_1=w>>1$, $w_2=...=w_L=1$, one subgroup resembles a linear topology and another resembles a star topology when one RR is placed in t^{th} level. For the same number of links, the linear topology has a much larger E[M] than the star topology, as per Fig. 4.10. The two subgroups have very different functional dependences: $f_0(N_0) \neq f_1(N_1)$. Thus, $N_0=N_1$ will not be the optimal conditions for this case. In order to meet optimal condition (4.69), the linear-like subgroup should have a small size while the star-like subgroup should have a larger size. In other words, subgroups with larger x values should be of a smaller size.

4.5.2 r RRs

One may use Lagrange multipliers to obtain the optimal condition. We need to minimize the function (4.54), subject to the following constraint

$$\sum_{i=0}^{r} N_i - N = 0 \quad . \tag{4.66}$$

Introducing Lagrange multiplier λ, we form the new objective function $\xi(N_0,N_1,..., N_r)$ to be minimized

$$\xi(N_0, N_1, \cdots, N_r) = \frac{1}{N}\sum_{i=0}^{r} N_i f_i(N_i) + \lambda(\sum N_i - N) \tag{4.67}$$

We find its partial derivatives with respect to $N_0, N_1,..., N_r$ and set them equal to zero

$$\frac{\partial}{\partial N_i}\xi(N_0, N_1, \cdots, N_r) = \frac{1}{N}[f_i(N_i) + N_i \frac{\partial f_i(N_i)}{\partial N_i}] + \lambda = 0, \qquad i = 0,1,\cdots,r. \tag{4.68}$$

Thus, we have the following optimal condition of r RRs:

$$f_0(N_0) + N_0 \frac{\partial f_0(N_0)}{\partial N_0} = f_1(N_1) + N_1 \frac{\partial f_1(N_1)}{\partial N_1} = \cdots = f_r(N_r) + N_r \frac{\partial f_r(N_r)}{\partial N_r} \tag{4.69}$$

where $f_i(N_i)$ is given by (4.43) for subgroup i.

In the same way, one can obtain results similar to those with one RR. If each subgroup has the same topology, i.e. $f_0(N_0)=f_1(N_1)=...=f_r\ (N_r)$ or $x_0(N_0)=x_1(N_1)=...=x_r(N_r)$, each subgroup should be of the same size in order to obtain the best performance.

4.6 Results and Discussions

Fig. 4.14--Fig. 4.17 give the optimal placements of RRs for type A topology. These curves show that the best RR locations are in level 1 for Fig. 4.14, Fig. 4.16, and Fig. 4.17. In Fig. 4.15, the optimal locations show that two RRs are in level 1, and another is in level 2, placements of which result from having the same number of links for each subgroup. Thus, RRs should be placed to give each subgroup the same number of nodes. These results are consistent with the analytical result (4.65). From these curves, one may easily see that bandwidth consumption increases with the increasing depth L and packet loss probability p. This is because the number of retransmissions will become larger and larger as N and p grow.

From the above analysis, the placements of RRs depend on the difference of topologies among subgroups. For the homogeneous topology structure, they have similar topologies

after partitioning according to RR locations. Thus, optimal placements depend on only the number of links of each subgroup. Each subgroup should have the same size. Type A illustrates this case, for when RRs are placed in a network of type A, each subgroup has a similar topology. Thus, RRs are always placed to have the same topology and the same number of links for each subgroup. Table 4.2 displays these results. For example, levels 1, 1, and 2 are the optimal placements for 3 RRs in the case of a 3-ary tree (shown in Fig. 4.15 where the x axis denotes the different locations of the RRs in Table 4.3), because they have the same topology and the same size for 4 subgroups, and thus yield the best RR combination. For pure linear topology, we have similar results. No matter how we partition the linear topology, each subgroup always has linear topology. Thus, each subgroup should have the same number of links to get the best multicast performance. This result coincides with the results of the linear topology in [12]. The optimal placements of RRs are trade-offs between the sender and each RR. When each subgroup has the same number and the same topology, the whole group has the best performance.

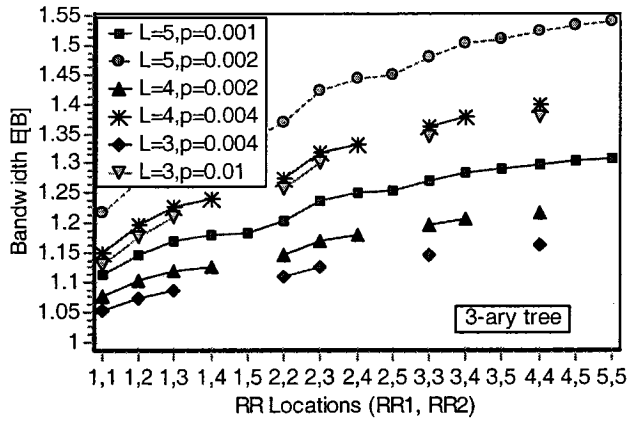


Fig. 4.14 Bandwidth consumption E[B] for type A using 2 RRs.

When each subgroup has a different topology, having the same size for each subgroup does not apply. Fig. 4.18 gives the optimal placement of RRs for a topology of type B where $w_1=w$, $w_2=\ldots=w_L=1$ and the location of RR is the level of RR. For example, the optimal placement for $w=150$ in Fig. 4.18 (square curve) is in level 20, which means that $N_0=w+20=170$ and $N_1=100-20=80$. This shows a different number of links for each subgroup to have the optimal placements of RRs. When the subgroups are the same size, $N_0=100$ and $N_1=100$ for $w=100$ in Fig. 4.18 (circle curve), RR is in level 1 and the multicast performance of type B is not optimal. The same number of links for each subgroup does not result in the best multicast performance for very different topologies. Subgroup S_1 is of a linear topology and subgroup S_0 has a star-like topology. The optimal results indicate (found by numerical enumeration of Fig. 4.18 over all RR locations) that S_1 should have a lower number of links and S_0 should have a higher number of links. It may as well be that the linear topology has a larger number of retransmissions than the star topology for the same number of links; therefore, a linear topology should have a lower number of links than a star topology, so that the whole population reaches a trade-off between the two subgroups.

The optimal placement of RRs in certain multicast networks is achieved through an exhaustive search of all possibilities. Yet one can see that allocating the RR possibility so as to divide the total network into equal partitions is not necessarily optimal. In fact, this allocation is only the case for homogeneous topologies, which do not often exist.

Table 4.2 Optimal Levels Of RRs for Type A

#RRs \ w	2	3	4	5
2 RRs	1,2	1,1	1,1	1,1
3 RRs	2,2,2	1,1,2	1,1,1	1,1,1

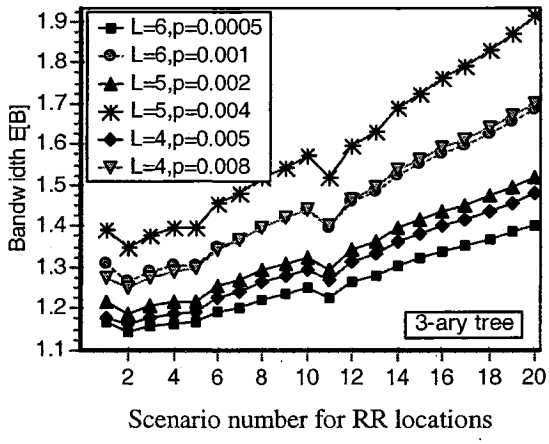


Fig. 4.15 Bandwidth E[B] of 3 RRs for 3-ary trees. RR locations corresponding to the x axis are shown in Table 4.3.

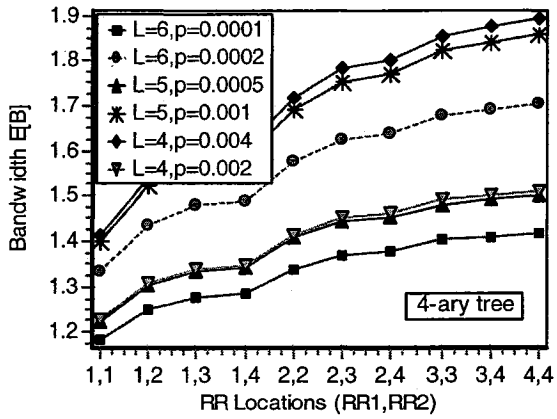


Fig. 4.16 Bandwidth consumption E[B] for type A using 2 RRs.

Table 4.3 Levels of 3 RRs corresponding to each scenario number on the x axis of Fig. 4.15 and Fig. 4.17

Scenario number of x axis	1	2	3	4	5	6
RRs levels	1,1,1	1,1,2	1,1,3	1,1,4	1,2,2	1,2,3
7	8	9	10	11	12	13
1,2,4	1,3,3	1,3,4	1,4,4	2,2,2	2,2,3	2,2,4
14	15	16	17	18	19	20
2,3,3	2,3,4	2,4,4	3,3,3	3,3,4	3,4,4	4,4,4

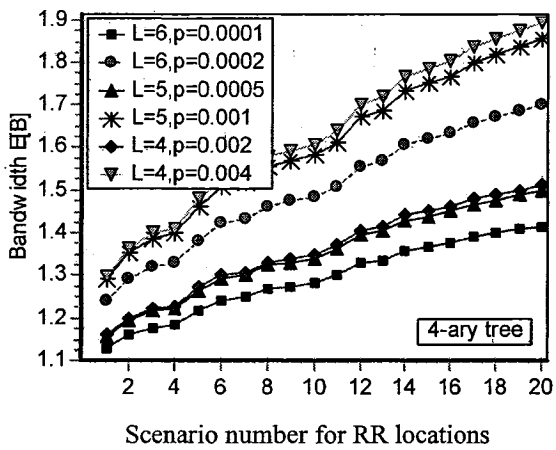


Fig. 4.17 The bandwidth E[B] of 3 RRs for 4-ary trees. The RR locations corresponding to the x axis are shown in Table 4.3.

We also find that the optimal placements of RRs shift to a higher level when w decreases (see Fig. 4.18), because the topology becomes increasingly homogeneous as *w* decreases. Thus, the optimal placement is to have subgroups of equal size. If the topology changes

significantly, the optimal placements of the RRs will change.

The optimal partitioning of a multicast group can effectively isolate the interactions among these subgroups and reduce the number of transmissions for each subgroup. Thus, one greatly reduces the bandwidth consumption that depends upon the number of transmissions. From the figures above, we find that the bandwidth saving is at least 30%.

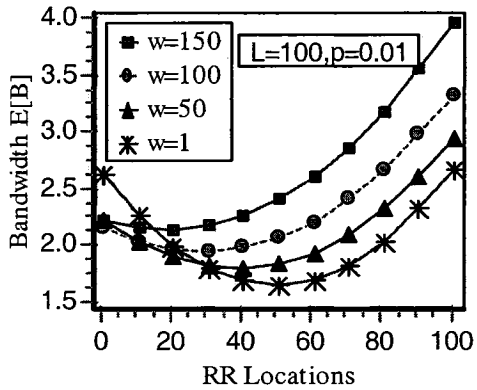


Fig. 4.18 Bandwidth E[B] of one RR for type B. $w_1=w$, $w_2=...=w_L=1$. Note that $w_1=1$ is a homogeneous topology structure

4.7 Conclusions

In this chapter, we have derived an analytical approximation for the number of transmissions that depends on the total number of links, the loss probability, and the topology. x approximately reflects the effects of the topology on multicast performance, and reaches its largest value for the linear topology and its smallest value for the star topology if the subgroups have the same number of links. We can use the number of transmissions to easily evaluate the bandwidth consumption.

Based on an estimate of bandwidth consumption, this chapter has investigated the effect of topology on optimal RR placements for multicast. For homogeneous topology, the multicast group should be partitioned into subgroups of equal sizes in order for the whole group to have the best performance; however, the opposite is true for topologies in which each subgroup has a very different topological structure.

Chapter 5 DYNAMIC PERFORMANCE OF RELIABLE MULTICAST

5.1 Introduction

Many references, namely [37]-[41], [69]-[81], discuss the performance analysis of tree-based reliable multicast on static networks. However, networks often change; receivers may prune from the network at any time, and some new receivers may join the network. Intermediate routers may also prune from or join the multicast session, which would result in a noticeable change in the network topology and performance. Thus, the total number of links and the topology in real multicast communications often change [82], [86]. Furthermore, the packet loss probability is often dynamic, which has been discussed for a given topology in [20]. In this chapter, we will focus only on the dynamics of topology and on the optimal placement of RRs. Because the bandwidth consumption depends on the topology, the optimal locations of repair routers will change accordingly.

In this chapter, we will consider a model for dynamic networks. Due to the random joining and pruning of nodes in an actual network, the topology of multicast groups changes over time. We will investigate multicast performance analysis based on this dynamic topology. We will also analyze the effects of dynamic topology on optimal partitioning and find the optimal placement of repair servers.

5.2 Analysis of Dynamic Multicast Networks

Any node of a dynamic network may prune from the network, and new nodes may join the network. Joining is similar to pruning, but in a reversed manner. For the convenience of analysis, the following study considers only the pruning of nodes, not their joining.

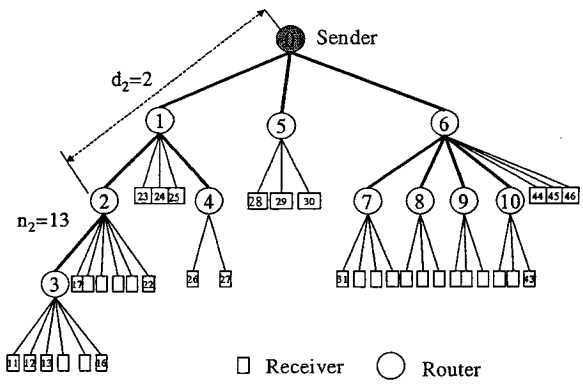


Fig. 5.1 An example of multicast trees

The nodes of a multicast group may be receivers or intermediate routers, either of which may perform pruning. The pruning of receivers may take place randomly; however, the pruning of intermediate routers is due to the burst pruning of subnets, in which all links under the intermediate router get pruned from the group. We will therefore discuss two pruning approaches in the following: the random pruning of receivers, and the burst pruning of subnets. In most cases, the pruning of receivers takes place more frequently in dynamic networks. This section first assumes a stationary process of dynamic topologies. The next section then discusses how to adapt the RR locations to the dynamic change of networks in time. Here we do not consider the effect of AWGN channel and interferences.

5.2.1 The Pruning of Receivers

For a multicast group having the same packet loss probability p for all links, contributions to E[M] are classified into two parts, i.e., one is due to the effect of the total number of links N, another is due to the effect of topology. For the analysis convenience of dynamic networks, we introduce parameter u as $u=xN$, then E[M] can be rewritten as follows.

$$E[M] = g(N) + up^2 \tag{5.1}$$

where $g(N)$ is a function only dependent on N and p and is independent of topology, i.e.,

$$g(N) = 1 + Np - \binom{N}{2}p^2 + \binom{N}{3}p^3 \tag{5.2}$$

u depends on the topology. From (4.49), u only depends on the distribution of the number of links in each level, i.e., u can be rewritten as follows.

$$u = 2\sum_{k=1}^{N} d_k - N = 2\sum_{t=1}^{L} tG_t - N \tag{5.3}$$

where G_t is the total number of links in level t, e.g. $G_1=3$ and $G_2=15$ in Fig. 5.1. u depends on how links are distributed to each level. The parameter u roughly reflects the effects of topology, e.g. the value u of a linear topology and a star topology is N^2 and N, respectively.

Suppose that each receiver can be pruned from the network with the probability λ. The individual pruning of each leaf is independent of that of any other. Assume that no intermediate routers or subnets are pruned. We need only know the distribution of the number of receivers in each level. We must distinguish receiver links and intermediate

router links for each level. Let w_t be the number of receivers in level t and w_t^T be the number of intermediate routers (bold links in Fig. 5.1) in level t. Assume that G_t is the total number of links in level t. Then $G_t = w_t + w_t^T$, for example, in Fig. 5.1, w_2=9, w_2^T=6, G_t=15. For a multicast tree with L levels, the total number w of receivers is defined as follows:

$$w = w_1 + w_2 + \cdots + w_L \tag{5.4}$$

For example, in Fig. 5.1, w_1=0, w_2=9, w_3=21, and w_4=6. For level t, each receiver may be pruned randomly from the network; therefore, the probability that j_t receivers are pruned from w_t receivers in level t is binomially distributed.

$$P(j_t) = \binom{w_t}{j_t} \lambda^{j_t} (1-\lambda)^{w_t - j_t} \tag{5.5}$$

where w_t is the total number of all receivers in level t, and λ is the probability of one receiver pruning.

The joint probability that $j_1, j_2, \ldots, j_L$ receivers prune from level $1, 2, \ldots, L$ respectively is given by:

$$P(j_1, j_2, \cdots, j_L) = P(j_1)P(j_2)\cdots P(j_L) = \binom{w_1}{j_1}\binom{w_2}{j_2}\cdots\binom{w_L}{j_L} \lambda^j (1-\lambda)^{w-j} \tag{5.6}$$

where w is defined by (5.4) while j is the total number of pruned receivers from the initial network, i.e.

$$j = j_1 + j_2 + \cdots + j_L \tag{5.7}$$

Table 5.1 Notations for Dynamic Multicast Networks

a.	N is the total number of links.
b.	p is the packet loss probability of each link.
c.	M, m is the number of transmissions (including retransmissions) for one packet.
d.	d_k is the number of links from the sender to node k. It is also the set of these links.
e.	n_k is the total number of links under node k. It is also the set of these links.
f.	$d_{k_1k_2\cdots k_i}$ is the total number of links from the sender to node $k_1,k_2,\ldots,k_i$, $i=1,2,\ldots$. It is also the set of these links.
g.	$P(M=m)$ is the probability of the m transmission and retransmissions for one packet.
h.	$P(M=m\|k_1k_2\ldots)$ is the conditional probability of m retransmissions after node k_1, node k_2, ..., lose the packet.
i.	G_t is the total number of links in level t.
j.	λ is the pruning probability of receivers.

After the j receivers are pruned from the initial network (at the start of the multicast session), the total number of links decreases to $N=N_0-j$, where N_0 is the number of links for the initial network. Due to the pruning of j_t receivers, the number of links in level t decreases to $w_t + w_t^T - j_t$, where w_t is the number of receiver links in level t and w_t^T is the number of intermediate routers in level t. Thus, the topology of the network changes accordingly, and u in (5.3) changes to:

$$u = 2\sum_{t=1}^{L} t(w_t + w_t^T - j_t) - (N_0 - j) = u_0 - (2\sum_{t=1}^{L} tj_t - j) \tag{5.8}$$

where u_0 is the topology parameters for the initial network, i.e. $u_0 = 2\sum_{t=1}^{L} t(w_t + w_t^T) - N_0$.

One can then calculate the average effects of the receiver pruning (though not of the router pruning, see Fig. 5.1) on the network topology parameter using (5.8), i.e.

$$E_1[u] = \sum_{j_1}^{w_1}\sum_{j_2}^{w_2}\cdots\sum_{j_L}^{w_L} P(j_1, j_2, \cdots, j_L)u = u_0 - (2\sum_{t=1}^{L} t\lambda w_t - \lambda w) = u_0 - \lambda u_r \tag{5.9}$$

where we denote by E_1 the expectation that is the average value of u over all possible topologies due to the receiver pruning, and u_r is defined as follows:

$$u_r = 2\sum_{t=1}^{L} tw_t - w \tag{5.10}$$

For a given network topology with the total number of links N and the topology parameter u, the number of transmissions has been obtained before ((5.1)), i.e.

$$E_2[M|N,u] = g(N) + up^2 \tag{5.11}$$

where E_2 is the average number of transmissions due to packets losses in a specific topology.

Thus, the average number of transmissions over all possible topologies is obtained by considering the pruning of receivers, i.e.

$$E_1[E_2[M|N,u]] = E_1[g(N)] + E_1[u]p^2 \tag{5.12}$$

where the E_1 is the average value of the transmission times over all possible network

topologies due to the pruning of receivers because different topologies have different number of transmissions, the E_2 is the average number of transmissions of multicast packets due to packet losses in a specific topology.

When j receivers are pruned from the initial network, $N=N_0-j$, and $g(N)$ becomes:

$$g(N) = g(N_0 - j) = g(N_0) - jp[1-(N_0 - \frac{1}{2})p + \frac{1}{6}(3N_0{}^2 - 6N_0 + 2)p^2] + \frac{1}{2}j^2[p^2 + (N_0 - 1)p^3] - \frac{1}{6}j^3 \quad (5.13)$$

For small λ, one can obtain the following approximation by substituting (5.9) and (5.13) into (5.12).

$$E_1[E_2[M \mid N,u]] = E_2[M_0] - \lambda[wp + (u_r - wN_0)p^2 + \frac{1}{2}wN_0(N_0 - 2)p^3] \quad (5.14)$$

where $E_2[M_0] = g(N_0) + u_0p^2$ is the number of transmissions for the initial multicast network in a case where no pruning has occurred.

In the same way, one can calculate the total bandwidth consumption C, which is the total number of links affected by one source multicast packet. Its expected value can be estimated if the retransmissions of repair packets are multicast.

$$E_1[C] = E_1[NE_2[M \mid N,u]] = E_1[Ng(N)] + E_1[uN]p^2 \quad (5.15)$$

Assuming that λ is small, $j<<N$, it is easy to obtain uN, $ug(N)$, $E[uN]$, and $E[Ng(N)]$in the following. From (5.8) and $N=N_0-j$, uN can be written by ignoring the term of j^2.

$$uN = u_0N_0 - [2N_0\sum_{t=1}^{L} tj_t - (N_0 - u_0)j] \quad (5.16)$$

Its expected value becomes

$$
\begin{aligned}
E_1[uN] &= \sum_{j_1}^{w_1}\sum_{j_2}^{w_2}\cdots\sum_{j_L}^{w_L} P(j_1, j_2, \cdots, j_L)uN \\
&= u_0N_0 - [2N_0\sum_{t=1}^{L} t\lambda w_t - (N_0 - u_0)\lambda w] \\
&= u_0N_0 - \lambda u_r N_0 - u_0\lambda w
\end{aligned}
\tag{5.17}
$$

where $P(j_1, j_2, \cdots, j_L)$ is the joint probability that j_1, j_2,..., j_L links are pruned from level *1*, *2*, ... , *L*.

From (5.2) and $N=N_0-j$, $ug(N)$ can be written by ignoring the terms of j^2, j^3, and j^4.

$$
\begin{aligned}
Ng(N) &= (N_0 - j)g(N_0 - j) \\
&\approx N_0 g(N_0) - j\{1 + 2N_0 p - (\frac{3}{2}N_0^2 - N_0)p^2 + \frac{1}{6}(4N_0{}^3 - 9N_0^2 + 4N_0)p^3\}
\end{aligned}
\tag{5.18}
$$

Taking the expected value of the equation above, one obtains the following equation.

$$
\begin{aligned}
E_1[Ng(N)] &= \sum_{j_1}^{w_1}\sum_{j_2}^{w_2}\cdots\sum_{j_L}^{w_L} P(j_1, j_2, \cdots, j_L)Ng(N) \\
&\approx N_0 g(N_0) - w\lambda[1 + 2N_0 p - (\frac{3}{2}N_0^2 - N_0)p^2 + \frac{1}{6}(4N_0{}^3 - 9N_0^2 + 4N_0)p^3]
\end{aligned}
\tag{5.19}
$$

Substituting (5.17) and (5.19) into (5.15), one can obtain an estimate of the bandwidth consumption.

$$
\begin{aligned}
E_1[C] &= E_1[Ng(N)] + E_1[uN]p \\
&= N_0 E_2[M_0] - w\lambda[1 + 2N_0 p - (\frac{3}{2}N_0^2 - N_0)p^2 + \frac{1}{6}(4N_0{}^3 - 9N_0^2 + 4N_0)p^3] \\
&\quad - (u_r N_0 + u_0 w)\lambda p^2
\end{aligned}
\tag{5.20}
$$

where $E_2[M_0] = g(N_0) + u_0 p^2$ is the number of transmissions for the initial multicast network in the case of no pruning.

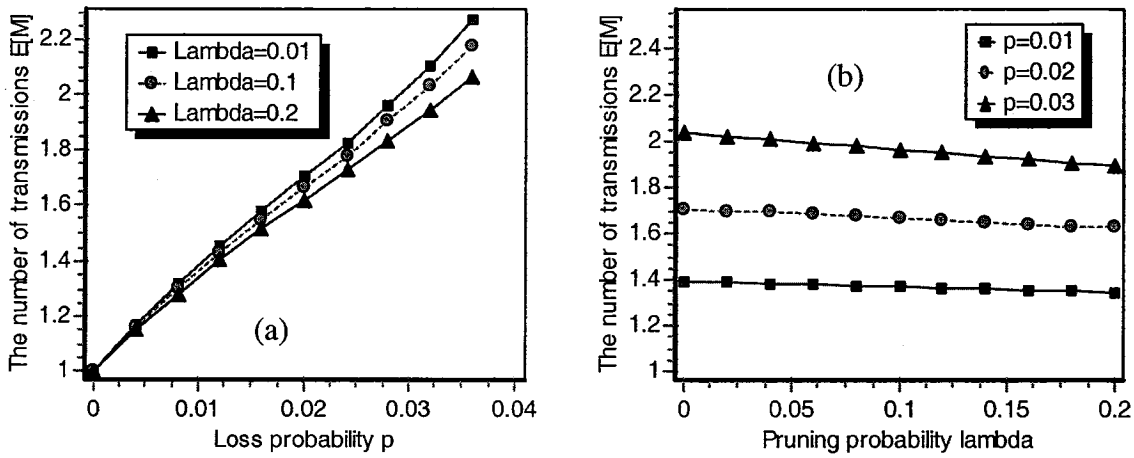


Fig. 5.2 The average number of transmissions for receiver pruning in the multicast tree of Fig. 5.1.

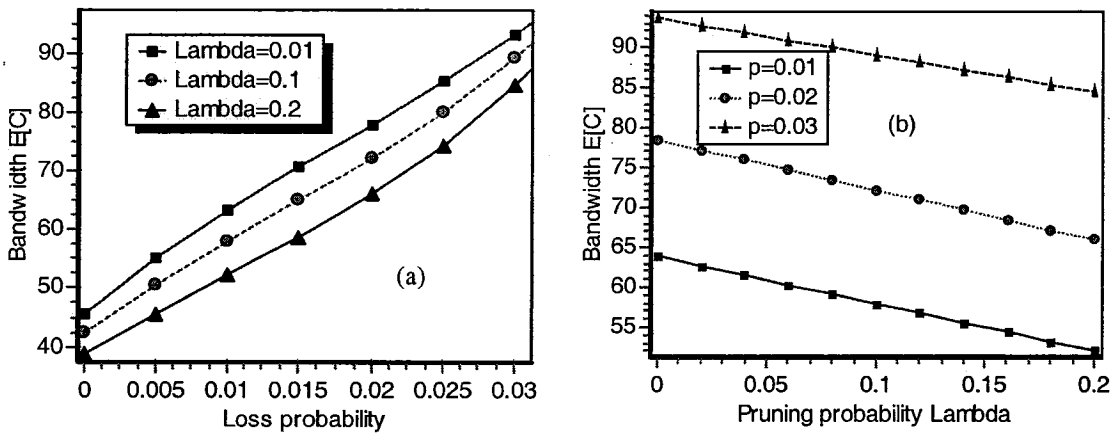


Fig. 5.3 The average bandwidth consumption for receiver pruning in the multicast tree of Fig. 5.1.

In Fig. 5.2, we give the results of the average number of transmissions (5.14) over all topologies for receivers pruning off the network in Fig. 5.1. The total number of transmissions increases with the increasing loss probability while it decreases with the increasing pruning probability λ. This is because more links may prune from the multicast

group and the total number of links decreases for large λ. From Fig. 5.2, the effect of receiver pruning on E[M] is not significant for small λ. However, Fig. 5.3 shows that the bandwidth consumption E[C] decreases as λ increases. These results have tendencies similar to those of the above number of transmissions.

5.2.2 The Pruning of Subnets

The pruning of intermediate routers will result in the pruning of subnetworks. If some subnetworks are pruned, the remaining network changes significantly, in that the total number of links and topology change according to the pruning that has occurred. This process resembles the burst pruning of links.

As shown in Fig. 5.4, if one subnet prunes, the initial multicast group is divided into two parts: the pruned subnet and the remaining subnet. Let N_0, N_i, and N be the number of links in the initial multicast network at the start of the session, in the pruned subnet, and in the remaining subnet, respectively. Then $N_0=N_i+N$. Let u_0, u_i, and u be the topology parameters in the initial multicast network, in the pruned subnet, and in the remaining subnet. Let n_k be the number of links under node k for the initial multicast network. Here we have $n_0=N_0$ if the sender is node 0.

From (5.3), the topology parameter u_0 of the initial multicast network is clearly divided into two parts, the summation of the remaining and pruned subnets.

$$u_0 = 2\sum_{k\in N_0} n_k + N_0 = 2\sum_{k\in N} n_k + N + 2\sum_{k\in N_i} n_k + N_i \tag{5.21}$$

where $2\sum_{k\in N_i} n_k + N_i$ is the topology parameter u_i of the pruned subnet, i.e.

$$u_i = 2\sum_{k\in N_i} n_k + N_i \tag{5.22}$$

However, $2\sum_{k\in N} n_k + N$ is not the topology parameter u of the remaining subnet because the pruning of subnets causes the n_k of some nodes to change.

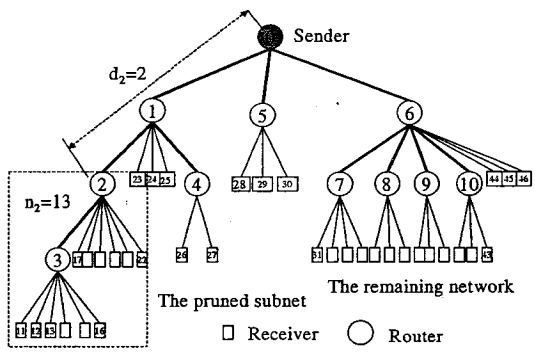


Fig. 5.4 The pruning of one subnet.

After the pruning of one subnet, the n_k values of some nodes change (n_k is the number of links under node k) for the remaining network of population N. For example, $n_1=20$ before the pruning of subnet 2, and $n_1=7$ after the pruning of subnet 2 in Fig. 5.4. If one subnet under node k prunes, the n_k value for node k decreases by N_i where N_i is the number of links for the pruned subnet. Assume that the pruned subnet is in level d_i. Then, for d_i -1 nodes along the path between the sender and node i, their n_i values will decrease by N_i. For the

nodes that do not belong to the path between the sender and node i, their n_i values remain unchanged. One therefore obtains the topology parameter u for the remaining subnet, i.e.

$$u = 2[\sum_{k \in N} n_k - (d_i - 1)N_i] + N \tag{5.23}$$

From (5.21), (5.22), and (5.23), one can obtain the following equation:

$$u_0 = u + u_i + 2N_i(d_i - 1) \tag{5.24}$$

For the pruning of one subnet, $N_i=n_i+1$, i.e. the link i and all links n_i under node i.

After determining u and N, one can easily calculate the number of transmissions E[M] for the remaining network:

$$E[M] = g(N) + up^2 \tag{5.25}$$

where $g(N)$ is given by (5.2), u is given by (5.24). Fig. 5.5 shows the result of the number of transmissions versus the pruning of one subnet.

For these subnets, these parameters are related to each other; the number of transmissions for the remaining subnet depends on the pruned subnet and on the initial multicast network topology. For the pruned subnet and the initial multicast net, the number of transmissions $E[M]_i$ and $E[M]_0$ can be expressed as follows:

$$E[M]_s = g(N_s) + u_s p^2, \qquad s = 0, i \tag{5.26}$$

Substituting (5.25) and (5.26) into (5.24), one obtains

$$E[M]_0 - E[M] - E[M]_i = g(N_0) - g(N) - g(N_i) + 2N_i(d_i - 1)p^2 \tag{5.27}$$

where function $g(N)$ is given by (5.2).

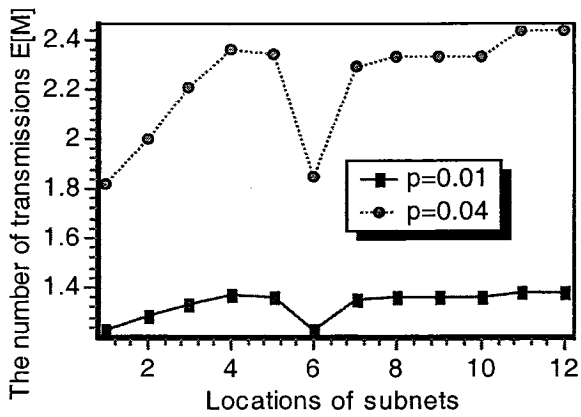


Fig. 5.5 The number of transmissions after the pruning of one subnet for the multicast tree shown in Fig. 5.4.

If two subnets are simultaneously pruned from the multicast network, the multicast network changes considerably. For the pruned subnetwork rooted at node i and j, we obtain a similar equation in the same way as we did for the pruning of one subnet.

$$\begin{aligned} & E[M]+E[M]_i+E[M]_j-E[M]_0 \\ & =g(N)+g(N_i)+g(N_j)-g(N_0)-2N_i(d_i-1)p^2-2N_j(d_j-1)p^2 \end{aligned} \tag{5.28}$$

where $N_i=n_i+1$, $N_j=n_j+1$, $N_0=N_i+N_j+N$.

Fig. 5.5 and Fig. 5.6 show the results of the number of transmissions E[M] for the subnet pruning where the x axis denotes the locations of subnets. Due to a noticeable change in the network topology, E[M] also changes drastically, depending on the locations of pruned subnets. Therefore, every time subnets prune, the multicast group needs to redetermine the locations of RRs; otherwise, E[M] would degrade considerably, a discussion of which will follow.

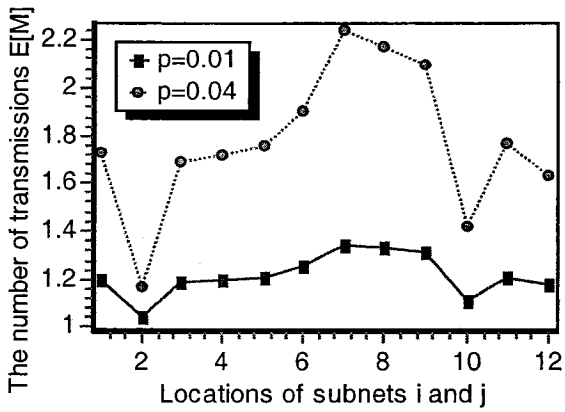


Fig. 5.6 The number of transmissions after the pruning of 2 subnets for the multicast tree shown in Fig. 5.4. The locations of subnets corresponding to the values of the x axis are shown in Table 5.2.

Table 5.2 The values of the x axis corresponding to the locations of two subnets rooted in node i and j shown in Fig. 5.6.

X Axis	1	2	3	4	5	6
Pruned subnets	1,5	1,6	1,7	1,8	5,6	2,5
X Axis	7	8	9	10	11	12
Pruned subnets	4,5	5,7	3,5	2,6	4,6	3,6

5.3 Optimal RR Placements –A New Algorithm for RR Router Assignment in Dynamic Trees

The topology of a real multicast session may change; as receivers join and leave the multicast session, the multicast tree will change. Furthermore, routers may be pruned from the network and new routers may join the session. In this section, we will simulate the dynamics of multicast networks and investigate a new policy for finding the best RR

locations based on the parallel processing of a certain 3-phase algorithm at each router. All routers are assumed to have RR capability; however, only a few of these routers dynamically enable this capability in order to optimize the overall performance in terms of bandwidth consumption. The 3-phase distributed algorithm that follows is the vehicle for selecting some routers to enable this RR function. We assume that all receivers may leave the multicast session with the probability λ_P. According to the Poisson process, new receivers (or routers) join the session at each router at a rate of λ_J (or λ_s). The initial network topology is shown in Fig. 5.7, and then the dynamic network is simulated according to the flowchart as shown in Fig. 5.9.

Dynamic network topologies change with time. The best RR allocations require an adaptive, distributed, and scalable policy. The best RR locations should efficiently recover the losses of various local loss patterns that occur more often. In order to focus on the effects of the different topologies, we assume here that the network has a homogeneous link error probability. Let n_k be the number of links under node k and d_k be the number of links from the sender to node k. Fig. 5.7 shows an example of multicast networks that explains such notations.

Worth noting in this thesis, many routers may have embedded RR capability but this capability of recovering errors and loss is only enabled at few routers not all. This minimizes congestion, NAK implosions etc in face of extra traffic added by the ARQ processes within the RR functionality. We will minimize bandwidth consumption to obtain

the optimal RR locations where the bandwidth is the total number of all links affected by one source multicast packet.

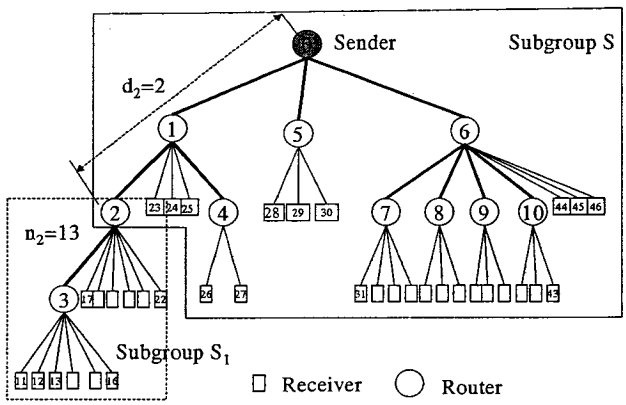


Fig. 5.7 One example of a general multicast network.

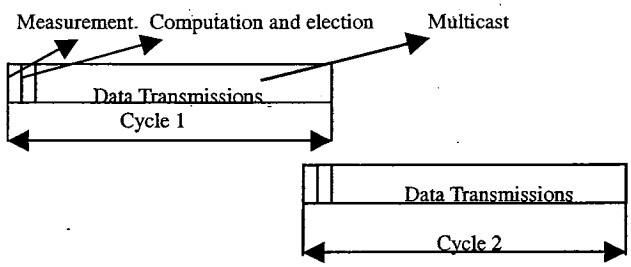


Fig. 5.8 A typical cycle of the 3-phase algorithm.

The purpose of the 3-phase algorithm is to optimally appoint a new router to become a RR, an appointment that takes place on every cycle. The cycle consists of the measurement phase, the computation and election phase, and the multicast phase. Different cycles are pipelined as in Fig. 5.8. A typical cycle length is 10000 packets; for example, the first phase (typically 10 packets) is used to transmit control topology information, as will be detailed. The second phase (typically 10 packets) is used by routers to compute the bandwidth used

and to elect a new RR. The third phase is the multicast data transmission phase, which consists of remainder of the 10000 packets. Due to the pipeline operation, as seen in Fig. 5.8, no time is wasted in the first 2 phases; however, some link bandwidth is used for the upstream transmissions of control packets. Other control information is also piggybacked on multicast packets downstream, thus wasting little capacity. The following shows the details of the 3-phase algorithm for finding the best RR locations.

Phase 1(measurements):

Each router is made aware of the number of links downstream and the number of links from the sender to that router, i.e. n_k and d_k, shown in Fig. 5.7, where the sender is node $k=0$. To acquire this knowledge about all n_k and d_k at each node, each router will, upon joining an upstream router, get the d_k value ($d_0=0$ at sender) from this upstream router and increase it by one. In this way, the d_k value will propagate down as routers join the upstream router. Similarly, the n_k value propagates up from downstream to upstream routers. A new router or receiver joining at the root of the tree will cause the update of the n_k values of all of its upstream routers up to the sender. The update of d_k takes place by piggybacking on regular data multicast packets. The update of n_k occurs on small control packets that are sent upstream. In this thesis, we leave bandwidth consumption loss due to the neglect of such control packets to future research.

Based on the information of n_k, each router can locally calculate the topology parameter u_k of the subtree T_k rooted in this router, which is defined as

$$u_k = 2\sum_{k_1 \in T_k} n_{k_1} + n_k \tag{5.29}$$

where we sum up only the routers under router k, u_k of node k can be recursively obtained from (5.29) through the use of the topology parameters of its children nodes.

$$u_k = \sum_{k_1 \in child(node.k)} u_{k_1} + e_k = n_k + \sum_{k_1 \in child(node.k)} (u_{k_1} - n_{k_1}) \tag{5.30}$$

where e_k is the number of children nodes for node k.

The sender will then be able to compute the total number of links n_0 and the topology parameter u_0 for the current network, and to multicast n_0 and u_0 to all nodes using piggybacking on one multicast data packet.

Phase 2 (computation, election and recognition):

Following phase one, all routers execute the same distributed algorithm parallel in real time, which is fed by the same control information. That is, the values of n_k and d_k; the locally calculated value of u_k in regards to all routers under this router; and u_0 and n_0 received from the sender or from the upstream RR if it has already been appointed. Each node will have all such control values at the end of phase 1. Needless to say, all nodes arrive at same conclusion in regard to the election of RR nodes in a distributed manner. According to this algorithm, each router (for example, i^{th} router) calculates K values of total bandwidth consumption corresponding to K possible router locations.

Suppose that router k is a RR ($k=1,2,...,K$). The total bandwidth consumption of a multicast group with n_0 links and the topology parameter u_0 is a summation of two

subgroups, i.e. subgroup S_k covered by RR, and subgroup S covered by the sender. Fig. 5.7 shows such an example of the multicast group partition due to the possible appointment of RR. If E[C] is denoted as the bandwidth consumption that is the total number of all links affected by one source multicast packet, we have the following equation:

$$E[C] = E[C(S)] + E[C(S_k)] = N(S)E[M(S)] + N(S_k)E[M(S_k)] \tag{5.31}$$

where E[C(S)] and E[C(S_k)] are the bandwidth consumed in sender subgroup S and RR subgroup S_k, respectively. E[M(S)] and E[M(S_k)] are the expected number of transmissions for subgroups S and S_k. N(S) and N(S_k) are the number of links for subgroups S and S_k. $N(S) = n_0 - n_k$ and $N(S_k) = n_k$, where n_k is the number of links under node k.

It is easy to calculate the E[M(S_k)] of subgroup S_k ($E[M(S_k)] = g(n_k) + u_k p^2$ where $u_k = 2\sum_{k_i \in T_k} n_{k_i} + n_k$ and $g(n_k) = 1 + n_k p - \binom{n_k}{2} p^2 + \binom{n_k}{3} p^3$), because router k knows the n_k and u_k of subgroup S_k. However, the sender (or upstream RR) subgroup S is the reduced multicast subnetwork handled by the sender (or upstream RR) alone. Its E[M] value can be obtained through the following, $E[M(S)] = g(n_0 - n_k) + up^2$ where u can be obtained similar to the derivation of (5.24).

$$u = u_0 - u_k - 2n_k d_k \tag{5.32}$$

Substituting these parameters into (5.31), each router can calculate the total bandwidth E[C].

$$E[C] = (n_0 - n_k)[g(n_0 - n_k) + (u_0 - u_k - 2n_k d_k)p^2] + n_k[g(n_k) + u_k p^2] \tag{5.33}$$

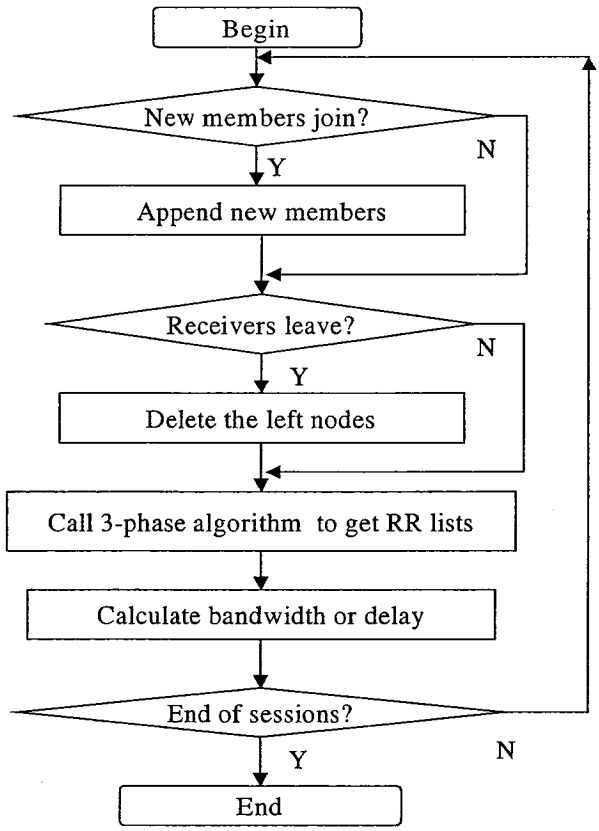


Fig. 5.9 The flowchart of dynamic networks.

Each router uses the above analysis results to compare many values of E[C]. The first such E[C] value assumes that RRs are only the sender (or upstream RR) and the k^{th} node. Another E[C] assumes that RRs are only the sender (or upstream RR) and the parent node. Another E[C] assumes that RRs are only the sender (or upstream RR) and one of the children nodes. Other E[C]s assume that RRs are only the sender and one of the sibling nodes. This k^{th} router appoints itself as a RR only if the corresponding $E[C]_k$ provides the minimal value over all other possible E[C] values above, in which case the sender is accordingly notified. Fig. 5.10 shows such process for electing a RR. It is possible that the

number of self-appointed RRs will grow, cycle after cycle. However, increasing the length of cycle time (Fig. 5.8) will protect against such an occurrence, for, as cycles evolve, users prune, and routers will have a smaller population under them to justify their self-appointment.

Once a router becomes a RR, it will compute the value E[C] (5.33). In every cycle, if pruning causes the E[C] to become less than a certain low threshold, this RR will resign. The parent RR (or the sender) will know about this recognition by the automatic exchange of the n_k and d_k values.

Phase 3 (multicast):

Following phase 2, each newly appointed RR uses control packets to convey the values n_k and u_k upstream, and to convey d_k downstream. Suppose that router i is a new RR. The RR informs all routers within the path from the sender to router i that no node under router i need ask for retransmissions from the sender. In this case, the routers locally update their values of n_k and u_k, i.e. $n_k' = n_k - n_i$ and $u_k' = u_k - u_i - 2n_i(d_i - d_k)$, where n_k' and u_k' are associated with router k. All routers under router i will also locally update their d_k values, i.e. $d_k' = d_k - d_i$, to reflect the new RR self-appointment once they become aware of the new appointment. So the RR is responsible for the retransmission of all nodes downstream, while the sender is responsible for the retransmission of the remaining links. The multicast of data packets then commences from the sender with only the sender and the RRs handling the ARQ process. Once the network topology changes, the above process repeats

periodically in order to dynamically adapt to the changes in the network topology.

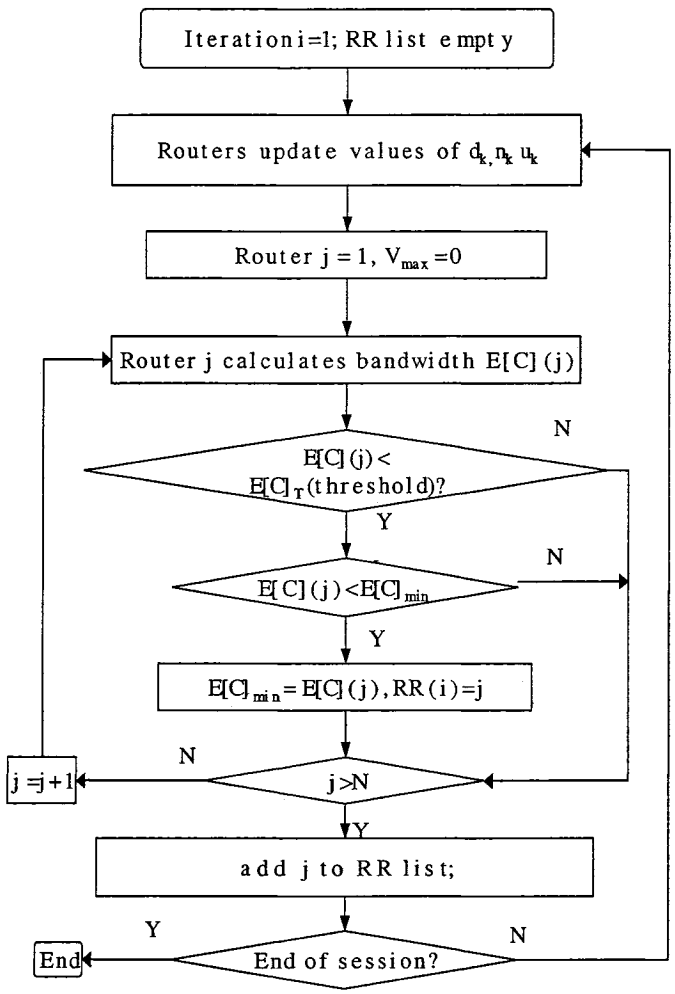


Fig. 5.10 The flowchart of the 3-phase algorithm.

5.4 Results and Discussions

For dynamic networks, the members of multicast sessions may randomly prune from the network and new members may join, which results in the network topology changing accordingly. For different topologies, optimal RR locations may be different. Thus, optimal

locations are functions of topology. Based on the above 3-phase algorithm, we can find the optimal locations of RRs; for example, the optimal RR location of 3-ary trees is in level 1.

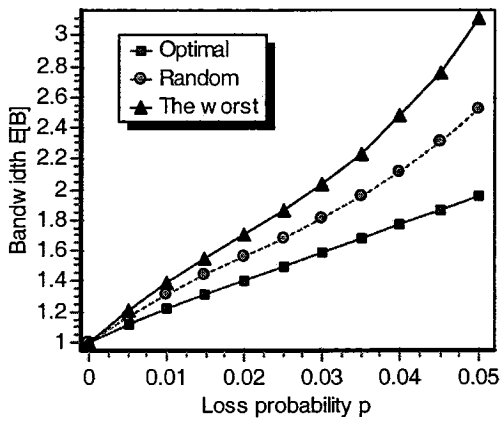


Fig. 5.11 A comparison of the bandwidth consumption for optimal, random, and the worst RR locations versus the loss probability p in the case of 1 RR appointment.

Table 5.3. The optimal RR locations in the case of the pruning of one subnet

Pruned subnet (node)	1	2	3	4	5	6	7	8	9	10
Location of RR (node)	6	6	6	6	1	2	1	1	1	1

If RRs are not selected optimally, a lot of bandwidth will be wasted. In the following, we compare the bandwidth consumption of the optimal placement (the minimal bandwidth consumption), the worst placement (the maximal bandwidth consumption), and the random placement of RR. For the sake of convenience, we first restrict the results of this thesis to the case where only one RR is appointed, where the sender selects the best RR out of many self-appointed ones. Fig. 5.11 provides a comparison of 1 RR for the network shown in Fig. 5.1. From Fig. 5.11, the bandwidth difference among the optimal placement, the worst placement, and the random placement increases with the loss probability p. When

$p=0.05$ per link, the bandwidth saving between the optimal and the random location reaches $(2.5\text{-}2.0)/2.0=25\%$.

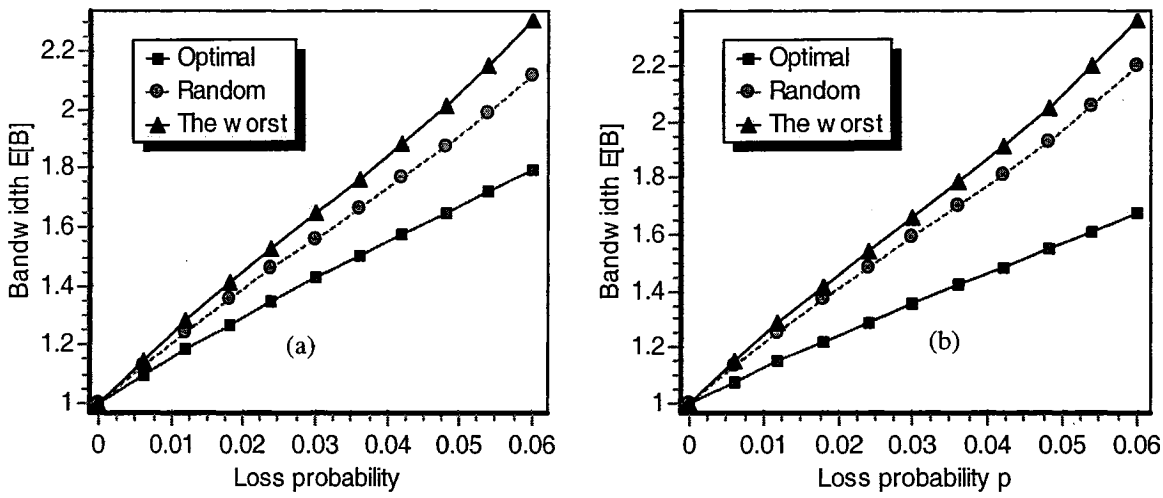


Fig.5.12 A comparison of the total bandwidth in the case of 1 RR. a: subnet 1 rooted at node 1 is pruned, b: subnet 6 rooted at node 6 is pruned

Although the effect of receiver pruning is small in Fig. 5.2, the burst pruning of subnets has a significant influence on multicast performance in Fig. 5.5 and Fig. 5.6. If one or more subnets prune from the multicast group, the optimal RR locations may change accordingly, depending on the locations of the pruned subnets. Table 5.3 shows the optimal locations of the remaining multicast network, based on the 3-phase algorithm after the pruning of 1 subnet. For example, if the subnet rooted at node 6 prunes, the optimal RR location changes to node 2. For the remaining multicast network without subnet 6, we also compare the total bandwidth consumption for the optimal, the random, and the worst location of 1 RR in Fig.5.12. The bandwidth saving is obvious. If RR is inappropriately selected at, for

example, the worst location, lots of bandwidth is wasted. Even if RR is randomly selected, the wasted bandwidth is significant.

The random pruning of receivers may take place in different subnets, while the pruning of subnets is confined to a subtree. Assume that the average number of pruned links is the same in both cases, i.e. the size of the pruned subtree for subnet pruning equals λw of the receiver pruning case.

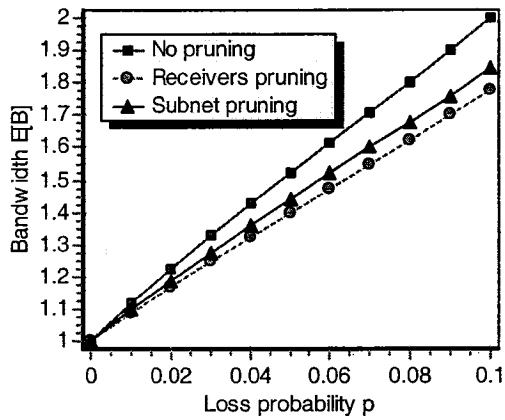


Fig.5.13 A comparison of the total bandwidth between receiver pruning and the subnet pruning for the topology of type B in Fig. 4.2.

We compare the results of the receiver pruning with 1 RR to the subnet pruning with 1 RR in the following. Fig.5.13 shows the results for the topology of type B in Fig. 4.2 of page 63, where $G_1=G_2=...=G_L=G=6$ and $\lambda=1/3$ so that the average pruned links is the same in both cases. It is interesting that both pruning cases have similar performances, however, the receiver pruning must have a very large pruning probability λ to reach the same performance as the small subnet pruning.

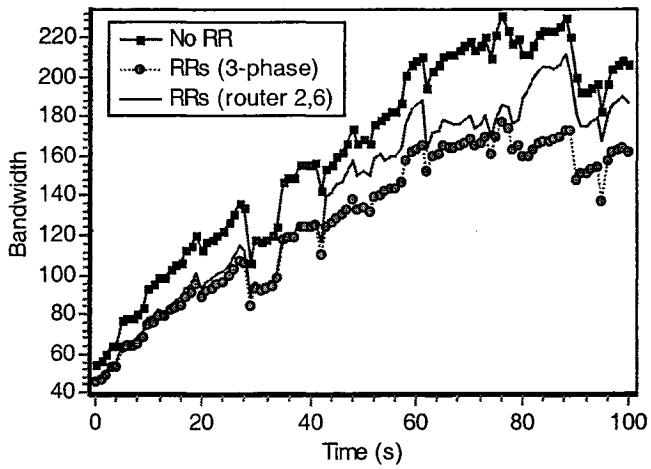


Fig. 5.14 The bandwidth performance E[C] of dynamic networks. λ_P=0.2,λ_J=0.8, p=0.01.

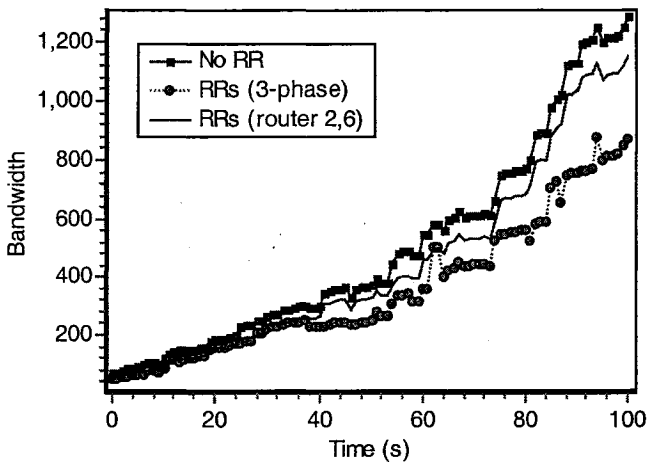


Fig. 5.15 The bandwidth performance E[C] of dynamic networks. λ_P=0.2,λ_J=0.7,

λ_S=0.05, p=0.04

Some simulation results of the 3-phase algorithm will be shown later in the following chapter. This compares to the analysis results in Fig. 5.14 and Fig. 5.15 where λ_P is the probability that a receiver prunes from the multicast session, and new receivers (or routers) join the session at each router at a rate of λ_J (or λ_s). In order to focus on the impact of dynamic networks, we utilize large values of joining and pruning parameters. The presence

of RRs demonstrates the obvious benefits of bandwidth savings. In the beginning, we find the best RR locations: routers 2 and 6. When the network evolves, we record a set of bandwidth for fixed RR locations. Another set of data is the bandwidth in the presence of the RRs, found by the 3-phase algorithm. Due to the change in the network, the previous RR locations may no longer be optimal. The RR locations given by the 3-phase algorithm always yield lower bandwidth consumption.

5.5 Conclusions

We have analyzed the performance of a few dynamic multicast networks and have discussed the effects of dynamics on the optimal partitions of multicast trees. The pruning of networks can be classified into two types of pruning: the random pruning of receiver links, and the burst pruning of subnets. Each has a different effect on the performance of networks. Random receiver pruning has less influence on multicast performance. This influence only becomes significant if the pruning probability is high enough. The burst pruning of subnets has a significant influence on the multicast performance.

The bandwidth consumption increases with the increasing loss probability p and decreases with the increasing pruning probability λ. We have found that the RR locations have large influence on multicast performance. If an RR is optimally selected, up to *50%* of the bandwidth can be saved. We have compared the total bandwidth consumption of the optimal, random, and the worst location of 1 RR. These results show that a lot of bandwidth is wasted if RR is not optimally selected.

We have suggested a new algorithm to adapt the optimal RR location to the dynamic change of the network topology based on bandwidth consumption. The parameters n_k and d_k are used to determine the topology of network. They change as any receivers or routers get pruned or join the multicast session. The RR locations selected by the 3-phase algorithm can save much bandwidth consumption of dynamic multicast networks.

Chapter 6 SIMULATIONS OF OPTIMAL ALLOCATION OF RRS IN RELIABLE MULTICAST NETWORKS

6.1 Introduction

Providing efficient reliable multicast support for large-scale multi-point applications has been proposed in the past few years [6], [9], [40], [47]. Loss recovery techniques are required to guarantee the reliability of multicast transmissions. Server-based recovery provides significant performance gains in terms of bandwidth and delay. However, repair servers perform efficiently only when they are properly placed. It is inefficient and unrealistic to have repair services enabled at all routers. It is therefore crucial to select some routers to retransmit lost packets.

Repair services are often applied to local recovery [29], [43], [49], [66], [95]. Repair servers locally retransmit the lost packets once they receive negative acknowledgements (NAK). This local recovery may use the Automatic Repeat reQuest (ARQ) [6], [43], [49] or the Forward Error Correction (FEC) approach to handle packet losses [40]. One may also integrate FEC with ARQ to efficiently recover temporally correlated losses due to buffer overflow or to random losses due to channel errors [40]. Repair packets can be original data packets or redundancy packets for the hybrid ARQ/FEC technique [40]-[46]. Repair servers have FEC functions and are capable of decoding and encoding. In this case, it is more important to supply only a few routers with repair capability so as to reduce the

cost, the processing load, and the bandwidth used. The optimal selection of the multicast routers that will serve as recovery points is the main theme of this thesis. These will have their FEC capability enabled so as to reduce the overall multicast bandwidth.

Problems with the locations of repair services have been considered [12], [18] and addressed in the previous chapters. The 3-phase algorithm was suggested to adapt the RR location to the topology change of dynamic networks. In this chapter, we will implement a reliable multicast algorithm over the Internet where unicast background traffic is also considered. We will investigate the simulation performance of RRs selected by the 3-phase algorithm for dynamic networks. Moreover, we will compare the delay and the bandwidth performance of the new algorithm and the fixed repair locations, i.e., their locations do not change as users prune or join the multicast session. In each of the above cases, we will compare the utilization of FEC and ARQ at both sender and optimally selected routers as per the new 3-phase algorithm.

6.2 Simulation Model

In this chapter, we implement a one-to-many reliable multicast communication over the Internet using C++. In addition to multicast traffic from the sender, routers generally receive and unicast background Internet traffic. The sender multicasts packets to all receivers. Receivers that lose a packet will send NAK upstream and request a retransmission. Packets arriving at routers queue in their input buffers according to First-In-First-Out (FIFO). Only a few routers provide repair services, i.e. only some routers

retransmit repair packets once they receive NAK from downstream. They buffer the multicast packets received from upstream and retransmit them upon requests from downstream. In their turn, RRs send NAK to the sender asking for retransmissions if they lose a multicast packet. Ordinary routers simply route the packets downstream without being responsible for loss recovery. The details are provided in the following.

Multicast data packets arrive at the sender according to a Poisson process with an arrival rate of λ_M, i.e. the probability that the k packets arrive at the sender during time interval Δt is

$$P(k) = \frac{G_M^k e^{-G_M}}{k!}, \tag{6.1}$$

where $G_M = \lambda_M \Delta t$.

Besides the multicast data stream, the background Internet traffic is considered. For example, unicast traffic may arrive at each router according to a Poisson process with an arrival rate of λ_B

$$P(k) = \frac{G_B^k e^{-G_B}}{k!}, \tag{6.2}$$

where $G_B = \lambda_B \Delta t$. Received packets will be buffered at routers according to a FIFO discipline. The average service time of the packets is $\delta = 1/\mu$, where μ is service rate.

If a multicast data packet is being served at a router, this router forwards it to its children routers. If a background packet is being served, the router serves only one of the links selected randomly from its children links for the background traffic, so as to simulate the

random effects of unicast traffic on the multicast performance.

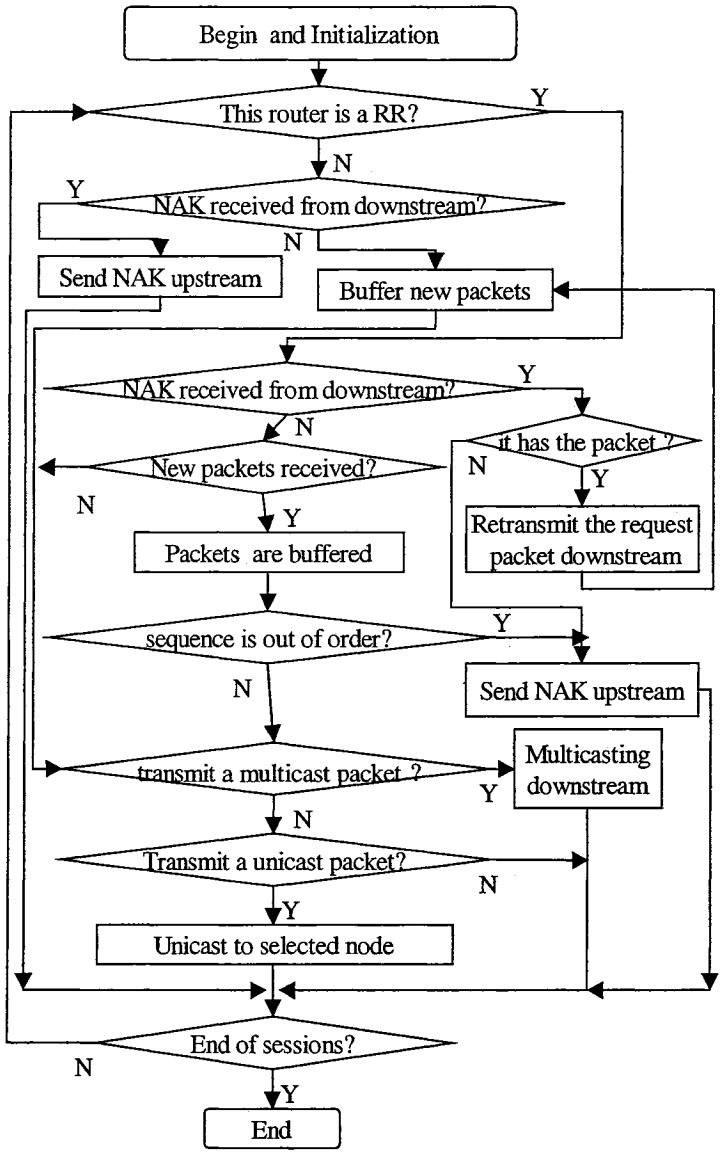


Fig. 6.1 The flowchart of multicast packet transmissions handling at the RR or regualar routers for an ARQ case.

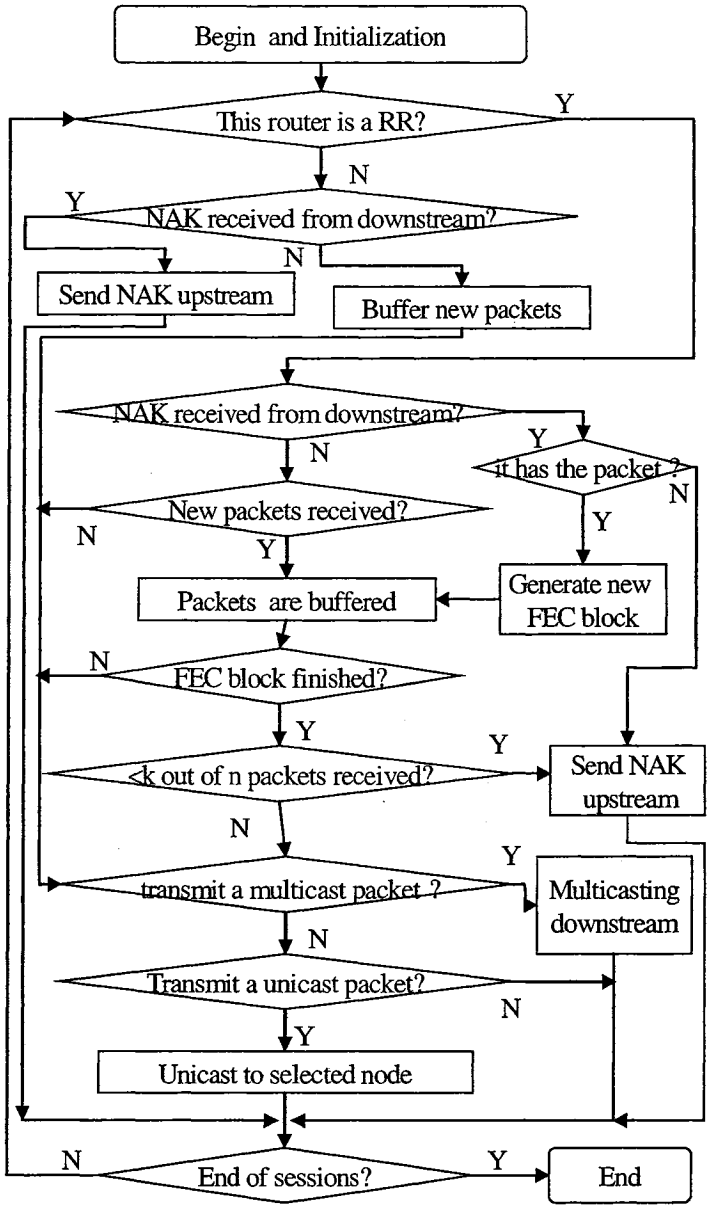


Fig. 6.2 The flowchart of multicast packet transmissions handling at a RR or a regular router for an FEC/ARQ case

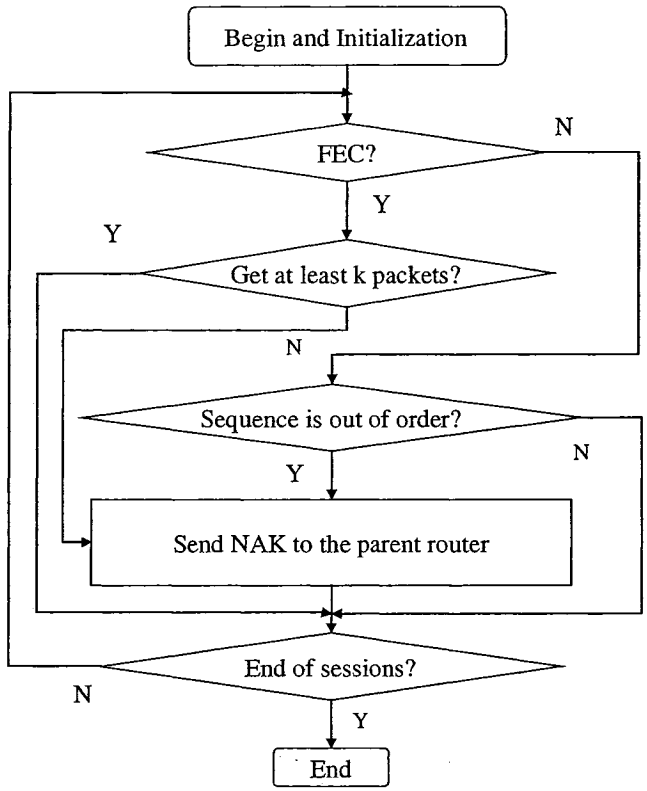


Fig. 6.3 The flowchart of multicast packet transmissions handling at end users

A multicast packet has higher priority than a background one when it queues. After n_S consecutive multicast packets are served, 1 unicast packet is served where n_S is a simulation variable. In this chapter, n_S=*3*.

The Reed-Solomon erasure (RSE) correcting code is used in the FEC case [40]. *k* original data packets are encoded into *n* packets where *n-k* parity packets are produced. If more than *k* packets are received, all data packets can be reconstructed at the receiver side. If less than *k* packets are received, more parity packets are required and retransmitted. All RRs are assumed to be capable of RSE decoding and encoding. Employing FEC or ARQ at the

router level can be seen by comparing Fig. 6.1 and Fig. 6.2. At the user level, Fig. 6.3 shows the possibility of both FEC and ARQ handling.

6.3 Bandwidth Performance

6.3.1 Simulation Measurement – Bandwidth Consumption

We start by simulating multicast transmissions. For a single-source multicast tree, a sender is located at the top of the tree. The sender multicasts the repair packet until each node receives it correctly at least one time. Different packets may experience different transmission times due to losses. In this section, we first consider the bandwidth performance of reliable multicast. Let M be a random sample of the number of transmissions. Furthermore, each transmission affects actually different links, which depend on the locations where loss takes place. In order to obtain the total bandwidth consumption C, we need to summarize all links actually affected by all multicast transmissions. Let C be a random sample of actual bandwidth consumption. Thus the sample mean E[M] (or E[C]) of the number of transmissions (or bandwidth consumption) can be averaged over many packets. Fig. 6.4 shows our simulation flowchart to obtain E[M] (or E[C]). Control traffic condition such as NAK packets was assumed ideal with neglected bandwidth consumption. Data packets were sent to stream nodes from sender.

In Fig. 6.4, we give the simulation flowchart where we average many packets to obtain the average number of transmissions. Let M_i denote the number of transmissions for one

specific multicast packet i. Averaged over θ packets (e.g., θ=*10000*), E[M] is,

$$E[M] = \sum_{i=1}^{\theta} M_i / \theta \tag{6.3}$$

The sample variance σ^2 of a random sample of size θ may be written as follows

$$\sigma^2 = \frac{\theta \sum_{i=1}^{\theta} M_i^2 - (\sum_{i=1}^{\theta} M_i)^2}{\theta(\theta-1)} \tag{6.4}$$

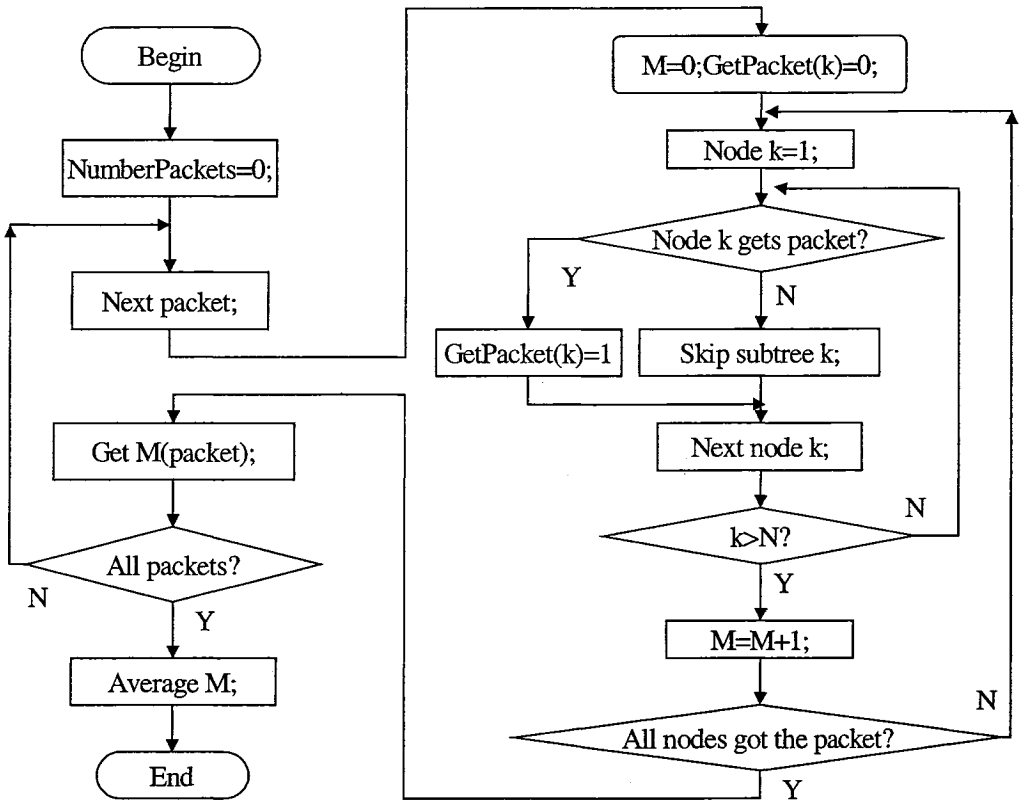


Fig. 6.4 The simulation flowchart for one subgroup

After evaluating the mean *Y* (an estimate of E[M]) and standard deviation of a random sample, we can calculate the *(1-α)100%* confidence interval for *Y* using the following formula.

$$E[M]-z_{\alpha/2}\frac{\sigma}{\sqrt{\theta}}<Y<E[M]+z_{\alpha/2}\frac{\sigma}{\sqrt{\theta}} \tag{6.5}$$

where $z_{\alpha/2}$ is the z-value leaving an area of $\alpha/2$ to the right.

6.3.2 Simulation Results

We simulate multicast transmissions and obtain the average number of transmissions E[M] over many packets. In these figures, we average the number of transmissions over 10000 packets (i.e. $\theta=10000$). This was found enough to provide us with a confidence interval of 95% about the true mean of each result in all of figures (Fig. 6.5 - Fig. 6.8). We also compare the simulation results with the approximate and exact results for some topologies.

Fig. 6.5 - Fig. 6.7 give the comparisons of several topologies among the simulation results, the exact (i.e, the earlier recursion solution) and approximate solutions for E[M]. For a small E[M] value, they match perfectly.

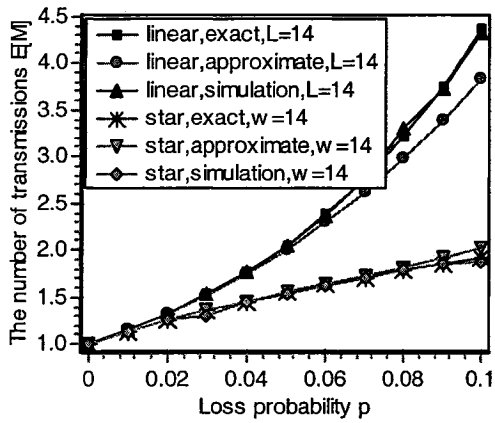


Fig. 6.5 A comparison of E[M] corresponding to analysis and simulation for the linear topology and the star topology. E[M] is obtained over 10000 packets.

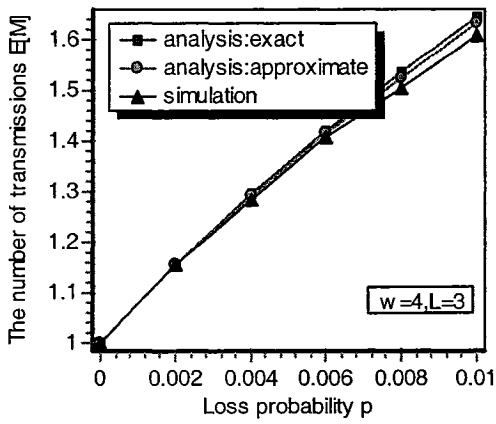


Fig. 6.6 A comparison of E[M] corresponding to analysis and simulation for 4-ary tree. E[M] is obtained over 10000 packets.

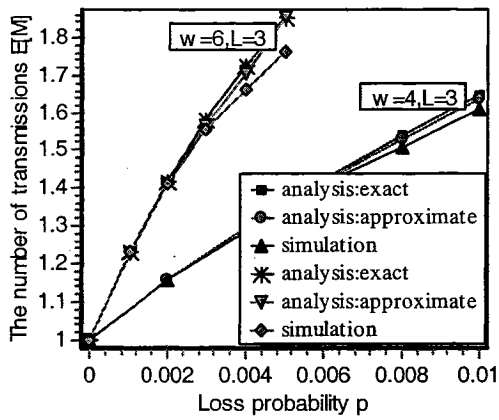


Fig. 6.7 A comparison of E[M] corresponding to analysis and simulation for the topology of type B in Fig. 4.2 of page 63 . E[M] is obtained over 10000 packets. Here w_1= w_2=...=w_L=w.

For an example of a general topology shown in Fig. 4.3, we also give the comparisons between analysis and simulations for homogeneous loss probability. Fig. 6.8 compares the analysis and simulation results where all links have the same loss probability.

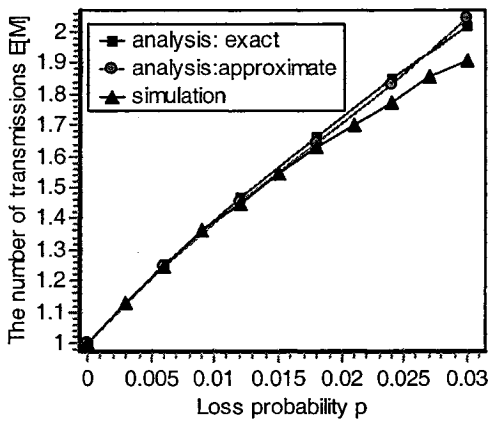


Fig. 6.8 A comparison of E[M] corresponding to analysis and simulation for the topology of type C in Fig. 4.3 of page 64. E[M] is averaged over 10000 packets.

In Fig. 6.9 and Fig. 6.10, we provide the 95% confidence interval of E[M] where $\alpha=0.05$. It is obvious that the more packets are averaged, the more accurate the simulation E[M] value will be.

For each multicast transmission, not all links are affected. If an intermediate link loses the packet in one transmission, all the nodes under this link do not receive the packet. Hence, the actual bandwidth consumption depends on the topology. In our simulation, we summarize all links actually affected by the multicast transmissions of a packet to get the bandwidth, then we average the bandwidth over 10000 packets to obtain the average bandwidth consumption E[C]. Fig. 6.11 shows the actual bandwidth consumption of different topologies. The actual bandwidth consumption of these topologies differs slightly for small loss probability in Fig. 6.11(a). With the increasing loss probability, the bandwidth difference of different topologies becomes larger and larger in Fig. 6.11(b).

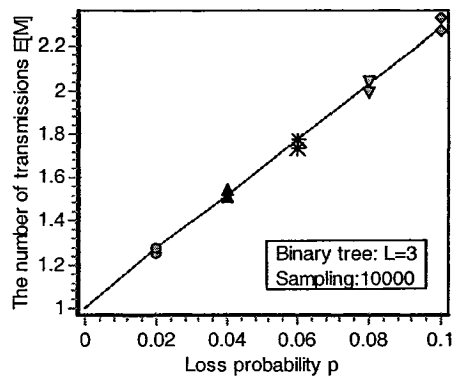


Fig. 6.9. 95% confidence interval. L is the depth of a binary tree.

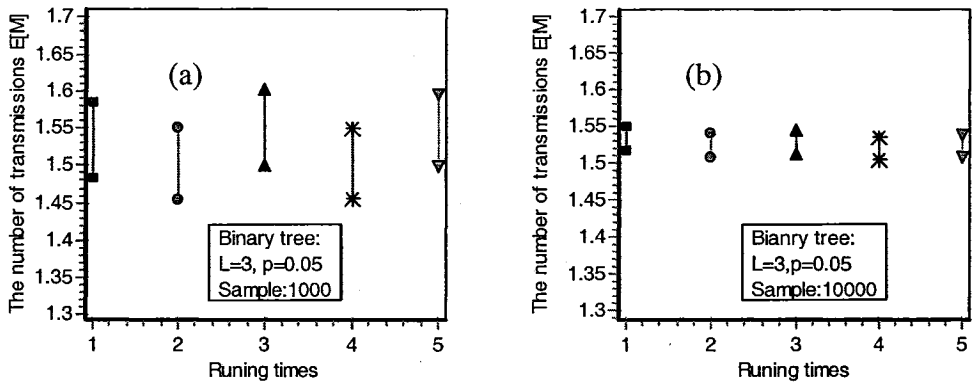


Fig. 6.10. 95% confidence interval. L is the depth, p is packet loss probability.

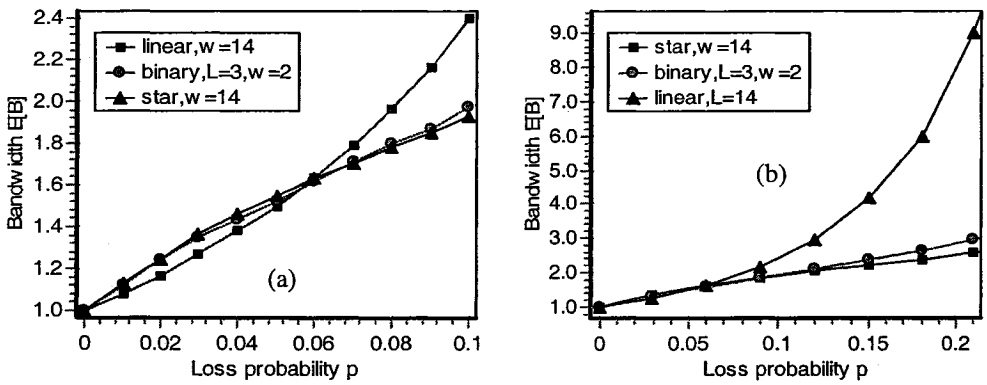


Fig. 6.11 The bandwidth consumption of different topologies versus the packet loss probability

6.4 Packet Losses due to Buffer Overflow

Losses from intermediate routers are often ignored in the performance analysis of reliable multicast [71][44]. These losses exist in an actual multicast communication, due to the buffer overflow of intermediate routers. In our simulation, we will consider the effects of intermediate links. Suppose that all routers have the same maximum buffer size A_B, and that A_{ij} is the number of packets in the queue of node i at iteration j. One may then obtain the average number of packets A_i in the buffer of router i and its average value $\overline{A}$ over all routers.

$$A_i = \frac{1}{S}\sum_{j=1}^{S} A_{ij} \tag{6.6}$$

$$\overline{A} = \frac{1}{R}\sum_{i=1}^{R} A_i \tag{6.7}$$

where R is the number of routers, and S is the number of iterations.

If a buffer is full, the arriving packets will be discarded. Assume that a_{ij}^M multicast packets and a_{ij}^B unicast background packets arrive at router i. By O_{ij}^M (or O_{ij}^B), we denote the number of multicast packets (or unicast background packets) lost in buffer i at iteration j. Then we have

$$O_{ij}^M = \max(0, a_{ij}^M - 1 + A_{ij} - A_B) \tag{6.8}$$

and

$$O_{ij}^B = \max(0, a_{ij}^B + a_{ij}^M - 1 + A_{ij} - A_B), \tag{6.9}$$

where A_{ij} is the number of packets queued in buffer i at iteration j. We can obtain the

multicast packet loss percent in buffer i and the average value over all routers. For example,

$$O_i^M = \sum_{j=1}^{S} O_{ij}^M / \sum_{j=1}^{S} a_{ij}^M \tag{6.10}$$

$$\overline{O^M} = \frac{1}{R} \sum_{i=1}^{N} O_i^M \tag{6.11}$$

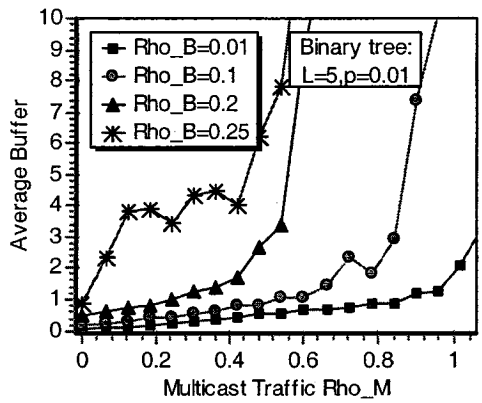


Fig. 6.12 The number of packets queued in the buffer. L is the depth of binary trees. p is the channel packet error probability.

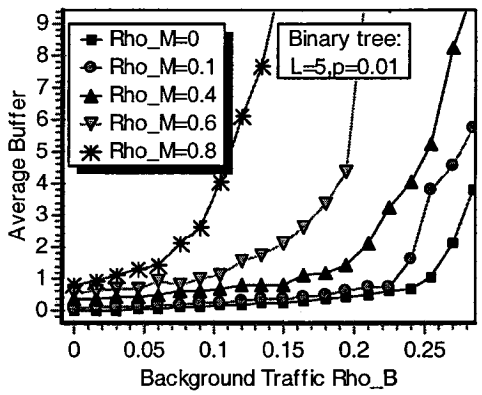


Fig. 6.13 The number of packets queued in the buffer. L is the depth of binary trees. p is the channel packet error probability

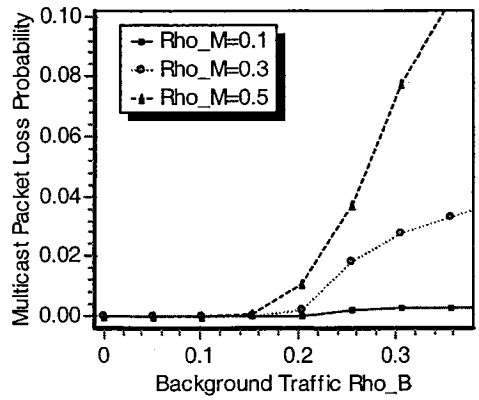


Fig. 6.14 The multicast packet loss probability versus the background traffic.

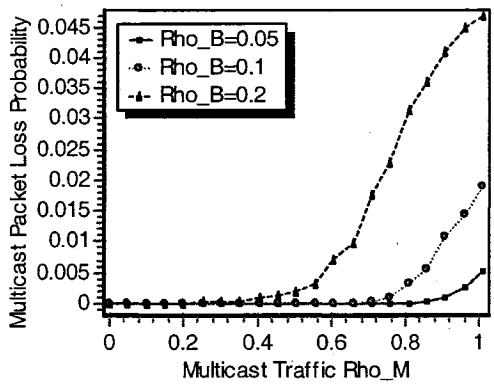


Fig. 6.15 The multicast packet loss probability versus the multicast traffic.

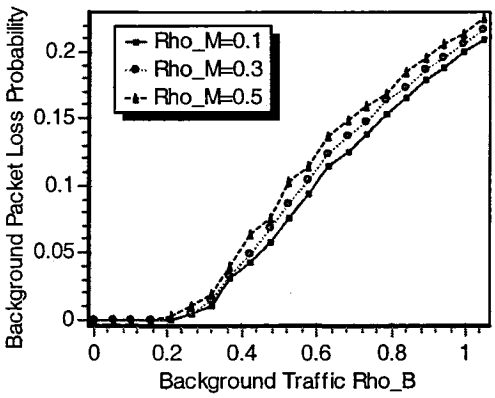


Fig. 6.16 The background packet loss probability versus the background traffic.

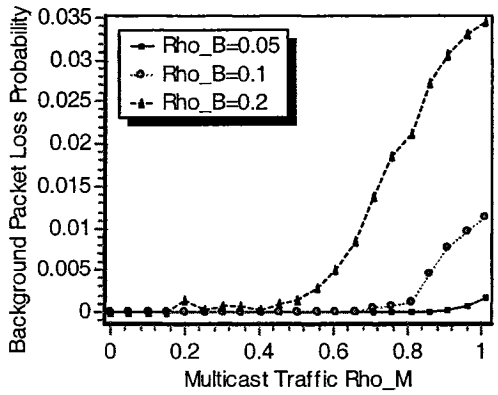


Fig. 6.17 The background packet loss probability versus the multicast traffic.

$$O_i^B = \sum_{j=1}^{S} O_{ij}^B / \sum_{j=1}^{S} a_{ij}^B \tag{6.12}$$

$$\overline{O^B} = \frac{1}{R} \sum_{i=1}^{N} O_i^B \tag{6.13}$$

Full binary trees were assumed in the simulation of this section. Fig. 6.12 - Fig. 6.17 show a sample of the simulation results where rho_M (or rho_B) is the multicast traffic intensity $\rho_M = \lambda_M/\mu$ (or background traffic intensity $\rho_B = \lambda_B/\mu$). Background traffic has a noticeable influence on multicast performance. Fig. 6.12 and Fig. 6.13 show that a larger value of ρ_B easily leads to buffer overflow, due to the lower priority of the queue. Background traffic is a main factor that causes buffer overflow and results in losses of multicast packets. Fig. 6.14 and Fig. 6.17 give the packet loss probability of multicast and background data due to buffer overflow. The packet loss probability increases with the increasing value of multicast or background traffic.

6.5 Delay Performance

In this section, we will study the delay performance that we will define shortly. A binary

multicast tree of depth $L=5$ was considered. In this section, we do not look at the new 3-phase algorithm. We assume that the retransmission of packets can only be performed by the sender (no RR here). However, both ARQ and FEC cases are considered. In our simulation, we average a large number of packet transmissions. Fig. 6.1, Fig. 6.2, and Fig. 6.3 show the flowcharts of a multicast packet transmission handling at one ARQ-type router, one FEC-type router and a typical user respectively.

The delay of a multicast packet is the time period from the instant a multicast packet is generated at the sender to the moment that all receivers receive it correctly. Clearly, it also includes the time consumed by retransmissions. Different packets may experience different delays. We obtain the mean and variance of delay over many packet transmissions, i.e.

$$\overline{T} = \frac{1}{S}\sum_{j=1}^{S} T_j, \qquad \sigma_T^2 = \frac{1}{S-1}\sum_{j=1}^{S}(T_j - \overline{T})^2, \tag{6.14}$$

where T_j is the delay of packet j, and S is the number of iterations in simulation. One iteration corresponds to one packet time. The following figures (Fig. 6.18 - Fig. 6.22) give the packet delay normalized by the average service time $\delta=1/\mu$, i.e. the value of $\overline{T}/\delta = \mu\overline{T}$. These values of delay are obtained with confidence interval of 95%.

In this section, we focus on the comparison of the delay of the ARQ and the FEC/ARQ techniques for a binary tree of depth $L=5$. Due to the transmissions of redundant packets, FEC consumes more bandwidth than the pure ARQ technique; however, it can considerably reduce packet delays. Fig. 6.18 and Fig. 6.19 make evident the benefits of

using FEC for combating packet loss. They demonstrate that FEC/ARQ outperforms ARQ because lost data packets may be recovered by other packets belonging to the same FEC block of n packets. Less retransmission is required and the delay is greatly reduced. However, this benefit is not obvious for a larger value of channel error loss probability. Packets are more frequently lost and the number of lost packets may be larger than the number of redundant packets, i.e. n-k, which is the limit of FEC capability. Thus, receivers may not decode correctly, which will cause retransmissions from the sender to increase. The packet delay increases accordingly. We can increase the value n/k of FEC to contend with the packet delay, but we must sacrifice more bandwidth due to the transmissions of more redundant packets. We can see this in Fig. 6.18. A higher value of n/k (lower coding rate k/n) is advantageous for the delay over a wider range of assumed channel packet error probability p.

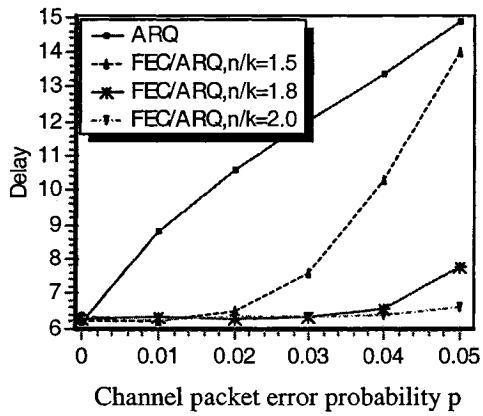


Fig. 6.18 A comparison of the delays for ARQ and FEC/ARQ in the case of binary trees. L is the depth of binary trees. ρ_M=0.3. k=10, ρ_B=0.0001. $\rho_M^{FEC} = \frac{n}{k}\rho_M^{ARQ}$.

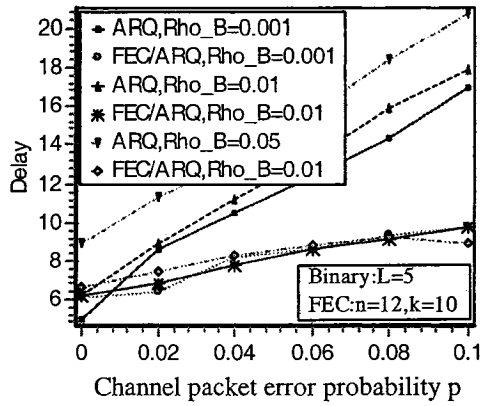


Fig. 6.19 A comparison of the delays for ARQ and FEC/ARQ in the case of binary trees. L is the depth of binary trees. $\rho_M=0.3$.

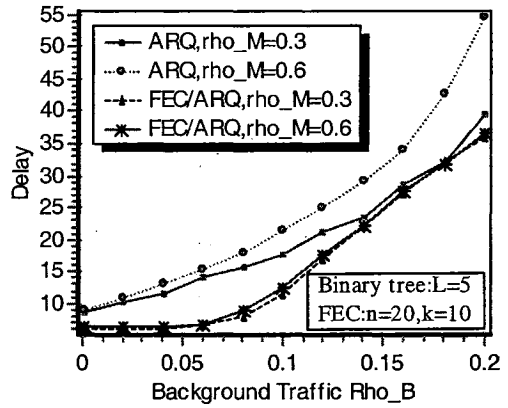


Fig. 6.20 A comparison of the delays for ARQ and FEC/ARQ. L is the depth of trees.

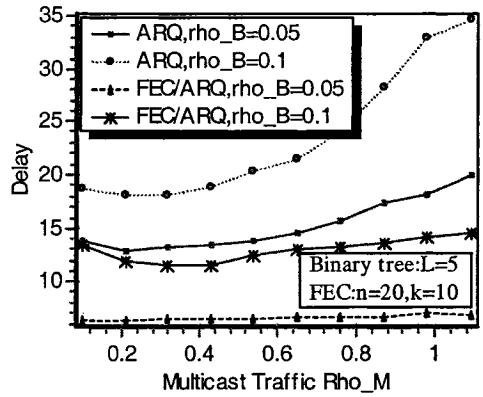


Fig. 6.21 A comparison of the delays for ARQ and FEC/ARQ. L is the depth of trees.

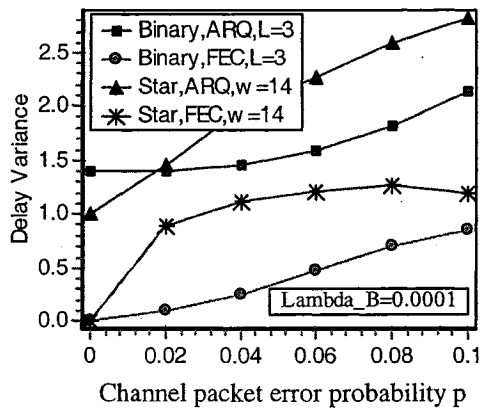


Fig. 6.22 The delay variance versus the channel packet error probability. L is the depth of trees. $\rho_M = 0.3$. w is the number of branches for the star topology.

Fig. 6.20 and Fig. 6.21 show that the effects of background and multicast traffic intensity on delay are different. Background traffic has a larger influence than multicast traffic. In Fig. 6.20, the packet delay improves with the use of FEC for smaller values of background traffic. A larger value of background traffic leads to the loss of multicast packets, due to buffer overflow. We observe from Fig. 6.21 that multicast traffic has less influence on the delay for $\rho_M < 1$. Therefore, even if FEC/ARQ and pure ARQ are assumed to have the same data traffic for fair comparison to prevail, i.e. $\rho_M^{FEC} = \frac{n}{k}\rho_M^{ARQ}$, FEC/ARQ has a better performance than pure ARQ. Some results of delay variance are shown in Fig. 6.22. The delay variance increases with the increase of the channel packet error probability.

The use of FEC can reduce the delay of a multicast packet within the specific ranges of certain parameters. For example, the large channel packet error probability or background traffic leads to the low efficiency of FEC. We see from Fig. 6.23 and Fig. 6.24 that the residual packet loss probability increases with these factors in the case of pure FEC.

Retransmissions of lost packets are required to provide multicast reliability. Only the proper use of FEC is cost effective for loss recovery.

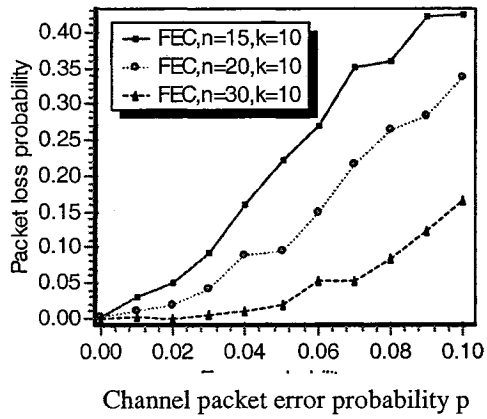


Fig. 6.23 Residual packet loss probability versus the channel packet error probability.

$\rho_M = 0.3, \rho_B = 0.01.$

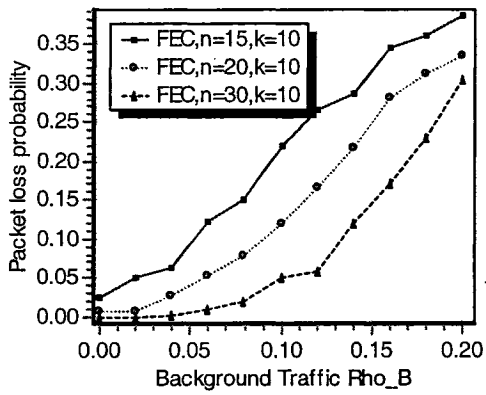


Fig. 6.24 Residual packet loss probability versus the background traffic ρ_B. $\rho_M = 0.3$, $p = 0.01$.

6.6 The RR Location Problem

6.6.1 The Optimal RR Locations for Some Network Topologies

From the analysis results in the previous chapter, the policy for placing RRs depends on the type of topologies of different subgroups. Multicast networks with homogeneous topology structure have similar topologies after partitioning according to RR locations. Thus, the optimal condition is that each subgroup should have the same size. We support this point of view by the simulation of k-ary trees. We can see these results from Fig. 6.25 and Table 6.1. For example, levels 1,1,2 are the optimal placements for 3 RRs in the case of 3-ary tree because this is the best RR combination having the same topology and the same size for 4 subgroups. Fig. 6.25 also shows that the optimal RR location is always level 1 for different number of branches w and loss probability p.

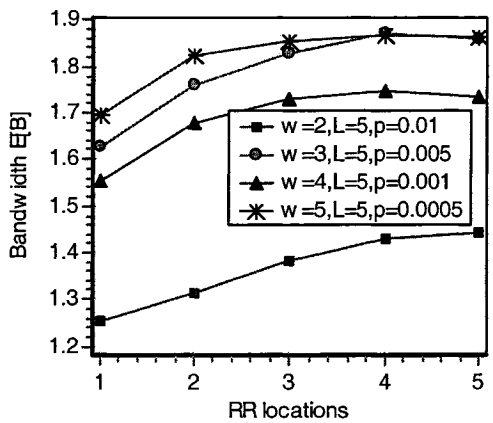


Fig. 6.25 The bandwidth E[B] of 1RR for *k*-ary trees where *w* is the number of children links for intermediate nodes and *L* is the depth of the tree, x-axis is the level of this RR.

Table 6.1 The Optimal RR Levels for *K*-ary Trees

#RRs \ w	2	3	4	5
2 RRs	1,2	1,1	1,1	1,1
3 RRs	2,2,2	1,1,2	1,1,1	1,1,1

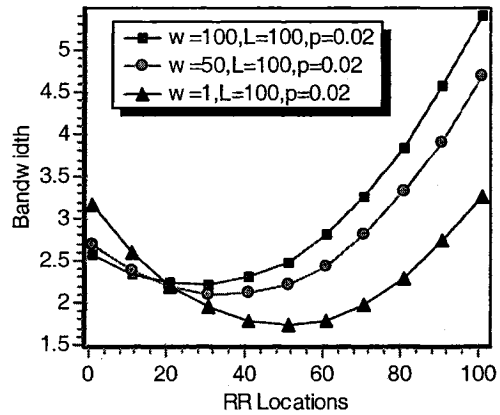


Fig. 6.26 The bandwidth E[B] of 1 RR for the topology in Fig. 4.2 of page 63 where $w_1=w$, $w_2=...=w_L=1$.Note $w_1=1$ is homogeneous topology structure

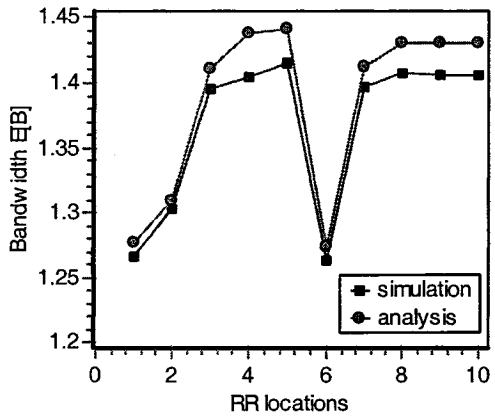


Fig. 6.27 The bandwidth E[B] of 1 RR for the network in Fig. 4.3

When each subgroup has different topology, the results of having the same size for each subgroup are not applicable. Fig. 6.26 gives the optimal RR placements where $w_1=w\neq1$, $w_2=...=w_L=1$ and the location of RR is the level of RR. For example, the optimal

placement for $w=100$ in Fig. 6.26 (square curve) is in level 30, that means, $N_0=w+30=130$ and $N_1=100-30=70$. We have different number of links for each subgroup to have the optimal placements of RRs. When we have the same size, $N_0=100$ and $N_1=100$ for $w=100$ in Fig. 6.26 (circle curve) i.e, RR is in level 1, the multicast performance is not optimal. The same number of links for each subgroup does not result in the best performance of multicast for very different topologies. This has been analyzed in the previous chapter.

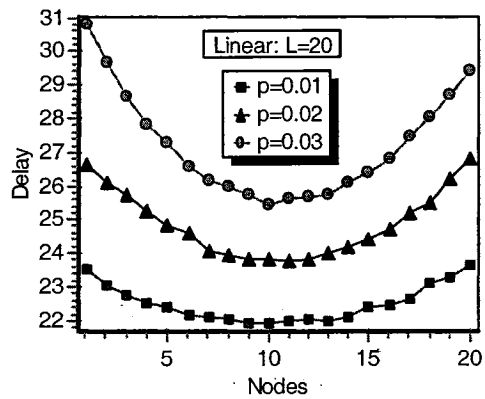


Fig. 6.28 The optimal RR location for a linear topology.

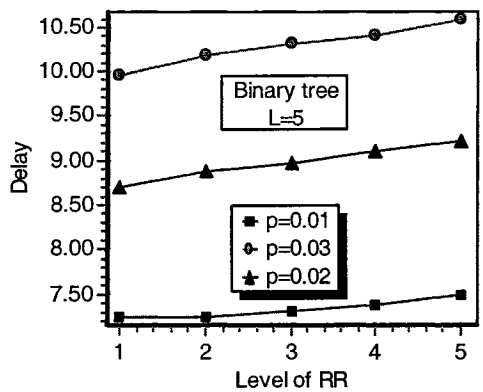


Fig. 6.29 The optimal RR location for a binary tree.

The above results indicate the optimal RR locations based on the bandwidth consumption. Fig. 6.28 and Fig. 6.29 show some simulation results based on the delay performance for a

linear tree (Fig. 6.28) and for a binary tree (Fig. 6.29). Although the use of FEC reduces delay to some extent, it is still necessary to place some intermediate repair servers to limit the scope of retransmissions for a larger network. The simulations follow the flowchart in Fig. 6.1, Fig. 6.2, and Fig. 6.3 for the RR case. In this section, different RR locations are attempted. The simulation is run and the corresponding delay is computed.

The simulation results in Fig. 6.28 and Fig. 6.29 suggest that the optimal RR location is in the middle router for the linear topology, while the optimal RR location is at level 1 for a binary tree. These results are identical to the previous analysis results of homogeneous networks based on bandwidth consumption.

6.6.2 The RR Locations of Dynamic Multicast Networks

In the previous chapter, we have proposed a new 3-phase algorithm for determining the RR locations of dynamic multicast networks. In this section, we will investigate the bandwidth and delay performance of this algorithm by way of simulations in the cases of the ARQ and FEC/ARQ. The initial network is shown in Fig. 5.7. The last chapter shows the flowchart of dynamic networks and the 3-phase algorithm.

6.6.2.1 Bandwidth Consumption

Some results of ARQ and FEC/ARQ retransmission mechanisms are shown in Fig. 6.30 - Fig. 6.35. From these results, the presence of RRs make obvious the benefits of bandwidth saving. In Fig. 6.30, we compare the bandwidth consumption for different RR locations. In

the beginning, we find the best RR locations, i.e. routers 2 and 6. When the network evolves, we record a set of the bandwidth for fixed RR locations. Another set of data is the bandwidth in the presence of the RRs that was found by the 3-phase algorithm. Due to the change in the network, the previous RR locations may not be optimal. The RR locations given by the 3-phase algorithm always give lower bandwidth consumptions. Bandwidth consumption and delay in the following figures are obtained with confidence interal of 95%.

In Fig. 6.31 - Fig. 6.33, some results are given for the use of FEC. Fig. 6.31 shows that the multicast network benefits much from the retransmission of RRs with high FEC coding rates (*k/n=10/12=5/6*). However, such benefits of the optimal location of RRs can be compromised by the use of an FEC with a lower coding rate, because more bandwidth is wasted in the transmissions of redundant packets. We can see from Fig. 6.32 that the bandwidth saving is not obvious for the use of FEC with *k/n=2/3*. Even if a low coding rate is used for FEC, the bandwidth saving for the small packet loss probability is not obvious. Comparing Fig. 6.30 to Fig. 6.33 in the case of $p=0.01$, it is found that a RR with pure ARQ can save *20-30%* of the bandwidth, while one with FEC wastes 20% of the bandwidth with redundant packets. The total bandwidth saving is not obvious; therefore, the use of FEC is limited to some ranges of parameters. The improper use of FEC may provide a limited improvement of bandwidth consumption.

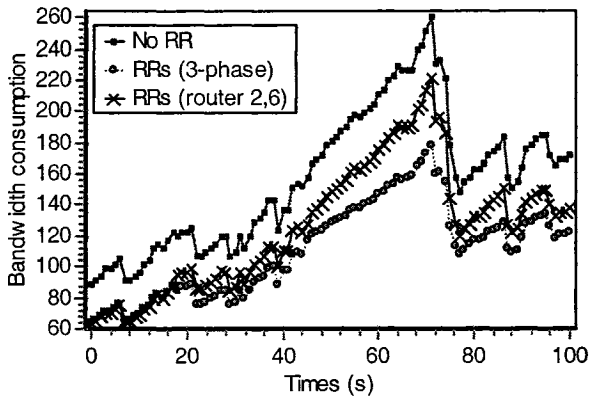


Fig. 6.30 The bandwidth performance of dynamic networks for pure ARQ retransmissions.

λ_P=0.2, λ_J=0.8, p=0.01.

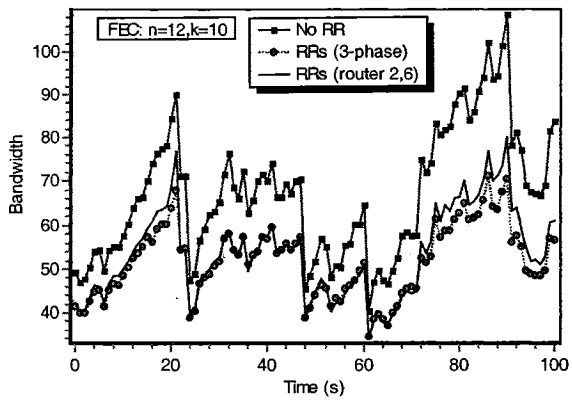


Fig. 6.31 The bandwidth performance of dynamic networks for FEC/ARQ retransmissions.

λ_P=0.2, λ_J=0.7, p=0.01

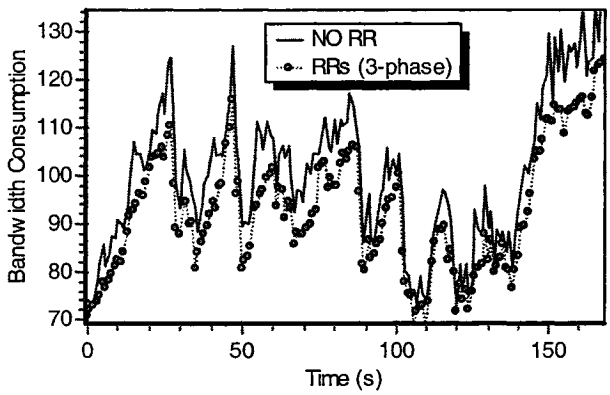


Fig. 6.32 The bandwidth performance of dynamic networks for FEC/ARQ retransmissions.

λ_P=0.2, λ_J=0.7, p=0.04, FEC: n=15, k=10.

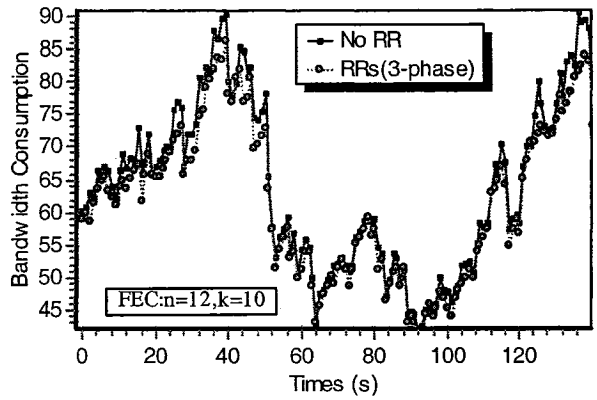


Fig. 6.33 The bandwidth performance of dynamic networks for FEC/ARQ retransmissions.

$\lambda_P=0.2$, $\lambda_J=0.7$, $p=0.01$.

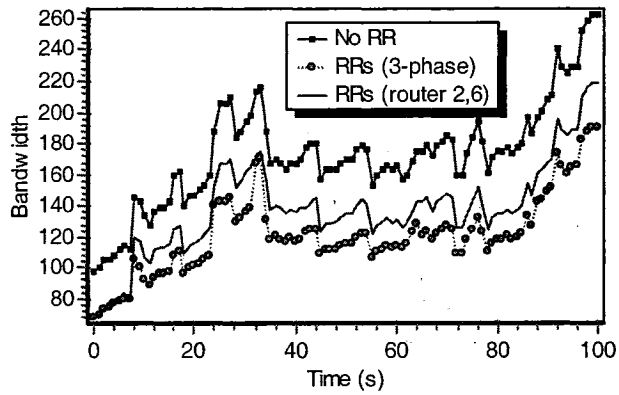


Fig. 6.34 The bandwidth performance of dynamic networks for ARQ retransmissions.

$\lambda_P=0.2$, $\lambda_J=0.7$, $\lambda_S=0.05$, $p=0.04$.

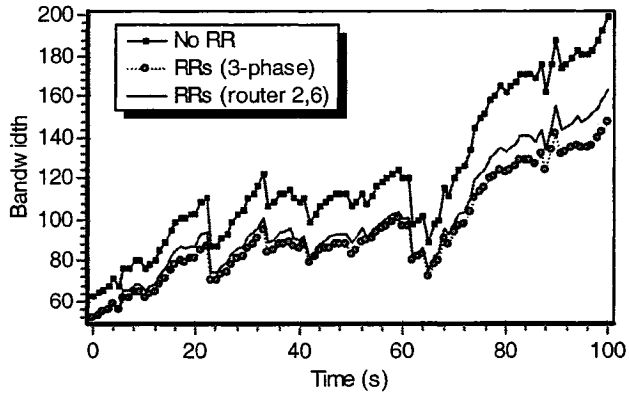


Fig. 6.35 The bandwidth performance of dynamic networks for ARQ retransmissions.

$\lambda_P=0.2$,$\lambda_J=0.7$, $\lambda_S=0.05$, $p=0.01$.

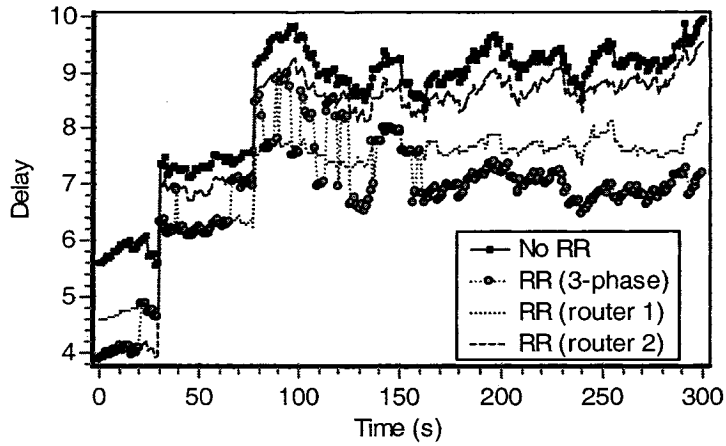


Fig. 6.36 The delay performance of dynamic networks for pure ARQ retransmissions.

λ_P=0.2, λ_J=0.5.

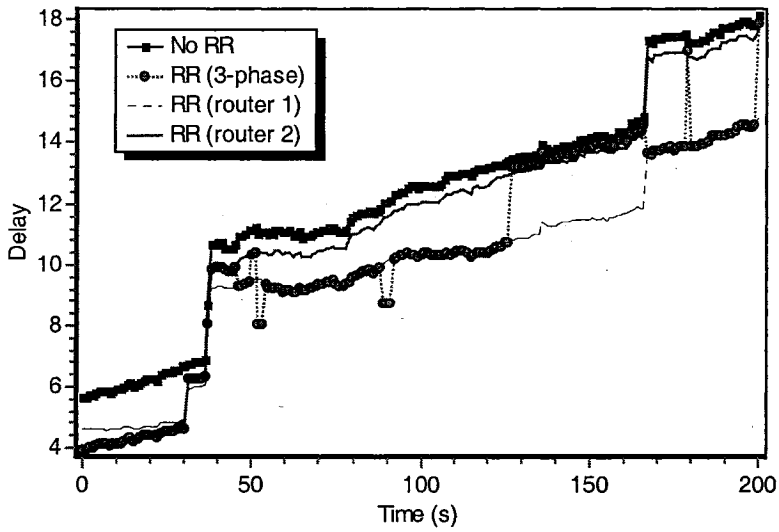


Fig. 6.37 The delay performance of dynamic networks for FEC/ARQ retransmissions.

λ_P=0.1, λ_J=0.5, λ_S=0.02, for FEC: n=12, k=10.

6.6.2.2 Delay

Fig. 6.36 and Fig. 6.37 show a sample of the simulation delay results. The 3-phase algorithm results do not seem to optimize the delay as they did for the bandwidth consumption. The use of RRs selected by the 3-phase algorithm cause the delay and the

bandwidth consumption to be improved significantly. In the future, we will optimize the delay performance.

6.7 Conclusions

In this chapter, we have simulated a reliable multicast protocol over the Internet, considering background traffic. Our results show that background traffic has a large influence on network performance. Background traffic may cause the overflow of router buffers and further results in multicast packet losses of intermediate nodes. This problem of intermediate losses has been addressed in the previous analysis.

We have investigated a performance comparison of pure ARQ and hybrid FEC/ARQ retransmissions. The advantages of the use of FEC are obvious to some extent, compared to the pure ARQ technique. FEC/ARQ has smaller delay than ARQ; however, if the packets are lost more frequently in the intermediate links, FEC/ARQ does not perform much better than ARQ.

In this chapter, we have simulated the tree-based multicast transmissions and compared the simulation results for the number of transmissions with the exact and approximate analysis results. They show a good consistency between the analysis and simulation results. We have also obtained the actual bandwidth consumption traversed by multicast transmissions. Based on the actual bandwidth consumption, the effect of topology on optimal RR placements has been investigated for QoS multicast. The multicast group with a

homogeneous topology should be partitioned to have the same size of each subgroup so that the whole group has the best performance. The same cannot be said for the topologies where each subgroup has very different topology. The delay performance has been addressed in this simulation. Buffer overflow causes delay to increase greatly and degrades the multicast performance.

We also simulated the 3-phase algorithm proposed in chapter 5 to find the optimal RR locations. Simulation results show that multicast performance benefits greatly from such a 3-phase policy. The delay performance may not be optimal while the bandwidth consumption is minimized; however, the use of RRs selected by the 3-phase algorithm causes the overall delay and bandwidth consumption to improve significantly.

Chapter 7 MULTICAST NETWORKS WITH HETEROGENEOUS LOSS PROBABILITY

7.1 Introduction

In previous chapters, we have focused on the performance analysis and simulations of multicast networks with homogeneous loss probability. Such assumption originates from the fact that members with the same loss probability are usually chosen to form a subgroup so that the multicast performance is uniform [76], [89]. A multicast with different loss probabilities is layered into different sessions according to loss probability [53], [76], [89], [99]. Homogeneous loss probability applies to these subgroups in this case. A real multicast network may have different loss probabilities that may result from the link loss or buffer overflow, e.g. wireless networks have higher link loss than wired networks. Even for layered subgroups, their loss probabilities may not be exactly the same due to buffer overflow. How multicast performance depends on loss characteristics of trees is an issue addressed in this chapter.

The performance of reliable multicast depends strongly on the loss characteristics of the tree, the topology, and the loss recovery mechanism [12], [69], [100]. The computation of the performance parameters such as bandwidth consumption and the number of transmissions is intensive for a general tree [12], [40]. However, reference [12] only provides analysis results and the optimal RR locations of one of the loss recovery policies

in this chapter in uniform loss and linear topologies. How to efficiently evaluate the performance of reliable multicast is significant for the design of loss recovery. Further, loss recovery should adapt to the topology and loss characteristics of a tree in order to greatly reduce the cost of multicast and the sender. The optimal RR locations can minimize the bandwidth consumption of networks that is important to the design of reliable multicast protocols. Unlike recent work [12], [40], [62], this chapter presents a more general approach to determining the optimal RR locations for different types of general topologies and addresses the two aspects of reliable multicast: the efficient evaluation of performance parameters and the determination of optimal RR locations based on different loss scenarios, topologies, and loss recovery schemes.

Motivated by the possible impact of RR locations on the performance of loss recovery mechanisms, we develop a new analysis and simulation of two new loss recovery policies. These are the main objectives of this chapter. However, in the process, we provide approximations that will illuminate the effects of control traffic, independent, correlated homogeneous and heterogeneous packet losses that were not addressed in recent work and previous chapters.

7.2 Contributions

This chapter considers two issues: performance analysis and the optimal RR locations, both of which are analyzed for two loss recovery schemes.

There is a lot of work on performance analysis of reliable multicast [12], [36]-[41], [62],

[71], [72], most of which are based on the recursive equations in [62]. The cost of the sender has been analyzed well by the number of transmissions E[M] defined before. Due to the intensive computation of E[M], approximations or simulations are often used [12]. We take a different approach to calculating these parameters after analyzing the transmission process of multicast packets. We discuss the effects of dependent and independent loss on networks and obtain the probability distribution of loss scenario over network links. Some basic characteristics are found. Further, we use them to derive E[M] and bandwidth consumption of different loss recovery schemes. By this approach, we can make some reasonable assumption and obtain an analytical evaluation of performance parameters for a multicast network with heterogeneous link loss probability.

In this chapter, we address the problem of finding the effects of loss recovery and loss characteristics on optimal RR locations. The optimal RR locations are suggested so as to minimize the bandwidth consumption for different loss characteristics of a tree and different topologies. An analytical comparison of two typical loss recovery schemes, namely, whole subgroup and selective retransmissions, are investigated. Moreover, in all analysis and simulations herein, we consider the effects of control traffic, independent, correlated, and heterogeneous losses.

7.3 Model

In Chapter 3, we discussed two loss recovery mechanisms and their performance characteristics in binary trees where only transmissions of data packets were considered.

Feedback implosion avoidance mechanism is important for reliable multicast. Multicast networks may not work due to feedback implosion. It is useful to analyze the performance of control packets in reliable multicast. Here, we still use two loss recovery mechanisms to avoid feedback implosion illustrated and explained in the following.

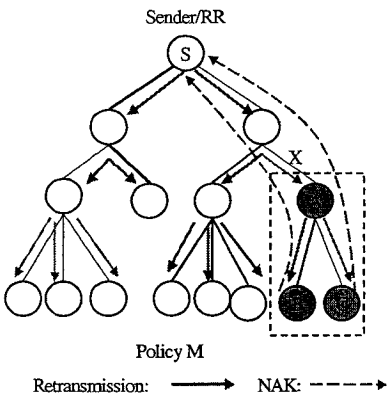


Fig. 7.1 The loss recovery of policy M. The loss occurs at link A.

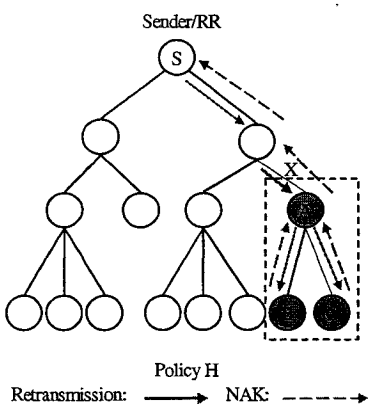


Fig. 7.2 The loss recovery of policy H. The loss occurs at link A.

Policy M – Whole subgroup retransmissions

Receivers missing packets send NAK upstream to the sender or RR to request retransmissions. The sender and/or RRs multicast the lost packet to all members in their

domain once they receive the retransmission requests (NAK) from receivers below [4]. Fig. 7.1 shows the loss recovery procedure of policy M where nodes B and C have not received a certain packet. This is a typical end-to-end scheme easily employed in reliable multicast. In this policy and as a worst case treatment, we assume all users suffering loss will transmit a NAK to sender or RR as will be reflected within the subsequent analysis. We do not assume here the presence of intelligent routers that may cut the NAK traffic and reduce NAK implosion [4].

Policy H – Selective retransmissions

When one receiver encounters a loss, it sends NAK to sender or RR depending on which is closer. The sender and/or RR then retransmit the packet only to these receivers that did not receive data in previous transmissions. This protocol needs the assistance of routers to recover the packet loss. Some simple functions are added for intermediate routers to remember which link downstream lost the packet and needs retransmissions. Routers merge all the NAK requests downstream and send only one NAK to the parent router to avoid NAK implosion [9]. Intermediate routers will retransmit packets only to some members of their subgroups. This policy adds a small extra capability to ordinary routers (and RRs). They should be able to route repair packets only to these nodes in their subgroup that have sent a NAK upstream through this router. But they would send the repair packets to all members of this subgroup if they themselves have lost the original packet. Fig. 7.2 shows the loss recovery of policy H. Also, nodes may randomize the

transmission times of their NAKs so as to further help the NAK implosion avoidance. These issues are not addressed in this chapter. However, in the analysis, we consider the reduction of the number of NAK transmissions on the total bandwidth consumption in this policy where the routers have more intelligence.

7.4 Performance Analysis

In previous chapters, we analyzed the performance of multicast networks with homogeneous loss probability. In this chapter, we extend the previous derivations to include other factors that affect the performance of multicast. This chapter analyzes multicast networks with heterogeneous loss probability that consider the independent, dependent losses, and control traffic. E[M], E[B], and E[C] defined before are valid in this chapter and reflect the bandwidth performance of multicast protocols. They depend on topologies, loss recovery policies, loss probabilities, loss characteristics, and the partitioning of trees. In the following sections, one first derives their expressions of no RR. Table 7.1 and Table 7.2 outline the notations used in this chapter.

7.4.1 Lemma

To introduce a lemma of this section, we start with a simple example for the loss probability p_k of link k and the number of links d_k from the sender to link k. Let H_k be a set that represents the links from the sender to node k. Let $\mathcal{M}_k$ be a set that represents the links under node k. We obtain the following equation for the topology shown in Fig.7.3:

$$\begin{aligned}
&\sum_{k=1}^{N} p_k \sum_{j\in H_k}^{j\neq k} d_j = p_2 d_1 + p_3 d_1 + p_4(d_1+d_3) + p_5(d_1+d_3) + p_6(d_1+d_3+d_5) \\
&+ p_7(d_1+d_3+d_5) + p_8(d_1+d_3) + p_{10} d_9 + p_{11} d_9 \\
&= d_1(p_2+p_3+p_4+p_5+p_6+p_7+p_8) + d_3(p_4+p_5+p_6+p_7+p_8) \\
&+ d_5(p_6+p_7) + d_9(p_{10}+p_{11}) = \sum_{k=1}^{N} d_k \sum_{j\in\mathcal{N}_k} p_j
\end{aligned} \tag{7.1}$$

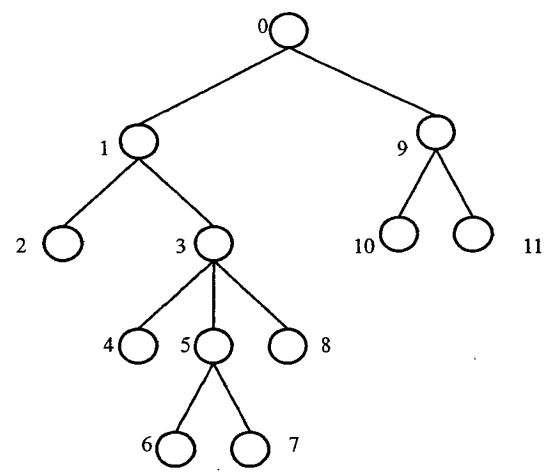


Fig.7.3 An example of tree topology

Lemma 1:

For an arbitrary tree, we define two functions ξ_k and φ_k affiliated with node k. For example, ξ_k or φ_k or both could be the number of links from sender to node k. In a different context, they could represent the number of links under node k, or the number of NAKs, or loss probability p_k, or any function of p_k, or just *1*. Many such combinations are possible scenarios, e.g. $\xi_k=n_k$ and $\varphi_k=p_k$, or $\xi_k= d_k$ and $\varphi_k=n_k$, or $\xi_k=1$ and $\varphi_k=p_k$, or $\xi_k=d_k$ and $\varphi_k=p_k$, or $\xi_k=d_k$ and $\varphi_k=d_k$, or $\xi_k=p_k$ and $\varphi_k=p_k$, or $\xi_k=d_k$ and $\varphi_k=1$, or $\xi_k= n_k$ and $\varphi_k=1$, and so on, to name but a few.

We claim that

$$\sum_{k=1}^{N} \xi_k \sum_{j \in H_k}^{j \neq k} \varphi_j = \sum_{k=1}^{N} \varphi_k \sum_{j \in \mathcal{O}\mathcal{N}_k} \xi_j \tag{7.2}$$

where H_k is a set of the links from the sender to node k, $\mathcal{O}\mathcal{N}_k$ is a set of the links under node k.

Proof:

Suppose that each node of a multicast tree has one counter whose initial value is 0. The k^{th} tick of the counter adds a value ($k=1, 2, \ldots, N$) $\xi_k \varphi_j$ as in the left hand of (7.2) to the counters of all nodes belonging to the path from the sender to the parent node of node k, i.e., the counter of node j will be incremented by $\xi_k \varphi_j$ if $j \in H_k, j \neq k$, the counters of other nodes are incremented by 0. The left hand side of (7.2) is sum of the final count of all counters when $k=N$. Noting that node j belongs to many paths, i.e., the paths from the sender to these nodes under node j, summing up all nodes (i.e, the first summation over k in the left hand of (7.2)) reveals that only the nodes under node j will affect the counter value of node j, i.e, the counter value of node j is $\sum_{k \in \mathcal{O}\mathcal{N}_j} \xi_k \varphi_j$. Thus the summations $\sum_{j=1}^{N} \sum_{k \in \mathcal{O}\mathcal{N}_j} \xi_k \varphi_j$ of all counters equal $\sum_{k=1}^{N} \xi_k \sum_{j \in H_k}^{j \neq k} \varphi_j$.

In addition, from (7.2), it follows that,

$$\sum_{k=1}^{N}\xi_k \sum_{j\in D_k}\varphi_j = \sum_{k=1}^{N}\xi_k \sum_{j\in H_k}^{j\neq k}\varphi_j + \sum_{k=1}^{N}\xi_k \sum_{j\in \overline{H_k}}\varphi_j + \sum_{k=1}^{N}\xi_k\varphi_k \tag{7.3}$$
$$= \sum_{k=1}^{N}\varphi_k \sum_{j\in \overline{H_k}}\xi_j + \sum_{k=1}^{N}\varphi_k \sum_{j\in H_k}^{j\neq k}\xi_j + \sum_{k=1}^{N}\xi_k\varphi_k = \sum_{k=1}^{N}\varphi_k \sum_{j\in D_k}\xi_j ,$$

$$\sum_{k=1}^{N}\xi_k \sum_{j\notin D_k}\varphi_j = \sum_{k=1}^{N}\xi_k \sum_{j\in N}\varphi_j - \sum_{k=1}^{N}\xi_k \sum_{j\in D_k}\varphi_j = \sum_{k=1}^{N}\varphi_k \sum_{j\notin D_k}\xi_j , \tag{7.4}$$

all of which have been used within the chapter.

In the same way as (7.2), we obtain the following equations:

$$\sum_{k}\sum_{j\notin D_k} f(j)Q(\Omega=\{k,j\}) = \sum_{k} f(k) \sum_{j\notin D_k} Q(\Omega=\{k,j\}) \tag{7.5}$$

$$\sum_{k\in N}\xi_k \sum_{l\in R_k}\varphi_l = \sum_{k\in R}\varphi_k \sum_{l\in H_k}\xi_l \tag{7.6}$$

where R_k is the set of receivers within the loss scenario $\Omega=\{k\}$, R is the set of all receivers in the multicast group. $Q(\Omega)$ is defined in the next section.

7.4.2 E[M]

As we said before, the computation of E[M] for a general topology is intensive [12], [40]. Thus, we try herein to find an approximate solution for the evaluation of the expected number of transmissions for general topology and heterogeneous loss probability.

Hierarchical reliable multicast is based on a tree topology where the loss of an intermediate link leads to the losses of all links below that link. Even if only one intermediate link loss takes place in a multicast tree, many links need retransmissions from the sender. In one multicast transmission, many losses may take place. Therefore, it is necessary to know how

many losses take place and how many links need to be involved in retransmissions because only corresponding nodes care whether they receive the correct packet in the next transmission.

Table 7.1. Notations of Multicast Networks with heterogeneous loss probability

ID	Meaning
p_k	Packet loss probability over link k.
d_k	The number of links from the sender to node k, also the set of these links.
n_k	The total number of links under node k, also the set of these links.
D_k	The total number of links involved in retransmissions if link k loses a packet. Also the set of these links.
H_k	The set of the links from the sender to node k.
$\mathcal{N}_k$	The set of the links under node k.
Ω	Link loss scenario, i.e. the set of independent link losses. For example, {1,2} means the losses from links 1 and 2, {0} means no loss.
$D(\Omega)$	The number of links involved in retransmissions for multiple loss scenarios Ω, also the set of these links.
$n(\Omega)$	The number of links under the links of the multiple loss scenarios Ω.
$\mathcal{N}(\Omega)$	The set of the links under the links of the multiple loss scenarios Ω.
M, m	The total number of packet transmission times per source packet (due to

	losses).
C	The total bandwidth consumed in a multicast group (the successful delivery of one source packet to all nodes including all retransmission).
$P(M=m)$	The probability that one packet multicast is successful in the m^{th} transmission (all nodes get the same packet).
$P(m\|\Omega)$	The probability that multicast is successful in the m^{th} transmission after the loss scenarios Ω take place.
$Q_N(\Omega)$	The probability that multiple loss scenarios Ω take place in one transmission over N links.
$\Lambda(\Omega)$	Bandwidth consumed by NAKs for multiple loss scenarios Ω.
Λ_k	Bandwidth consumed by NAKs if link k loses a packet.
β	The ratio of a NAK packet to a data packet.

Multicast packet losses are not independent in a general tree. In Fig.7.4, the packet loss of node A will result in the losses of all links below node A. These losses are dependent (correlated) because they involve the nodes of subtree rooted at node A. However, if one node is not under the subtree of another node, the losses such as node A and B are independent (uncorrelated). They do not affect each other. One may consider the effect of one link loss on other nodes of its subtree. Suppose that each node has a sequence number and the sender has the sequence number 0, shown in Fig.7.4. For a given packet transmission, if node k (e.g., node A) does not receive the packet, n_k nodes will also lose the

packet resulting in a correlated loss where n_k is the total number of links below node k. The sender retransmits the repair packet to these nodes of $D_k = d_k + n_k$ where d_k is the number of links from the sender to node k. Therefore, we only investigate if these nodes receive the packet correctly in the next retransmission. D_k reflects the effect of loss on link k on dependent losses of the network tree.

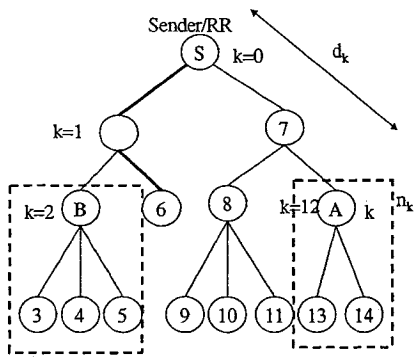


Fig.7.4 The links involved in retransmissions. $D_k=d_k+n_k$.

In addition to dependent (correlated) loss, we also analyze the effects of independent loss. In a multicast transmission, some independent losses may take place over different links due, for example, to other session or unicast traffic. We use Ω to represent such independent loss scenario, e.g. $\Omega=\{2,8\}$ means the packet losses from link 2 and 8 in Fig.3.7, $\Omega=\{0\}$ means no loss. Here, we denote by $Q_N(\Omega)$ the probability that multiple loss scenario Ω takes place in a multicast transmission over N links of a certain part of multicast tree. $Q_N(\Omega)$ has the following probability distribution.

$$Q_N(\Omega=\{0\})=\prod_{k\in N}(1-p_k) \tag{7.7}$$

$$Q_N(\Omega=\{k_1\}) = p_{k_1} \prod_{k\neq k_1, k\notin \mathcal{N}_{k_1}}^{k\in N} (1-p_k) \tag{7.8}$$

$$Q_N(\Omega=\{k_1,k_2,\cdots,k_j\}) = \prod_{l=1}^{j} p_{k_l} \prod_{k\notin\Omega}^{k\notin \mathcal{N}(\Omega)} (1-p_k), \tag{7.9}$$
$$k_1\in N, k_2\notin D_{k_1},\cdots,k_j\notin D(\Omega=\{k_1,k_2,\cdots,k_{j-1}\})$$

where p_k is the packet loss probability over link k, $\mathcal{N}_{k_1}$ is the set of links below node k_1, $\mathcal{N}(\Omega)$ is the set of links missing the packet under the multiple loss scenarios Ω, i.e. $\mathcal{N}(\Omega=\{k_1,k_2,\cdots,k_j\})=\mathcal{N}_{k_1}\cup\mathcal{N}_{k_2}\cup\cdots\cup\mathcal{N}_{k_j}$. N is the total number of links in the network or a part of network under the error control of a RR. D_k was defined before and the exclusion of $k_2\notin D_{k_1}$, for example, is to include only independent, not dependent loss. $D(\Omega)$ is the number of links involved in retransmissions for multiple loss scenarios Ω and represents the effect of Ω on dependent losses. For example, $D(\Omega=\{2,12\})$ represents the links of d_2, n_2, d_{12}, and n_{12}. Please refer to Table 7.1 for these notations. The following can be obtained from (7.7)-(7.9).

$$\sum_{\Omega\subset N} Q_N(\Omega) = 1 \tag{7.10}$$

$$Q_N(\Omega=\{k_1,k_2,\cdots,k_j\}) = \eta_{k_1} p_{k_1} Q_N(\Omega=\{k_2,\cdots,k_j\}\,|\,k_1) \tag{7.11}$$

where $\sum_{\Omega\subset N}$ stands for the summation of all possible multiple loss scenarios for N links, i.e..

$$\sum_{\Omega\subset N} = \sum_{k\in N} + \frac{1}{2!}\sum_{k_1\in N}\sum_{k_2\notin D_{k_1}} + \frac{1}{3!}\sum_{k_1\in N}\sum_{k_2\notin D_{k_1}}\sum_{k_3\notin D(\{k_1,k_2\})} + \cdots \tag{7.12}$$

$$\eta_k = \prod_{l\in H_k, l\neq k}(1-p_l) \tag{7.13}$$

$Q_N(\Omega=\{k_2,\cdots,k_j\}\,|\,k_1)$ is the probability distribution of independent multiple loss scenarios Ω not considering the links of D_{k_1} after packet loss takes place at link k_1, i.e.

$$Q_N(\Omega=\{k_2,k_3,\cdots,k_j\}\,|\,k_1)=\prod_{l=2}^{j}p_{k_l}\prod_{k\notin\Omega,.k\in H_{k_1}}^{k\notin\Lambda^{'}(\Omega)}(1-p_k), \tag{7.14}$$
$$k_1\in N,k_2\notin D_{k_1},\cdots,k_j\notin D(\{k_1,k_2,\cdots,k_{j-1}\})$$

Theorem 1.

Any function $f(\Omega)$ of multiple independent loss scenario Ω, which could be defined as the number of links $n(\Omega)$ as in Table 7.1, or the number of independent losses, or the number of NAKs resulting from occurrence of independent errors over links k_1, k_2,... etc as will follow, can be decomposed into the functional sum of different single independent losses, i.e.

$$f(\Omega=\{k_1,k_2,\cdots,k_j\})=f(\{k_1\})+f(\{k_2\})+\cdots+f(\{k_j\}) \tag{7.15}$$

In the case of no loss, i.e. $\Omega=\{0\}$, their values are 0, i.e.

$$f(\Omega=\{0\})=0 \tag{7.16}$$

Further, one can obtain its average value over different loss scenarios, i.e.

$$\overline{f_N}=\sum_{\Omega\subset N}f(\Omega)\cdot Q_N(\Omega)=\sum_{k}p_k\eta_k f(\{k\}) \tag{7.17}$$

where the summation $\sum_{\Omega\subset N}$ means all possible multiple loss scenarios whose probabilities are in (7.7) and (7.9).

Proof:

Equation (7.15) is made evident by the independence of the packet losses on link k_1, k_2, ... For example, if $f(\Omega)$ is the number of NAKs generated from losses situated on link k_1, this is completely independent and adds up to the number of NAKs generated from loss on link k_2, and so on, to provide the function of sum of these events.

$$\overline{f_N} = \sum_{\Omega \subset N} f(\Omega) \cdot Q_N(\Omega) = f(\{0\}) Q_N(\{0\}) + \sum_{k \in N} f(\{k\}) \cdot Q_N(\{k\}) \tag{7.18}$$

$$+\frac{1}{2} \sum_{k_1} \sum_{k_2 \notin D_{k_1}} [f(\{k_1\}) + f(\{k_2\})] Q_N(\Omega = \{k_1, k_2\}) +$$

$$\frac{1}{3!} \sum_{k_1} \sum_{k_2 \notin D_{k_1}} \sum_{k_3 \notin D(\{k_1, k_2\})} [f(\{k_1\}) + f(\{k_2\}) + f(k_3)] Q_N(\{k_1, k_2, k_3\}) + \cdots$$

$$\overline{f_N} = f(\{0\}) Q_N(\{0\}) + \sum_k f(k) p_k \eta_k \cdot Q_N(\{0\} \mid k)$$

$$+ \sum_{k_1} f(\{k_1\}) p_{k_1} \eta_{k_1} \sum_{k_2 \notin D_{k_1}} Q_N(\{k_2\} \mid k_1)$$

$$+ \sum_{k_1} f(\{k_1\}) p_{k_1} \eta_{k_1} \frac{1}{2!} \sum_{k_2 \notin D_{k_1}} \sum_{k_3 \notin D(\{k_1, k_2\})} Q_N(\Omega = \{k_2, k_3\} \mid k_1) + \cdots$$

$$= f(\{0\}) Q_N(\{0\}) + \sum_{k_1} f(\{k_1\}) p_{k_1} \eta_{k_1} [Q_N(\{0\} \mid k_1)$$

$$+ \sum_{k_2 \notin D_{k_1}} Q_N(\{k_2\} \mid k_1) + \frac{1}{2!} \sum_{k_2 \notin D_{k_1}} \sum_{k_3 \notin D(\{k_1, k_2\})} Q_N(\Omega = \{k_2, k_3\} \mid k_1) + \cdots]$$

$$= f(\{0\}) Q_N(\{0\}) + \sum_k p_k \eta_k f(\{k\})$$

For $f(\Omega=\{0\})=0$, we have $\overline{f_N} = \sum_k p_k \eta_k f(\{k\})$.

The probability density function P(M=m) representing the number of the final successful transmission trials M can be obtained recursively, i.e.

$$P(M=1) = Q_N(\Omega = \{0\}) = \prod_{k \in N} (1 - p_k) \tag{7.19}$$

$$P(M=m) = \sum_{\Omega \neq \{0\}}^{N} Q_N(\Omega) P(M = m-1 \mid \Omega), \qquad m = 2,3,..., \tag{7.20}$$

where $\sum_{\Omega}$ stands for all possible link loss scenarios. The conditional probability $P(M=m-1|\Omega)$ is the probability that m-1 retransmissions are required to recover multiple loss scenario Ω. It can be recursively calculated from (7.19) and (7.20). However, only $D(\Omega)$ not N nodes are involved in retransmissions if loss scenario Ω takes place where $D(\Omega)$ is the subset of links involved in retransmissions for the loss scenario Ω.

Thus one can easily obtain the expected number of total transmissions, which can be

recursively calculated.

$$E[M] = \sum_{m=1}^{\infty} mP(M=m) = P(M=1) + \sum_{m=2}^{\infty} m \sum_{\Omega \neq \{0\}}^{N} Q_N(\Omega) P(M=m-1 \mid \Omega) \tag{7.21}$$
$$= P(M=1) + \sum_{\Omega \neq \{0\}}^{N} Q_N(\Omega)[1 + \sum_{m=2}^{\infty}(m-1)P(M=m-1 \mid \Omega)]$$
$$= 1 + \sum_{\Omega \neq \{0\}}^{N} Q_N(\Omega) E[M \mid \Omega]$$

where E[M|Ω] is the expected number of transmissions to correct the multiple loss scenario Ω. Only these nodes that did not receive the packet are involved in retransmissions. So E[M|Ω] can also be similarly obtained.

$$E[M \mid \Omega] = 1 + \sum_{\Omega_1 \neq \{0\}}^{D(\Omega)} Q_{D(\Omega)}(\Omega_1) E[M \mid \Omega_1] \tag{7.22}$$

where Ω_1 is the set of links incurring losses after second transmissions.

Substituting (7.22) into (7.21), one may obtain the following approximation for E[M].

$$E[M] = 2 - Q_N(\{0\}) + \sum_{\Omega \neq \{0\}}^{N} Q_N(\Omega) \sum_{\Omega_1}^{D(\Omega)} Q_{D(\Omega)}(\Omega_1) E[M \mid \Omega_1] \tag{7.23}$$

Repeating the same procedure, i.e. $E[M \mid \Omega_1] = 1 + \sum_{\Omega_2 \neq \{0\}}^{D(\Omega_1)} Q_{D(\Omega_1)}(\Omega_2) E[M \mid \Omega_2]$ and we obtain

$$E[M] = 2 - Q_N(\{0\}) + \sum_{\Omega \neq \{0\}}^{N} Q_N(\Omega)[1 - Q_{D(\Omega)}(\Omega_1 = \{0\})] \tag{7.24}$$
$$+ \sum_{\Omega \neq \{0\}}^{N} Q_N(\Omega) \sum_{\Omega_1 \neq \{0\}}^{D(\Omega)} Q_{D(\Omega)}(\Omega_1) \sum_{\Omega_2 \neq \{0\}}^{D(\Omega_1)} Q_{D(\Omega_1)}(\Omega_2) E[M \mid \Omega_2]$$

Taking $Q(\Omega_1 = \{0\}) = \prod_{k=1}^{D(\Omega)}(1-p_k) \approx 1 - \sum_{k=1}^{D(\Omega)} p_k$ for low p_k and ignoring the 4th term on the right hand side of (7.24) relative to the first few terms, one obtains the following approximation.

$$E[M] \approx 2 - Q_N(\{0\}) + \sum_{l=1}^{N} p_l \eta_l \sum_{k}^{D_l} p_k \tag{7.25}$$

where $Q_N(\Omega = \{0\}) = \prod_{k \in N}(1-p_k)$ depends on link loss probability and is independent of

topology. $\sum_{l=1}^{N} p_l \eta_l \sum_{k}^{D_l} p_k$ is a term dependent on the topology. Assume $q = \sum_{l=1}^{N} p_l$. For a small value of q, one obtains the following approximation.

$$E[M] \approx 1 + q - \frac{1}{2}q^2 + \frac{1}{6}q^3 + \sum_{l=1}^{N} p_l \eta_l \sum_{k}^{D_l} p_k \tag{7.26}$$

7.4.3 The Evaluation of the Bandwidth Consumption

Although E[M] gives the expected number of transmissions, the number of links affected by each transmission is different. The actual bandwidth consumption of networks depends not only on E[M] but also on the number of links traversed by each transmission. Thus, it is necessary to evaluate the bandwidth consumed on all links until final success. By final success, we mean to include all link transmissions used until all nodes receive the packet correctly. The first original transmission always affects the whole multicast nodes regardless of which loss recovery mechanism is used. If node k does not receive the packet, n_k links will lose the packet as shown in Fig.3.7. Then the actual bandwidth consumption is $N\text{-}n(\Omega)$ for multiple loss scenarios Ω where $n(\Omega)$ is the number of dependent losses for multiple loss scenarios Ω. Thus, we obtain the expected bandwidth consumption over N links as follows,

$$E[C] = \sum_{\Omega \subset N} Q_N(\Omega)\{N - n(\Omega) + \beta\Lambda(\Omega) + E[C \mid \Omega]\} \tag{7.27}$$

where $Q_N(\Omega)$ is the probability of having multiple loss scenarios Ω in a transmission trial as defined in Table 7.1, and E[C|Ω] is the extra bandwidth consumed in all retransmission trials to correct the loss scenarios Ω that take place in the first transmission. $\Lambda(\Omega)$ is the

bandwidth consumed by NAKs for the loss scenario Ω. β is the ratio of NAK packet size to regular data packet size.

Due to $n(\Omega=\{k_1,k_2,\cdots,k_j\})=n_{k_1}+n_{k_2}+\cdots+n_{k_j}$ and $n(\Omega=\{0\})=0$, one has the following from (7.17).

$$\sum_{\Omega\subset N} Q_N(\Omega)n(\Omega)=\sum_{k\in N} p_k\eta_k n_k=\sum_{k\in N} p_k\eta_k \sum_{l\in\mathcal{N}_k} 1 \tag{7.28}$$

Using (7.13) and (7.2) where we take $\varphi_k=p_k\eta_k$ and $\xi_k=1$, one obtains

$$\sum_{\Omega\subset N} Q_N(\Omega)n(\Omega)=\sum_{k\in N}\sum_{l\in H_k, l\neq k} p_l\eta_l=\sum_{k\in N}(1-\eta_k) \tag{7.29}$$

Substituting (7.29) into (7.27), one has

$$E[C]=\sum_{k\in N}\eta_k+\sum_{\Omega\subset N} Q_N(\Omega)\{\beta\Lambda(\Omega)+E[C\,|\,\Omega]\} \tag{7.30}$$

Equation (7.30) suggests that the bandwidth consumption of original data packets in the first transmission is $\sum_{k\in N}\eta_k$. In order to evaluate $\Lambda(\Omega)$ and the expected conditional bandwidth consumption of retransmissions, i.e. E[C|Ω], we encounter two cases depending on the recovery methodology, i.e., policy M and H to follow.

7.4.3.1 Policy M

Due to the lack of router assistance, each receiver losing a packet unicasts a NAK packet to the sender/RR, which costs a bandwidth d_k for receiver k. All receivers within the loss scenario Ω send NAK upstream, i.e. $\Lambda(\Omega=\{k\})=\sum_{l\in R_k} d_l$ where R_k is the set of receivers within the loss scenario $\Omega=\{k\}$. Thus, we have the following result using (7.6) with $\varphi_k=$

$p_k\eta_k$ and $\xi_k=d_k$,

$$\sum_{\Omega \subset N} Q_N(\Omega)\Lambda(\Omega) = \sum_{k\in N} p_k\eta_k \sum_{l\in R_k} d_l = \sum_{k\in R} d_k \sum_{l\in H_k} p_l\eta_l$$
$$= \sum_{k\in R} d_k[1-\eta_k(1-p_k)] = \sum_{k\in R} d_k(1-\eta_k+p_k\eta_k) \tag{7.31}$$

where $\sum_{l\in R_k} d_l$ is the sum of affected links between the sender or RR and the location of the k^{th} loss since the corresponding NAKs sent upstream transverse these links. The derivation of (7.31) follows steps similar to these leading to (7.29).

In this loss recovery scheme, the packet is always multicast to the whole node population by the sender or RR, i.e., each transmission or retransmission affects all members of the same subgroup. The actual bandwidth consumption of each retransmission is $\sum_{k\in N}\eta_k$ (does not change in policy M) as can be seen by repeated application of (7.30) to each retransmission trial. Thus, noting we have E[M|Ω] transmission trials and each has a corresponding transmission cost of $\sum_{k\in N}\eta_k$ plus cost of E[M|Ω]-1 NAK messages, one obtains the following equation:

$$E[C\,|\,\Omega] = E[M\,|\,\Omega]\cdot\sum_{k\in N}\eta_k + \beta(E[M\,|\,\Omega]-1)\sum_{k\in R} d_k(1-\eta_k+p_k\eta_k) \tag{7.32}$$

where E[M|Ω] is the expected number of retransmissions after multiple loss scenarios Ω take place, the first term is bandwidth consumption of data packets in retransmissions and the 2nd term accounts for the cost of NAKs in retransmission trials.

By substituting (7.32) into (7.30) and using (7.21), one may get E[C].

$$E[C] = E[M]\cdot\sum_{k\in N}\eta_k + \beta(E[M]-1)\sum_{k\in R} d_k(1-\eta_k+p_k\eta_k) \tag{7.33}$$

7.4.3.2 Policy H

Policy H uses routers to recover losses. Receivers losing a packet send a NAK to the upstream router, and routers merge all received NAKs and send only one NAK upstream. Thus the bandwidth cost of NAKs is $\Lambda(\Omega) = D(\Omega) \le \sum_k D_k$ for the loss scenario Ω. The upper bound of NAK cost is obtained similar to (7.31),

$$\sum_{\Omega \subset N} Q_N(\Omega)\Lambda(\Omega) = \sum_{k \in N} p_k \eta_k D_k \tag{7.34}$$

For each retransmission, the sender and/or RRs will retransmit the packet to only these receivers that did not receive data in previous transmissions, e.g. if link k loses a packet, only D_k links are involved in retransmissions. If a loss scenario Ω with multiple independent losses takes place, one can sum up the bandwidth of single independent loss, i.e. $E[C \mid \Omega] = \sum_k E[C \mid k]$. We can evaluate E[C|k] in the following equation for both data and NAKs transmissions.

$$E[C \mid k] \le E[M \mid k] \sum_{l \in D_k} \eta_l + \beta(E[M \mid k] - 1)D_k \tag{7.35}$$

The upper bound stems from the fact that only the first retransmission will lead to loss$\in D_k$ and further retransmissions are expected to handle less loss. Considering (7.17) and (7.35) with $f(\{k\}) = E[C|k]$, one obtains the bandwidth consumption of retransmissions for policy H as follows:

$$\begin{aligned} &\sum_{\Omega} Q_N(\Omega) E[C \mid \Omega] = \sum_k p_k \eta_k E[C \mid k] \\ &\le \sum_k p_k \eta_k \{E[M \mid k] \sum_{l \in D_k} \eta_l + \beta(E[M \mid k] - 1)D_k\} \end{aligned} \tag{7.36}$$

One may estimate the bandwidth consumed in protocol H by substituting (7.34) and (7.36) into (7.30).

$$E[C] = \sum_{k\in N}\eta_k + \beta\sum_{k\in N} p_k\eta_k D_k + \sum_k p_k\eta_k\{E[M\mid k]\sum_{D_k}\eta_i + \beta(E[M\mid k]-1)D_k\} \qquad (7.37)$$
$$= \sum_{k\in N}\eta_k + \sum_k p_k\eta_k E[M\mid k]\cdot\sum_{l\in D_k}(\eta_l+\beta)$$

7.5 Effect of RRs on Bandwidth

Hierarchical loss recovery is often used to provide the scalability in which case some special receivers or routers such as RR have the FEC/ARQ capacity and retransmit the repair packet to reduce the load of the sender. Each RR is the representative of a multicast group and sends NAK upstream to the sender on behalf of receivers downstream, e.g. RR_1 is responsible for the retransmissions of N_1 links in Fig. 7.5 and reports the state to the sender. Here we denote by N_i the number of links covered by the RR_i. N_0 is the number of links not covered by any RRs. Fig. 7.5 is an example of a multicast group having 2 RRs.

We consider first the case of 1 RR. For the multiple loss scenarios Ω in the first transmission, these losses occur within, i.e. Ω_1 or outside, i.e. Ω_0, the RR coverage. We denote by $Q_{N_1}(\Omega_1)$ ($Q_{N_0}(\Omega_0)$) the probability that multiple loss scenarios Ω_1 (Ω_0) take place within (outside) RR coverage. Ω_1 and Ω_0 are two independent events occurring in different loss recovery domains. Thus, the probability $Q_N(\Omega_0,\Omega_1)$ that the loss scenarios Ω_0 and Ω_1 take place over N links is given by the following:

$$Q_N(\Omega_0,\Omega_1) = Q_{N_0}(\Omega_0)Q_{N_1}(\Omega_1) \qquad (7.38)$$

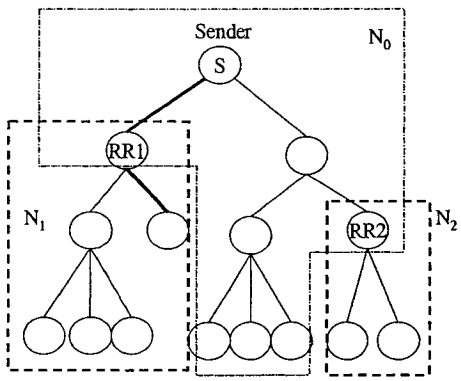


Fig. 7.5 The partition of a multicast group

Table 7.2. Notations for Multicast Groups

ID	Meaning
N	The total number of links within a multicast group, also the set of these links.
N_i	The number of links covered by RR_i, also the set of these links.
C	The total bandwidth consumed in the multicast group N (the successful delivery of one source packet to all nodes including all retransmission.)
B	Average bandwidth consumption per link for the multicast group.
$E[M_i]$	The expected number of transmissions for N_i links to receive a correct packet.
$E[C_i]$	The total expected bandwidth consumed in N_i links over all transmissions.

The total bandwidth consumed in N links can be obtained recursively

$$E[C]=\sum_{\Omega_0 \subset N_0}\sum_{\Omega_1 \subset N_1} Q_N(\Omega_0,\Omega_1)\{N_0-n(\Omega_0)+N_1-n(\Omega_1)+\beta\Lambda(\Omega_0)+\beta\Lambda(\Omega_1)+E[C\,|\,(\Omega_0,\Omega_1)]\} \tag{7.39}$$

where N_0 and N_1 are the number of links outside and within RR coverage, respectively.

$\Lambda(\Omega_0)$ *(or* $\Lambda(\Omega_1)$*)* is the bandwidth consumed by NAKs for the loss scenario Ω_0 (or Ω_1). β is the ratio of NAK packet size to regular data packet size. $E[C|(\Omega_0,\Omega_1)]$ is the extra bandwidth consumption when multiple loss scenarios Ω_0 and Ω_1 take place. These losses are recovered by either the RR or the sender, depending on their locations. RR recovers only the losses under this RR while the sender recovers the losses outside all RRs. The sender and the RR work independently in the next retransmissions. Thus, the total bandwidth $E[C|(\Omega_0,\Omega_1)]$ can be divided into the summation of bandwidth $E[C|\ \Omega_0]$ and $E[C|\Omega_1]$ for two subgroups:

$$E[C\,|\,\Omega_0,\Omega_1] = E[C\,|\,\Omega_0] + E[C\,|\,\Omega_1] \tag{7.40}$$

where $E[C|\Omega_1]$ ($E[C|\Omega_0]$) is the total conditional bandwidth consumption to recover the loss scenario Ω_1 (Ω_0).

One obtains the following results for the total bandwidth consumption by substituting equation (7.40) into (7.39) and using (7.30):

$$\begin{aligned} E[C] &= \sum_{\Omega_0 \subset N_0} Q_{N_0}(\Omega_0) \sum_{\Omega_1 \subset N_1} Q_{N_1}(\Omega_1)\{N_0 - n(\Omega_0) + \beta\Lambda(\Omega_0) + E[C\,|\,\Omega_0]\} \\ &+ \sum_{\Omega_0 \subset N_0} Q_{N_0}(\Omega_0) \sum_{\Omega_1 \subset N_1} Q_{N_1}(\Omega_1)\{N_1 - n(\Omega_1) + \beta\Lambda(\Omega_1) + E[C\,|\,\Omega_1]\} \\ &= E[C_0] + E[C_1] \end{aligned} \tag{7.41}$$

where $E[C_k]$ ($k=0,1$) is the expected bandwidth consumed in N_k links and can be calculated by (7.33) or (7.37) depending on the loss recovery policy. The above equation shows that the total bandwidth consumed in the whole group is equal to the summation of bandwidth consumed in every subgroup.

Based on the discussion above, the total bandwidth is equal to the summation of bandwidth consumed in N_0 and N_1 links, i.e., $E[C_0]$ and $E[C_1]$. In the same way, we can extend this result to the case of r RRs. The total bandwidth consumption is the summation of bandwidth consumed by all subgroups.

$$E[C] = \sum_{i=0}^{r} E[C_i] \tag{7.42}$$

where $E[C_i]$ is the expected bandwidth consumed in N_i links handled by the i^{th} RR. The optimal RR locations make the network consume the minimal bandwidth consumption E[C].

As previously discussed, the evaluation of bandwidth $E[C_i]$ depends on the policy of loss recovery. Similarly, the following discusses two policies of repair packet retransmissions.

7.5.1 Policy M

The sender or RRs multicast repair packets to the subgroup after losses take place. We assume that each subgroup is independent of each other. The losses within the RR_1 have influence only on N_1 links. The losses outside the RR_1 do not have influence on the bandwidth consumption of N_1 links but they affect N_0 links. Therefore, one may obtain the following bandwidth consumption of N_0 (or N_1) links from (7.33).

$$E[C_i] = E[M_i] \cdot \sum_{k \in N_i} \eta_k + \beta(E[M_i] - 1) \sum_{k \in R_i} d_k (1 - \eta_k) \tag{7.43}$$

where $E[M_i]$ is the expected number of transmissions for N_i links and can be easily

evaluated from (7.26) ($i=0,1,\ldots,r$).

7.5.2 Policy H

The sender or RRs retransmit the repair packet to only those receivers that did not receive the packet. We can estimate the bandwidth consumption of N_i links ($i=0, 1, \ldots, r$) using (7.37).

$$E[C_i] = \sum_{k \in N_i} \eta_k + \sum_{k \in N_i} p_k \eta_k E[M \mid k] \cdot \sum_{l \in D_k} (\eta_l + \beta) \tag{7.44}$$

7.6 Results and Discussions

7.6.1 The Number of Transmissions

We use an example of a general topology in Fig. 7.6 to compare the exact equations and the approximate equation (7.26) with given heterogeneous link loss probability. First, we compare one special case where all router links have the same loss probability p_router while all receiver links have the same loss probability p_receiver. The approximate results with different p_router and p_receiver in Fig. 7.7 show that they match well to the exact results.

We then compare exact and approximate E[M] where the various loss probability are given in Fig. 7.6. However, in Fig. 7.8, we increment all heterogeneous loss probabilities in Fig. 7.6 by the same variable amount delta p. In addition, two link losses p_a and p_b are varied as

per Table 7.3 leading to the HL0, HL1, HL2 cases. The exact and approximate results in Fig. 7.8 match closely.

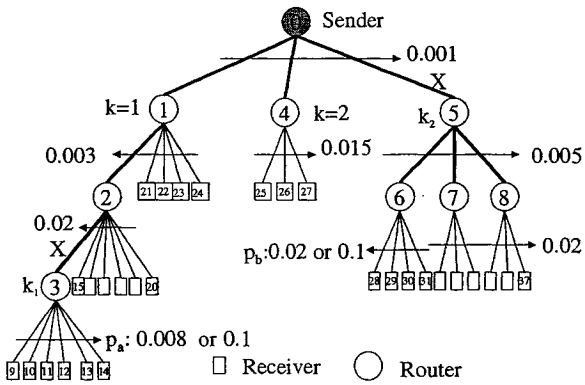


Fig. 7.6 One example of a general multicast network. Table 7.3 provides the values of p_a and p_b for the different scenarios HL0, HL1 and HL2.

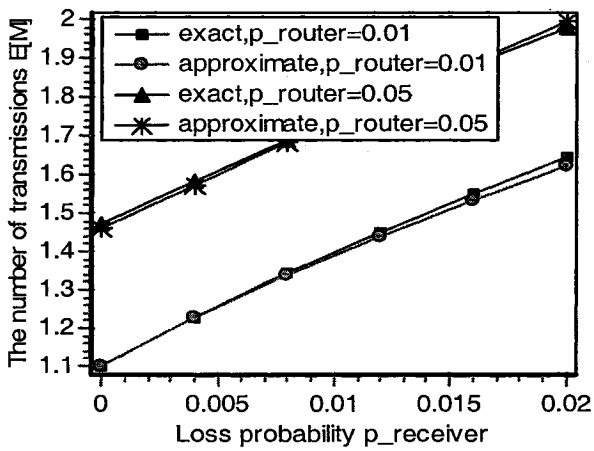


Fig. 7.7 Comparison of E[M] between exact and approximate solutions

In Fig. 7.9, We compare E[M] values of 3 different topologies for a uniformly distributed random link loss probability in the range *{0, 0.01}*. These are linear, star and full binary trees. We take *N* and all loss probability to be the same for all these topologies under all

random loss scenarios. Fig. 7.9 shows that even if these topologies have the same value of *N*, E[M] differs greatly depending on the type of topology.

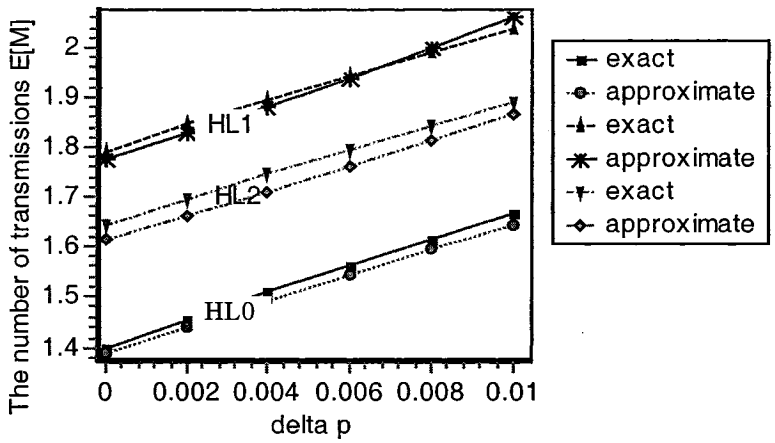


Fig. 7.8 Comparison of exact and approximate E[M] for a multicast network with heterogeneous loss probability where the loss probability of each link is the summation of the value of the link in Fig. 7.6 and delta p.

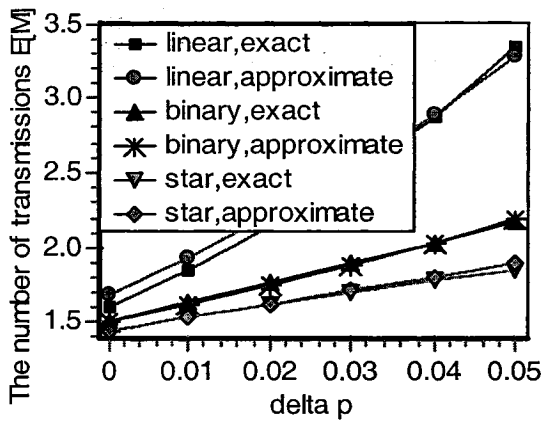


Fig. 7.9 Comparison of E[M] for different topologies. *N=14*.

7.6.2 Performance of Multicast Networks with Heterogeneous Loss Probability

In this section, we discuss the optimal RR locations of the multicast network with

non-uniform link loss probability, i.e. each link may have different packet loss probability. In this case, we also simulate the multicast transmissions with two loss recovery schemes and compare the analysis and simulation results as shown in Fig.7.10 and Fig.7.11. In the simulation, we average the number of transmissions and bandwidth consumption over 100000 packets. This was found enough to provide us with a confidence interval of 95% about the true mean of each result in all simulation figures. These results show that the analysis and simulations give the same optimal RR locations for both loss recovery schemes. Simulation results prove that bandwidth consumption can be evaluated by our analysis equations.

Table 7.3 Packet loss probability and the optimal RR locations of Fig. 7.6 for different scenarios shown in Fig.7.10 and Fig.7.11.

Scenario		HL0	HL1	HL2
Loss probability p_a		0.008	0.1	0.008
Loss probability p_b		0.02	0.02	0.1
The Optimal RR Location	Policy M	1	2	5
The Optimal RR Location	Policy H	2	3	6

We assume different packet loss probability for some links of the network in Fig. 7.6. One finds from Fig.7.10 and Fig.7.11 that the optimal RR location depends on the policy and loss scenario. RR should be located closer to the links of high packet loss probability for both loss recovery schemes. The RR locations depend on the distribution of packet loss

probability over different links. The high loss links frequently request retransmissions. In order to reduce the effects of such retransmissions on other nodes, the RR moves to the high loss links to support efficient loss recovery, which applies to the two schemes. The two policies yield similar RR locations.

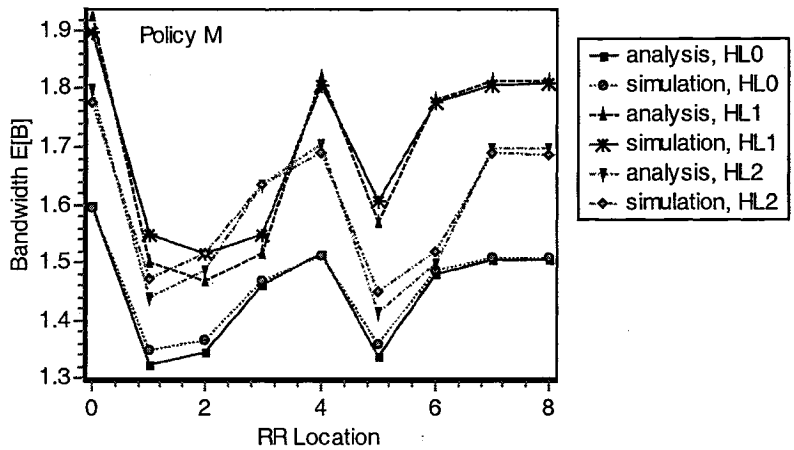


Fig.7.10 The bandwidth E[B] of heterogeneous packet loss probability in Fig. 7.6 for policy M

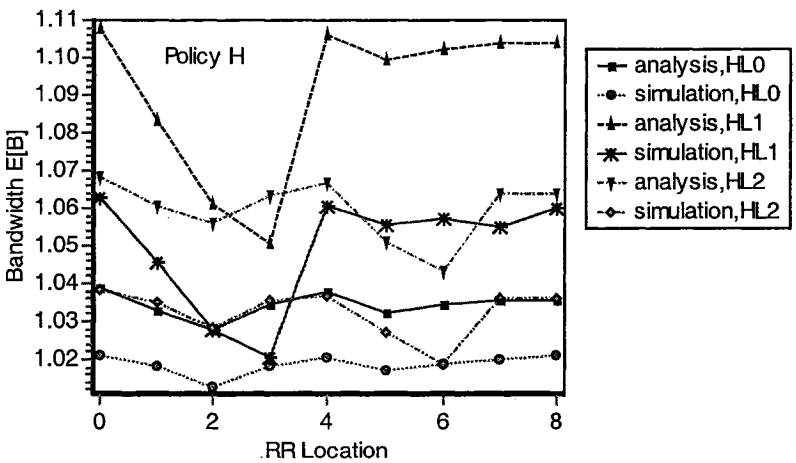


Fig.7.11 The bandwidth E[B] of heterogeneous packet loss probability in Fig. 7.6 for policy H

Fig.7.12 gives the effects of NAK overhead on the bandwidth consumption of multicast networks. NAK packets are usually much smaller than data packets and require less bandwidth consumption. Due to the existence of RRs, NAK packets are noticeably reduced for policy M and are minimized for policy H.

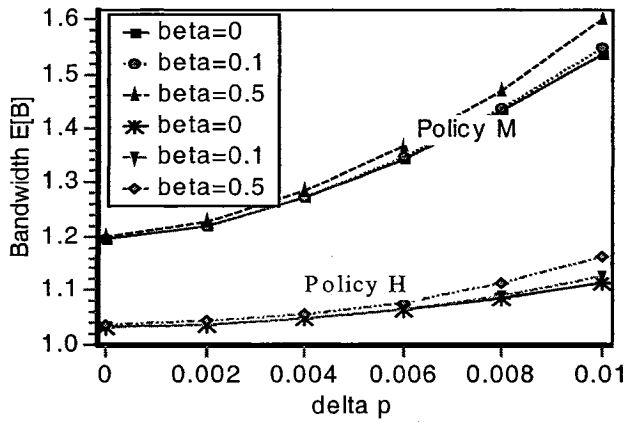


Fig.7.12 The effect of NAKs.

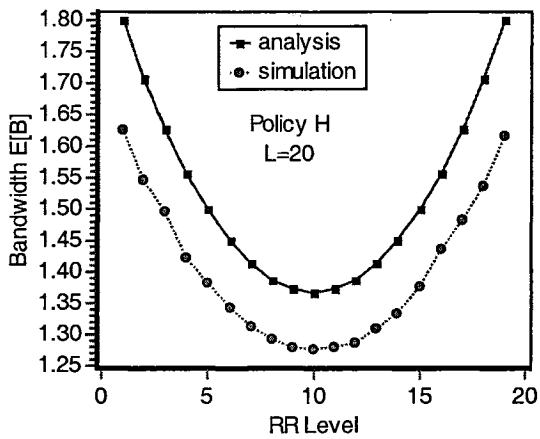


Fig.7.13 The bandwidth E[B] of 1DR for policy H in the linear topology.

Policy H can work efficiently for some topologies without RR. However, in the case of linear-like topologies, RRs are required to further aid the loss recovery of Policy H. Fig.7.13 shows the use of a RR in Policy H saves up to 30% bandwidth cosnumption of a

linear topology.

7.7 Conclusions

In this chapter, we have derived an analytical approximation for the number of transmissions for general multicast trees. Furthermore, we have analyzed the bandwidth consumption of two loss recovery polices, i.e., policy M and H. We have also compared the values of bandwidth consumption provided by the analysis and simulation. The bandwidth consumption of policy H was found to be much smaller than that of policy M. The bandwidth consumption of NAK packets was found to be negligible compared to data bandwidth.

Based on the estimation of bandwidth consumed by the two policies, the optimal RR placements have been investigated in this chapter. The optimal placements for the two policies are highly similar, although they do differ slightly. We have also investigated the optimal RR locations for the multicast network with heterogeneous loss probability. The optimal RR location should be located closer to the links of high loss probability for both loss recovery schemes.

Chapter 8 CONCLUSIONS

8.1 Contributions

The thesis addresses the performance characteristics of reliable multicast, which includes efficient and fast estimation of multicast parameters, and adaptive and distributed algorithm to a dynamic network. We list below the main findings and conclusions.

Performance evaluation

A general approach has been proposed to address the number of transmissions, bandwidth consumption, and delay, which depends on many factors such as the loss recovery schemes, the network topology, the loss characteristics, control traffic, and the RR locations. The classification of dependent, correlated, and independent loss contributes to performance analysis of general multicast trees.

A new topology parameter is introduced to approximately analyze the effects of topology on multicast performance. A bandwidth comparison of two loss recovery schemes, i.e. whole subgroup retransmissions and selective retransmissions, has been investigated by analysis and simulation. Delay performance has been obtained by simulating a reliable multicast protocol that provides reliable delivery in a best effort network such as the Internet. The effects of the background unicast traffic and channel errors on multicast performance cannot be ignored, as they may lead to buffer overflow and multicast packet

losses. We have investigated a comparison of pure ARQ and hybrid FEC/ARQ retransmissions. FEC/ARQ has smaller delay than ARQ; however, if the packets are lost more frequently in the intermediate links, FEC/ARQ does not perform much better than ARQ.

Dynamic performance

We focus mainly on dynamics of topology and the number of links, which discuss two types of pruning, i.e. the random pruning of receivers and the burst pruning of subnets. The newly defined topology parameter is used to adapt the topology change of dynamic networks for the first time. Random receiver pruning has less influence on multicast performance. This influence only becomes significant if the pruning probability is high enough. The burst pruning of subnets has a significant influence on the multicast performance.

The placement problem of RRs

Based on the bandwidth consumption of reliable multicast, the optimal RR locations have been found for different scenarios.

First, the optimal RR locations of two loss recovery policies have been found to be similar. We consider the effects of the loss characteristics of multicast networks on the optimal RR locations, which should be close to the links of high loss probabilities.

The effect of topology on optimal RR placements has also been investigated. The multicast group with homogeneous topology should be partitioned into different subgroups of equal sizes in order for the whole group to have the best performance; however, the opposite is true for topologies in which each subgroup has a very different topological structure.

Finally, we proposed a new 3-phase algorithm as an adaptive and distributed policy for RR locations. The algorithm is designed to adapt the optimal RR locations to the change of dynamic networks by the aid of some control information such as d_k and n_k. Furthermore, the control information is used to determine the optimal RR locations based on the minimization of bandwidth consumption. The efficient adaptation of repair services to the resulting dynamic network is significant to loss recovery design.

8.2 Future Work

Based on this thesis, some related future works are suggested in the following.

- The effects of joining. The joining is in a reverse way as the pruning. We consider simultaneously the processes of joining and pruning only in our simulations. There is still some analysis work to do on the joining.

- The optimization of the overall multicast packet delay in the 3-phase algorithm for different application requirement. Although the 3-phase algorithm can reduce greatly delay, the delay performance is not optimal for some cases. Different

measures are often used to optimize the performance of reliable multicast.

- The bandwidth consumption due to the control packets of the 3-phase algorithm. Although the bandwidth of these control packets is ignored (small), the results considering its impact are complete.

- The topology parameter can be used to construct the reasonable topology of reliable multicast. How to elaborate on an efficient multicast network is an interesting issue. It can be extended to other applications such as ad hoc network.

- The analytical approach can be applied to performance analysis of other reliable multicast protocols. For example, we can use our general approach to analyze the advantages and disadvantages of different multicast algorithms.

REFERENCES

[1] Sanjoy Paul, Multicasting on the Internet and its Applications, Kluwer Academic Publishers, 1998.

[2] R.Wittmann and M.Zitterbart, Multicast Communication: protocols, programming and applications, Morgan Kaufmann Publishers, May 2000.

[3] Steven Deering, "RFC 1112 - Host Extensions for IP Multicasting", http://www.ietf.org/rfc/rfc1112.txt., 1989.

[4] John Lin, Sanjoy Paul, "Reliable Multicast Transport Protocol (RMTP)", *IEEE Journal on Selected Areas in Communications*, Vol. 5, No. 3, pp. 407-421, April 1997.

[5] Brian Whetten, Gursel Taskale, "An Overview of Reliable Multicast Transport Protocol II", IEEE Networks, vol.14, No.1, pp.37-47, January 2000.

[6] John Lin, Sanjoy Paul, "RMTP: A Reliable Multicast Transport Protocol", *Proc. of IEEE INFOCOM,* pp.1414-1424, San Francisco, USA, 1996.

[7] Christos Papadopoulos, Guru Parulkar and George Varghese, "An Error Control Scheme for Large-Scale Multicast Applications", *Proc. of IEEE INFOCOM,* pp.1188-1196, San Francisco, USA, 1998.

[8] Christos Papadopoulos, Guru Parulkar and George Varghese, "Light-weight multicast services (LMS): a router-assisted scheme for reliable multicast", *IEEE/ACM Transactions on Networking,* Volume: 12 , Issue: 3 , June 2004, pp.456 – 468.

[9] Dino Farinacci, "PGM Reliable Transport Protocol Specification", Internet draft rfc3208, Aug. 1998.

[10] M.Hoffman, "Impact of virtual group structure on multicast performance", Proceedings of the 4th International COST 237 Workshop, Lisboa, Portugal, 1997, pp.165-180.

[11] Markus Yavatkar, James Griffioen, and Madhu Sudan," A reliable dissemination protocol for interactive collaborative applications," in Proceedings of ACM Multimedia, 1995.

[12] Athina P. Markopoulou, Fouad A. Tobagi, "Hierarchical Reliable Multicast: performance analysis and placement of proxies". *Proceedings of the Second International Workshop on Networked Group Communications (NGC 2000)*, Palo Alto, USA, ACM Press, pp. 27-35, Nov. 2000.

[13] Sudipto Guha, Athina Markopoulou, and Fouad Tobagi, "Hierarchical reliable multicast: performance evaluation and optimal placements of proxies", *Computer Communication*, pp.2070-2081, 2003.

[14] Hugh Holbrook,Sandeep Singhal, and David Cheriton,"Log-based receiver-reliable multicast for distributed interactive simulation", in *Proceedings of ACM SIGCOMM*, 1995

[15] L.Lehman, S.Garland, and D.Tennenhouse, "Active Reliable Multicast," *Proc.IEEE INFOCOM*, Mar.1998.

[16] Gang Feng, Kwan L.Yeung, Siew Chee Kheong.David, "Optimal Cache-Partitioning for Active Reliable Multicast", *IEEE ICC(3),* pp.1406-1410, 2000.

[17] Gang Feng, Kwan L.Yeung and Ho-Lun Wong, "Optimal Cache Allocation and Probabilitic Caching for Local Loss Recovery in Reliable Multicast", *IEEE ICC(3),* pp. 1401-1405, 2000.

[18] Sneha K. Kasera, Jim Kurose, Don Towsley, "Buffer Requirements and Replacement Policies for Multicast Repair Service", *Proceedings of the Second International Workshop on Networked Group Communicaton (NGC 2000),* Palo Alto, USA, ACM Press, pp.5-14, Nov.2000.

[19] Hwa-Chun Lin, Kuen-Feng Yang, "Placement of Repair Servers to Support Server-Based Reliable Multicast", *IEEE ICC*, vol.16, pp.1802-1806, 2001.

[20] Per-Oddvar Osland, Sneha Kumar Kasera, Jim Kurose, Don Towsley, "Dynamic Activation and Deactivation of Repair Servers in a Multicast Tree", Technical report TR1999-56 of computer science department, University of Massachusetts at Amherst, 1999.

[21] Michael J.Donahoo,Sunila.R.Ainapure, "Scalable Multicast Representative Member Selection", *IEEE INFOCOM*, pp.259-268, 2001

[22] S.Armstrong, A.Freier, and K.Marzullo, "Multicast Transport Protocol", DARPA *RFC 1301*, Feb.1992.

[23] Zhen Xiao, Kenneth P.Birman, "A Randomized Error Recovery Algorithm for Reliable Multicast", *Proc.* of *IEEE INFOCOM*, pp. 239-248, 2001.

[24] Matthew T. Lucas, Bert J.Dempsey, Alfred C. Weaver, "MESH: Distributed Error Recovery for Multimedia Streams in Wide-Area Multicast Networks", *In Proceedings of IEEE International Conference on Communication (ICC '97)*, pp.1127-1132, June 1997.

[25] Sally Floyd, Van Jacobson, Steven McCanne, Ching-Gung.Liu, Lixia.Zhang, "A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing", *Proceedings of ACM SIGCOM* '95, pp.342-356,Oct.1995.

[26] Matthew T. Lucas, Bert J.Dempsey, Alfred C. Weaver, "MESH-R:Large-Scale,Reliable Multicast Transport", *Proceedings of IEEE ICC '99*, Vancouver, BC, pp.657-664, June 1999.

[27] Ahmed Helmy, "Architectural Framework for Large-Scale Multicast in Mobile Ad Hoc Networks", *IEEE ICC*, pp.2036-2042, 2002.

[28] L.Rizzo. "pgmcc: a TCP-friendly single-rate multicast". In *Proc. of ACM SIGCOMM*, pp.17-28, 2000.

[29] Yatin Chawathe, "Scattercast: An Architecture for Internet Broadcast Distribution as an Infrastructure Service", University of California, Berkeley. Dec 2000. http://berkeley.chawathe.com/thesis/thesis-single.pdf.

[30] Kang-Won Lee, Sungwon Ha, Vaduvur Bharghavan, "IRMA: A Reliable Multicast Architecture for the Internet", *Proc. of IEEE INFOCOM,* pp.1274-1281 ,1999.

[31] C. K. Yeo, B. S. Lee and M. H. Er, "A survey of application level multicast techniques", *Computer communications*, Vol.27, issue 15, pp.1547-1568, 2004.

[32] Wong, K.-F.S.; Chan, S.-H.G.; Wan-Ching Wong; Qian Zhang; Wen-Wu Zhu; Ya-Qin Zhang; "Lateral error recovery for application-level multicast", *INFOCOM 2004*,Volume: 4 , 7-11 March 2004, pp.2708 – 2718.

[33] B.N. Levine and J.J. Garcia-Luna-Aceves, "A Comparison of Known Classes of Reliable Multicast Transport Protocols", *Proceedings of ICNP,* Oct 1996.

[34] B. N. Levine and J. J. Garcia-Luna-Aceves, "A Comparison of Reliable Multicast Protocols," *ACM Multimedia Systems Journal,* Vol. 6, No.5, August 1998. pp 334-348.

[35] Y.Chawathe, S.McCanne, E.Brewer, "RMX: Reliable Multicast in Heterogeneous Networks", INFOCOM. 2000.

[36] Joerg Nonnenmacher, E.W.Biersack, "Performance Modeling of Reliable Multicast Transmission", *Proc. of IEEE INFOCOM*, Vol. 2, pp. 471 -479, 1997.

[37] Joerg Nonnenmacher, E.W.Biersack, "Reliable Multicast: where to use FEC", IEEE INFOCOM 1997.

[38] Joerg Nonnenmacher and Ernst W.Biersack, "The impact of routing on multicast error recovery", *Comput.Commun.*, vol.21. No.10, pp.867-879, July 1998.

[39]Nonnenmacher, J.; Biersack, E.W., "Optimal multicast feedback", Proceedings of *IEEE INFOCOM,*. Vol.3, pp.964 -971, 29 Mar-2 Apr 1998.

[40]Joerg Nonnenmacher, E.W.Biersack, and Don Towsley "Parity-Based Loss Recovery for Reliable Multicast Transmission", *IEEE/ACM Transactions on Networking,* pp.349-361, Vol.6, No.4, August 1998.

[41]Joerg Nonnenmacher, Martin Lacher, Matthias Jung, Ernst W.Biersack, Georg Carle, "How bad is Reliable Multicast without Local Recovery?", *Proceedings of IEEE INFOCOM*, vol.3. pp. 972 -979, 29 Mar-2 Apr 1998.

[42]Luigi Rizzo, "Effective erasure codes for reliable computer communication protocols", *Comput. Commun. Rev.,* vol.27, No.2, pp. 24-36, Apr. 1997.

[43]Taku Noguchi, Miki Yamamoto, "Reliable multicast protocol applying local FEC", *IEICE Transactions on communications*, pp. 690-698, Vol. E86-B, No. 2, 2003.

[44]Dan Rubenstein, Sneha Kasera,Don Towsley, and Jim Kurose, "Improving Reliable Multicast Using Active Parity Encoding Services (APES)", *Proceedings of IEEE INFOCOM,* pp.1248-1255, 1999.

[45]D.Rubenstein, J.Kurose and D.Towsley, "Real-time reliable multicast using proactive forward error correction", *Proc.NOSSDAV* 98, UK, 1998.

[46]Dan Li and David R.Cheriton, "Evaluating the utility of FEC with reliable multicast", *Proceedings of ICNP'99*, Toronto, Canada, October 1999.

[47] Adam M.Costello, Steven McCanne, "Search Party: Using Randomcast for Reliable Multicast with Local Recovery", *Proc. Of IEEE INFOCOM*, pp.1256-1264, 1999.

[48] Radoslavov, P.; Papadopoulos, C.; Govindan, R.; Estrin, D.; "A comparison of application-level and router-assisted hierarchical schemes for reliable multicast", *IEEE/ACM Transactions on Networking,* Volume: 12 , Issue: 3 , June 2004, pp.469 – 482.

[49] Pavlin Radoslavov, Christos Papadopoulos, R. Govindan, Deborah Estrin, "A Comparison of Application-Level and Router-Assisted Hierarchical Schemes for reliable Multicast", *IEEE INFOCOM*, pp. 229-238,2001

[50] D.Li, D.Cheriton, "OTERS:a reliable multicast protocol", *Proc. ICNP'98*, October 1998, Austin, Texax, pp.237-245.

[51] Christos Papadopoulos, Emmanouil Laliotis, "Incremental Deployment of a Router-assisted Reliable Multicast Scheme", *Proceedings of the Second International Workshop on Networked Group Communicaton (NGC 2000),* Palo Alto, USA, ACM Press, pp.37-46, Nov.2000.

[52] Jun Peng and Biplab Sikdar, "Multicast loss recovery with active injection", *the 12th IEEE International Conference on Computer Communications and Networks (ICCCN'03)*, Dallas Texas, USA, October 20 - 22, 2003, pp.81-86.

[53] Kamil Sarac, Pavan Namburi, and Kevin C.Almeroth, "SSM Extensions: Network layer support for Multiple senders in SSM", *the 12th IEEE International Conference on*

Computer Communications and Networks (ICCCN'03), Dallas Texas, USA, October 20 - 22, 2003, pp.74-80.

[54] John.Byers, Michael. Luby, Michael Mitzenmacher, et al, "A Digital Fountain approach to reliable distribution of bulk data", *SIGCOMM 1998*,Volume 28 Issue 4. pp.56-67.

[55] C.Huitema, "The case for packet level FEC", in Proc.IFIP 5th Int. Workshop on protocols for high speed networks (PfHSN'96), Sophia Antipolis, France, Oct.1996, pp.110-120.

[56] R.Kermode, "Scoped Hybrid Automatic repeat request with forward error correction (SHARQFEC)", *Proceedings of ACM SIGCOMM'98*, Vancouver, CA, September 1998.

[57] D.Rubenstein, J.Kurose, D.Towsley, "A study of proactive hybrid FEC/ARQ and scalable feedback techniques for reliable, real-time multicast", *Computer Communications*, 24, 2001, pp.563-574.

[58] John Byers, Michael Luby, Michael Mitzenmacher, "Accessing multiple mirror sites in parallel : Using Tornado Codes to speed up downloads", *INFOCOM 1999.*

[59] S.Pingali, D.Towsley, and J.F.Kurose, "A comparison of sender-initiated and receiver-initiated reliable multicast protocols", in *Proceedings of the Sigmetrics Conference on Measurement and Modeling of Computer Systems*, New York, p221-230, ACM Press, May 1994.

[60]E.Amir,S.McCanne,and R.Katz, "An Active Service Framework and Its Application to Real Time Multimedia Transcoding," *Proc. ACM SIGCOMM.*, Sept. 1998.

[61]C.C.Cheung, D.H.K. Tsang and S. Gupta, "State Dependent Multicast Routing for Single Rate Loss Networks", *IEICE Trans. Commun.* Vol.E84-B, No.5, May 2001.

[62]Pravin Bhagwat, Partho P.Mishra, Satish K.Tripathi, "Effect of Topology on Performance of Reliable Multicast Communication", *Proc. of IEEE INFOCOM.* 1994

[63]Christian Mailhofer, "A Bandwidth Analysis of Reliable Multicast Transport Protocols", *Proceedings of the Second International Workshop on Networked Group Communicaton (NGC 2000)*, Palo Alto, USA, ACM Press, pp.15-26, Nov.2000.

[64]Sneha K. Kasera, Supratik Bhattacharyya, Mark Keaton, et al, " Scalable Fair Reliable Multicast Using Active Servies", *IEEE Networks*, vol.14, No.1, pp.48-57, January 2000.

[65]Sneha Kumar Kasera,Gisli Hjallmtysson, DonaldF. Towsley, and James F.Kurose, "Scalable Reliable Multicast Using Multiple Multicast Channels", *IEEE/ACM Transactions on Networking,* vol.8, No.3, pp.294-309,June 2000.

[66]Ching-Gung Liu, Deborah Estrin, Scott Shenker, and Lixia Zhang, "Local error recovery in SRM: Comparison of two approaches", in *IEEE /ACM Transactions on Networking*, Dec.1998.

[67]B.Levine and J.Garcia-Luna-Aceves, " A Comparison of reliable multicast protocols", *ACM Multimedia Systems*, V.6,No.5,pp334-348, Sept.1998.

[68]Bo Li, M.Golin, G.Italiano, et al, " On the optimal placement of web proxies in the Internet", *Proc. of IEEE INFOCOM*, pp.1282-1290, 1999.

[69]Zuo Wen Wan, Michel Kadoch, and Ahmed Elhakeem, "Performance Evaluation Of Tree-based Reliable Multicast", *the 12th IEEE International Conference on Computer Communications and Networks (ICCCN'03)*, Dallas Texas, USA, October 20 - 22, 2003, pp.67-73.

[70]MiKi Yamamoto, Makoto Yamaguchi, Takashi Hashimoto, Hiromasa Ikeda, "Performance Evaluation of Reliable Multicast Communication Protocol with Network Support", *IEEE Globecom*, San Francisco, USA, pp,1736-1741,2000.

[71]M.S.Lacher, J.Nonnenmacher, and E.W.Biersack, "Performance comparison of centralized versus distributed error recovery for reliable multicast", *IEEE/ACM Transactions on Networking,* Vol.8, No.2, pp.224-238, April 2000.

[72]Christian Mailhofer, "A Bandwidth Analysis of Reliable Multicast Transport Protocols", *Proceedings of the Second International Workshop on Networked Group Communicaton (NGC 2000),* Palo Alto, USA, ACM Press, pp.15-26, Nov.2000.

[73]S.Pingali, D.Towsley, and J.F.Kurose, "A comparison of sender-initiated and receiver-initiated reliable multicast protocols", *Proceedings of the Sigmetrics Conference on Measurement and Modeling of Computer Systems*, New York, p221-230, ACM Press, May 1994.

[74]Zuo Wen Wan, Michel Kadoch, and Ahmed Elhakeem, "Topological effect on optimal partitioning of multicast trees", *The 10th IEEE International Conference On Electronics, Circuits And Systems (ICECS 2003),* University Of Sharjah, Sharjah, United Arab Emirates (UAE), December 14 - 17, 2003, pp.531-534.

[75]Zuo Wen Wan, Michel Kadoch, and Ahmed Elhakeem, "Optimal partition of binary tree for hierarchical reliable multicast", *Proceedings of the IASTED International Conference on Communications and Computer Networks (CCN 2002),* November 4-6, 2002, Cambridge, MA,USA, pp.391-396, November 2002.

[76]C.Mailhofer, K.Rothermet and N. Mantei, "A Throughput Analysis of Reliable Multicast Transport Protocols", *IEEE ICC 2000.*

[77]D.A.Villela, O.C.Duarte, "Improving scalability on reliable multicast communications", *Computer Communications* 24(2001), pp. 548-562,2001.

[78]Zuo Wen Wan, Michel Kadoch, and Ahmed Elhakeem, "Performance Comparison of Two Loss Recovery Policies for Hierarchical Reliable Multicast", *the IASTED International Conference on Communications Systems and Applications (CSA 2004),* Banff, Canada, July 8-10, 2004, pp132-137.

[79]Miki Yamamoto, James F.Kurose, Donald F.Towsley and H. Ikeda, "A Delay Analysis of Sender-Initiated and Receiver-Initiated Reliable Multicast Protocols", *IEEE INFOCOM 1997.* pp. 480-488.

[80] Brosh, E.; Shavitt, Y.; " Approximation and heuristic algorithms for minimum delay application-layer multicast trees", *INFOCOM 2004*,Volume: 4 , 7-11 March 2004, pp.2697 – 2707.

[81] Miki Yamamoto, Takashi Hashimoto, Hiromasa Ikeda, "Performance Evaluation of Local Recovery Group Configuration in Reliable Multicast", *IEICE Trans.Commun.*, vol.E83-B, no.12, pp.2675-2683, December 2000.

[82] Sara Alouf, Eitan Altman and Philippe Nain, "Optimal on-line estimation of the size of a dynamic multicast group", *IEEE INFOCOM*, pp. 1109-1118, 2002.

[83] G.Armitage, " IP Multicasting over ATM Networks", *IEEE Journal on Selected Areas in Communications*, Vol.15,No.3, April 1997.

[84] Ryuji Somegawa, Kenjiro Cho, Yuji Sekiya, and Suguru Yamaguchi, "The effects of server placement and server selection for Internet services', *IEICE Transactions on Communications*, Vol.E86-B, No.2, Feburary 2003, pp.542-551.

[85] Danny Dolev, Osnat Mokryn, Yuval Shavitt, "On multicast trees: structure and size estimation", *IEEE INFOCOM*, 2003, pp.1011-1021.

[86] Qi He, Mostafa Ammar, "Dynamic host-group/multi-destination routing for multicast sessions", *the 12th IEEE International Conference on Computer Communications and Networks (ICCCN'03)*, Dallas Texas, USA, October 20 - 22, 2003, pp.428-433.

[87] Sassan Pejhan, Mischa Schwartz, and Dimitris Anastassiou, "Error control using retransmission schemes in multicast transport protocols for real-time media", *IEEE/ACM Transactions on Networking,* vol.4, no.3, June 1996,pp.413-427.

[88] M.Handley, "An examination of Mbone performance", Technical Report, UCL and ISI, January 1998.

[89] B.Levine, S.Paul, J.J.Garcia-Luna-Aceves, "Organizing multicast receivers deterministically by packet-loss correlation", *Proc.ACM Multimedia'98*, pp.201-210, Oslo, Norway, Sept.1998.

[90] M.Yajnik, J,Kurose, D.Towsley, "Packet loss correlation in the Mbone multicast network", *Proceedings of IEEE Global Internet conferenc*e. London, November 1996.

[91] P.Krishnan, D.Raz, Y.Shavitt, "The cache location problem", *IEEE Transactions on Networking,* 8(5), pp.568-582, Oct.2000.

[92] Christian Mailhofer, Kurt Rothermel, "Optimal branching factor for tree-based reliable multicast protocols", *Computer Communications*, 25(2002), pp.1018-1027, 2002.

[93] L.Zhang, S.Floyd, V.Jacobson, "Adaptive web caching", NLANR Web cache workshop, Boulder, CO, June 1997.

[94] Sherlia Shi, Jonathan S. Turner, "Placing Servers in Overlay Networks", *Proc. of International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPETS)*, San Diego, July 2002.

[95] Kwan L. Yeung and Ho-Lun T. Wong, "Caching policy design and cache allocation in active reliable multicast", *Computer Networks, Volume 43, Issue 2, 7,October, 2003,pp.177-193.*

[96] De-Nian Yang; Wanjiun Liao, "Optimizing state allocation for multicast communications", *INFOCOM 2004*, Vol,4, 7-11 March 2004, pp.2719 – 2730.

[97] Wonyong Yoon, Dongman Lee, Hee Yong Youn, Seungik Lee, Seok Joo Koh, "A Combined Group/Tree Approach for Scalable Many-to-many Reliable Multicast", *IEEE INFOCOM 2002*, pp.1336-1345.

[98] Charikar, M.; Naor, J.; Schieber, B.; "Resource optimization in QoS multicast routing of real-time multimedia", *IEEE/ACM Transactions on Networking*, Volume: 12 , Issue: 2 , April 2004, pp.340–348.

[99] Injong Rhee, Srinath R. Joshi, Minsuk Lee, S.Muthukrishnan and V. Ozdemir, "Layered Multicast Recovery", *IEEE INFOCOM*, pp.805-813, 2000.

[100] Sarac, K.; Almeroth, K.C. "Tracetree: a scalable mechanism to discover multicast tree topologies in the Internet", *IEEE/ACM Transactions on Networking*, Volume:12, Issue: 5 , Oct. 2004, pp.795 – 808.

APPENDIX A

From the lemma 1- lemma 3 in chapter 4 ((4.20) - (4.27)), one has the following equations:

$$\sum_{k_1=1}^{N}\sum_{k_2\in D_{k_1}} d_{k_2} = \sum_{k_1=1}^{N}\sum_{k_2\in d_{k_1}} d_{k_2} + \sum_{k_1=1}^{N}\sum_{k_2\in n_{k_1}} d_{k_2} = \frac{1}{2}\sum_{k_1=1}^{N}(3d_{k_1}-1)d_{k_1} \tag{A.1}$$

$$\sum_{k_1=1}^{N}\sum_{k_2\in D_{k_1}} n_{k_2} = \sum_{k_1=1}^{N}\sum_{k_2\in d_{k_1}} n_{k_2} + \sum_{k_1=1}^{N}\sum_{k_2\in n_{k_1}} n_{k_2} = \sum_{k_1=1}^{N}(n_{k_1}+d_{k_1})n_{k_1} \tag{A.2}$$

$$\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}} (n_{k_1}+n_{k_2}) = \sum_{k_1=1}^{N} n_{k_1}(N-d_{k_1}-n_{k_1}) + \sum_{k_1=1}^{N}\sum_{k_2=1}^{N} n_{k_2} \tag{A.3}$$

$$-\sum_{k_1=1}^{N}\sum_{k_2\in D_{k_1}} n_{k_2} = N^2(x-1) - 2\sum_{k_1=1}^{N} n_{k_1}(d_{k_1}+n_{k_1})$$

$$\frac{1}{2!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}} 1 = \frac{1}{2!}\sum_{k_1=1}^{N}\Big(\sum_{k_2=1}^{N} 1 - \sum_{k_2\in D_{k_1}} 1\Big) = \frac{N(N-x)}{2} \tag{A.4}$$

$$\frac{1}{3!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}\sum_{k_3\notin D_{k_1}, D_{k_2}} 1 = \frac{1}{3!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}(N-d_{k_1k_2}-n_{k_1}-n_{k_2}) \tag{A.5}$$

$$= \frac{N^2(N-x)}{6} - \frac{1}{3!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}(d_{k_1k_2}+n_{k_1}+n_{k_2})$$

$$\sum_{k_1=1}^{N}\sum_{k_2\in D_{k_1}}(d'_{k_2}+n'_{k_2}) = \sum_{k_1=1}^{N}\sum_{k_2\in d_{k_1}}(d'_{k_2}+n'_{k_2}) + \sum_{k_1=1}^{N}\sum_{k_2\in n_{k_1}}(d'_{k_2}+n'_{k_2})$$

$$= \sum_{k_1=1}^{N} d_{k_1}(d_{k_1}+n_{k_1}) + \sum_{k_1=1}^{N}\sum_{k_2\in n_{k_1}}(d_{k_2}+n_{k_2}) \tag{A.6}$$

$$= \sum_{k_1=1}^{N}[(d_{k_1})^2 + 4d_{k_1}n_{k_1} - n_{k_1}] = 3\sum_{k_1=1}^{N}[(d_{k_1})^2 - d_{k_1}] + N$$

$$\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}} d_{k_1k_2} \approx \sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}(d_{k_1}+d_{k_2})$$

$$= \sum_{k_1=1}^{N}\Big[\sum_{k_2=1}^{N}(d_{k_1}+d_{k_2}) - \sum_{k_2\in D_{k_1}}(d_{k_1}+d_{k_2})\Big]$$

$$= N^2(x+1) - \frac{N(x+1)}{4} - \sum_{k_1=1}^{N}\Big[\frac{3}{2}(d_{k_1})^2 + 3d_{k_1}n_{k_1}\Big] \tag{A.7}$$

$$= N^2(x+1) - \frac{N(x+1)}{2} - 3\sum_{k_1=1}^{N}(d_{k_1})^2$$

$$\sum_{k_1=1}^{N}\binom{N+d_k-1}{2} = \frac{N(N-2)(N+x)}{2} + \frac{N(x+1)}{4} + \sum_{k_1=1}^{N}\frac{1}{2}(d_k)^2 \tag{A.8}$$

From (4.45), y is rewritten as the follows:

$$
\begin{aligned}
y = -2\binom{N}{3} + 2[&\frac{1}{3!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}\sum_{k_3\notin D_{k_1k_2}}1 + \sum_{k_1=1}^{N}\binom{N+d_k-1}{2} \\
&-\frac{1}{2!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}(N+d_{k_1k_2}-2)] - 3\sum_{k_1=1}^{N}(N+\frac{d_{k_1}-n_{k_1}-3}{2})(d_{k_1}+n_{k_1}) \\
&+\frac{3}{2!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}(d_{k_1k_2}+n_{k_1}+n_{k_2}) + \sum_{k_1=1}^{N}\sum_{k_2\in D_{k_1}}(d'_{k_2}+n'_{k_2})
\end{aligned}
\tag{A.9}
$$

Substituting (A.1) - (A.8) into (A.9), we obtain the following y value:

$$
y = \frac{5}{6}\sum_{k=1}^{N}[(d_k)^2 - (n_k)^2] + \frac{xN}{6}. \tag{A.10}
$$