Performance Evaluation of Dynamic Tree-based Reliable Multicast

Zuo Wen Wan[†], Michel Kadoch[††], *nonmembers,* and Ahmed Elhakeem[†], *IEICE member*

Summary Due to the pruning and joining of members, multicast groups are dynamic. Both the topology and the total number of links change during multicast sessions, and the multicast performance, measured in terms of the bandwidth consumption, will change accordingly. In this paper, we investigate the dynamic performance of multicast communication with homogeneous packet loss probability; indeed, we evaluate the effects of the pruning of receivers and of subnets, after which we find the optimal placements of repair servers. A new 3-phase algorithm for adapting the optimal repair server placements to the dynamic changes of network topologies is presented and analyzed.[1]

***Keywords**: reliable multicast, dynamic, optimization, topology, repair service.*

1. Introduction

Multicast provides an efficient means of transmitting data to a large number of receivers. Many applications such as the distribution of software, financial and billing data require reliable multicast delivery. In order to provide reliability, the retransmissions of lost packets are necessary and loss recovery plays a key role in the design of reliable multicast. Hierarchical loss recovery is often used in tree-based reliable multicast. For this loss recovery, special receivers or routers that have repair functions are assigned to the retransmission handling of the lost packets [1][2]. Loss recovery is limited to a local region, which saves a lot of bandwidth. In this work, we use routers with added repair functionality, which we call repair routers (RR), to remulticast the missing packet. RRs function like the designated receivers in RMTP[1]. It is obvious that the placement of these special receivers or routers has influence on the multicast performance. The total bandwidth consumption depends greatly on the partitioning of the tree, i.e. the placement of special receivers or routers. [2]

Some references, namely [3][4][5][6][7][8][11], 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. Furthermore, the packet loss probability is often dynamic, which has been discussed for a given topology in [9]. In this work, we will focus only on the dynamics of topology and on the optimal placement of RRs mainly for non delay sensitive applications.

In this paper, we have concentrated on applications requiring reliable transmissions. For such applications, every packet should be received correctly, that means, occasional retransmissions of information data. This may occasionally lead to delay and delay jitter (packets are not received in sequence or synchronized). However, the new technique used in this paper, as indicated by the results, will reduce the frequency of such retransmissions and consequently delay jitter. By our new optimal router locations, less retransmission means less delay jitter, as compared to recent works on reliable multicast [1][3][5]. We evaluate only the improvement in bandwidth consumption specifically for non delay sensitive applications. Due to our optimal location technique, however, one can easily see that even for delay sensitive applications, our new policy will work better to reduce bandwidth consumption and consequently delay jitter. For delay sensitive applications (not subject of this paper), one can use pure FEC (forward error correction) techniques, which do not need retransmissions. But, occasional loss of information packets will be encountered. Our companion work [12] investigates the applications of our new optimal policy, and the results show that bandwidth consumption is optimized while delay is just reduced compared to the technique not using our optimal location policy.

2. Analysis of Static Multicast Trees

The expected number of transmissions E[M] of a packet until received by all receivers is often used to evaluate the multicast performance for a static multicast tree [2][3][4]. It can be calculated by the numerical evaluation of recursive equations [3][4]. However, the computation of E[M] may be very intensive for a general topology. Even if the reduction technique is used, the computation of E[M] can be exponential with the number of nodes [2]. Based on the intensive computations of E[M], it is difficult to calculate criteria, such as bandwidth consumption, for a large multicast network. It also is difficult to calculate E[M] for a

Manuscript received August 16, 2004
Manuscript revised October 30, 2004

[†]The authors are with the Department of Electrical and Computer Engineering at Concordia University, Montreal, Canada, Email: {zw_wan, ahmed}@ece.concordia.ca.
[††]The author is with the Department of Electrical Engineering at Ecole de Technologie Superieure, Montreal, Canada. Email: kadoch@ele.etsmtl.ca.

[1]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.

dynamic network where the topology and the number of links change. Efficient estimation of E[M] is necessary. In this paper we follow a completely analytical approach for the evaluation of the E[M] without needing recursion. For analysis convenience, we use the homogeneous packet loss probability in the following, i.e, all links have the same packet loss probability p.

2.1 The Probability Distribution 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 7 loses a packet in Fig. 1, n_7 nodes will not receive the packet where n_7 is the number of links under the node 7. The purpose of the next retransmission is to have the requested packet received at the n_7 nodes. The requested packet must pass through the path from the sender to node 7 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 7. Therefore, we only concentrate on whether these nodes receive the packet correctly in the next retransmission. If losses out of the n_7 nodes recur in the next retransmission, we need to repeat the above process until all nodes receive the packet correctly.

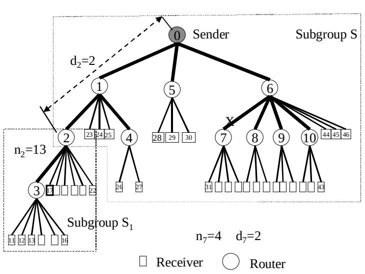

Fig. 1 An example of multicast trees

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, e.g. the losses of node 3 and 6 are independent in Fig. 1. However, if one link is under the subtree of another link loss, the losses are dependent; for example, the loss of node 2 clearly results in the loss of node 3. In the following, losses refer to independent losses that do not affect each other, and dependent loss is considered by factor n_k defined in Table 1.

1) No loss occurs

The probability that no packet loss over N links occurs is

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

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

$$\text{Prob.(only node } k_1 \text{ loses)} = p(1-p)^{N-n_{k_1}-1}, \tag{2}$$

where n_{k1} is the number of links under node k_1. Readers are urged to read Table 1 for the multicast notations.

Table 1 Notations For a Reliable Multicast

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,.... 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 in previous transmissions.
i.	G_t is the total number of links in level t.
j.	λ is the pruning probability of receivers.

3) j losses occur

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

$$\text{Prob.(node } k_1, k_2 \ldots k_j \text{ lose)} = \begin{cases} p^j(1-p)^{N-\sum_{s=1}^{j} n_{k_s} - j}, & k_2 \notin D_{k_1}, \cdots, k_j \notin D_{k_1k_2\cdots k_{j-1}} \\ 0, & otherwise \end{cases} \tag{3}$$

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 $k_1,k_2,\ldots,k_i$, (and is also the set of these links). For example, d_7=4, n_7=2, $d_{3,6}=4$, $d_{3,4}=4$, and $d_{3,4,6}=5$ in Fig. 1. $D_{k_1k_2\cdots k_i} = d_{k_1k_2\cdots k_i} \cup n_{k_1} \cup n_{k_2} \cup \cdots \cup n_{k_i}$ (i=1,2,...,j) is the 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:

$$P(M=1) = (1-p)^N \tag{4}$$

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

$$\begin{aligned} P(M=2) &= \sum_{k_1=1}^{N} p(1-p)^{N-n_{k_1}-1} P(M=1\,|\,k_1) \\ &+ \frac{1}{2!}\sum_{k_1=1}^{N} \sum_{k_2 \notin D_{k_1}} p^2(1-p)^{N-n_{k_1}-n_{k_2}-2} P(M=1\,|\,k_1k_2) + \cdots \end{aligned} \tag{5}$$

where $P(M=1|k_1k_{2\ldots}k_s)$ is the conditional probability that one retransmission is needed to make multicast successful if nodes $k_1, k_2, \ldots, k_s$ lose the packet in the first transmission.

Similarly, we can obtain the probability of m transmissions.

$$P(M=m)=\sum_{k_1=1}^{N} p(1-p)^{N-n_{k_1}-1} P(M=m-1|k_1)+\cdots$$
$$+\frac{1}{j!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}\cdots\sum_{k_j\notin D_{k_1\cdots k_{j-1}}} p^j(1-p)^{N-\sum_{s=1}^{j} n_{ks}-j} P(M=m-1|k_1k_2\cdots k_j)+\cdots \tag{6}$$

where $P(M=m-1|k_1k_2\ldots k_j)$ is the conditional probability of m-1 retransmissions after node k_1, node k_2, ..., k_j lose the packet in previous transmissions.

2.2 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. For a multicast subgroup of $Np<1$, we may expand the above probability density function of M. In the following, we will consider the function until the 3rd order approximation of p or Np.

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 $P(M=1|k_1)=(1-p)^{d_{k_1}+n_{k_1}}$. Similarly, one can obtain

$$P(M=1|k_1k_2\cdots k_j)=(1-p)^{d_{k_1k_2\cdots k_j}+\sum_{s=1}^{j} n_{k_s}}, \quad k_2\notin D_{k_1},\cdots,k_j\notin D_{k_1k_2\cdots k_{j-1}} \tag{7}$$

Substituting (7) into (5), one has

$$P(M=2)=Np-[\binom{N}{2}+\sum_{k_1=1}^{N}(n_{k_1}+d_{k_1})]p^2+a_{23}p^3+\cdots \tag{8}$$

where

$$a_{23}=\frac{1}{3!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}\sum_{k_3\notin D_{k_1k_2}}1+\sum_{k_1=1}^{N}\binom{N+d_k-1}{2}-\frac{1}{2!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}(N+d_{k_1k_2}-2) \tag{9}$$

Similarly, one may use (8) to evaluate $P(M=2|k_1,k_2,\ldots,k_j)$. If one only considers the 3rd order approximation, one achieves the following result:

$$P(M=3)=\sum_{k_1=1}^{N}(d_{k_1}+n_{k_1})p^2+a_{33}p^3+\cdots \tag{10}$$

where

$$a_{33}=-\sum_{k_1=1}^{N}[(N-\frac{3}{2}+\frac{d_{k_1}-n_{k_1}}{2})(d_{k_1}+n_{k_1})+\sum_{k_2\in D_{k_1}}(d'_{k_2}+n'_{k_2})]$$
$$+\frac{1}{2!}\sum_{k_1=1}^{N}\sum_{k_2\notin D_{k_1}}(d_{k_1k_2}+n_{k_1}+n_{k_2}) \tag{11}$$

Using (10), one can obtain P(M=4).

$$P(M=4)=p^3\sum_{k_1=1}^{N}\sum_{k_2\in D_{k_1}}(d'_{k_2}+n'_{k_2})+\cdots \tag{12}$$

So far, we derive until the 3rd order approximation for the probability density of the number of transmissions. One may find the expected number of transmissions easily using (4), (8), (10), and (12): [8], [12]

$$E[M]=\sum_{m=1}^{\infty} mP(M=m)\approx 1+Np+[u-\binom{N}{2}]p^2+\binom{N}{3}p^3 \tag{13}$$

where $$u=\sum_{k=1}^{N}(n_k+d_k) \tag{14}$$

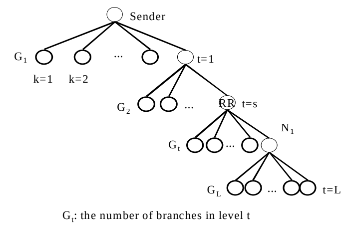

Fig. 2 Another multicast tree.

2.3 Discussions

For a homogenous multicast group having the same packet loss probability p for all links, (13) shows that E[M] depends on the number of links N, the packet loss probability p and the topology parameter u. 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 topology. Thus, (13) can be rewritten as follows.

$$E[M]=g(N)+up^2 \tag{15}$$

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

$$g(N)=1+Np-\binom{N}{2}p^2+\binom{N}{3}p^3 \tag{16}$$

Links in the same level have the same d_k, thus u only depends on the distribution of the number of links in each level, i.e., u can be rewritten as follows due to $\sum_{k=1}^{N}n_k=\sum_{k=1}^{N}(d_k-1)$ [8].

$$u=2\sum_{k=1}^{N}d_k-N=2\sum_{t=1}^{L}tG_t-N \tag{17}$$

where G_t is the total number of links in level t, e.g. $G_1=3$ and $G_2=15$ in Fig. 1. u depends on how links are distributed to each level. The parameter u roughly reflects the effects of topology.

Fig. 3, Fig. 4, and Fig. 5 each show a comparison of several topologies for the exact (i.e, the earlier recursion solution) and the approximate solutions (13) for E[M]. For the small E[M] value, (13) gives a simple approximation to the exact solution. We will use the parameter u to adapt the topological changes of dynamic networks.

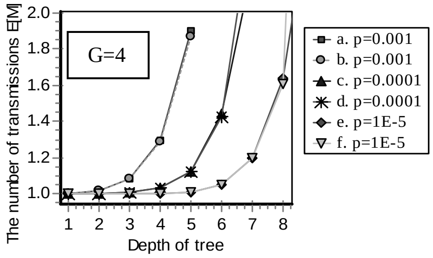

Fig. 3 A comparison of E[M] for full k-ary trees where G is the number of branches for each intermediate node. a,c,e: exact solution; b,d,f: approximate solution.

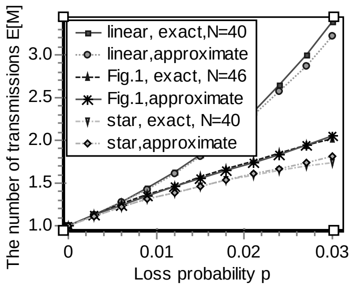

Fig. 4 A comparison of E[M] for different topologies,i.e. linear, star and the topology in Fig. 1

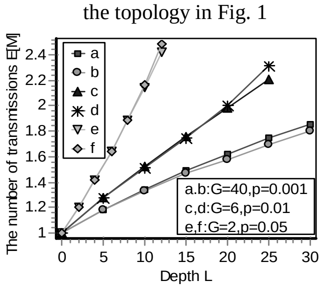

Fig. 5 A comparison of E[M] for the trees in Fig. 2. $G_1=G_2=G_3=...=G_L=G$. a,c,e: exact solutions; b,d,f: approximate solutions.

3. Analysis of dynamic Multicast networks

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

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.

3.1 The pruning of receivers

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. 1) in level t. Then $G_t = w_t + w_t^T$, for example, in Fig. 1, w_2=9, w_2^T =6. For a multicast group with L levels, the total number w of receivers is defined as follows:

$$w = w_1 + w_2 + \cdots + w_L \tag{18}$$

For example, in Fig. 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 $P(j_t)$ that j_t receivers are pruned from w_t receivers in level t is binomially distributed.

$$P(j_t) = \binom{w_t}{j_t} \lambda^{j_t} (1-\lambda)^{w_t - j_t} \tag{19}$$

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 $P(j_1, j_2, \cdots, j_L)$ that j_1, j_2, ..., j_L receivers prune from level 1,2, …, L, respectively is given by:

$$P(j_1, j_2, \cdots, j_L) = P(j_1)P(j_2)\cdots P(j_L) \tag{20}$$

where w is defined by (18) while j is the total number of pruned receivers for the initial network, i.e.

$$j = j_1 + j_2 + \cdots + j_L \tag{21}$$

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 $w_t + w_t^T - j_t$. Thus, the network topology changes accordingly, and (17) writes u in the following manner:

$$u = 2\sum_{t=1}^{L} t(w_t + w_t^T - j_t) - (N_0 - j) = u_0 - (2\sum_{t=1}^{L} tj_t - j) \tag{22}$$

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. 1) on the network topology parameter using (22), i.e.

$$E_1[u] = \sum_{j_1}^{w_1}\sum_{j_2}^{w_2}\cdots\sum_{j_L}^{w_L} P(j_1, j_2, \cdots, j_L)u = u_0 - \lambda u_r \tag{23}$$

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

$$u_r = 2\sum_{t=1}^{L} t w_t - w \tag{24}$$

For a given network topology, the number of transmissions is given by (15):

$$E_2[M|N,u] = g(N) + up^2 \tag{25}$$

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.

$$E_1[E_2[M \mid N,u]] = E_1[g(N)] + E_1[u]p^2 \tag{26}$$

where the E_1 is the average value of the transmission times over all possible network topologies due to the pruning of receivers, 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. For small λ, one can obtain the following approximation by substituting (23) and (16) into (26) and ignoring the terms of j^2, j^3, and j^4.

$$\begin{aligned} E_1[E_2[M \mid N,u]] &= E_2[M_0] \\ &- \lambda[wp + (u_r - wN_0)p^2 + \frac{1}{2} wN_0(N_0 - 2)p^3] \end{aligned} \tag{27}$$

where $E_2[M_0] = g(N_0) + u_0 p^2$ 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.

$$E_1[C] = E_1[NE_2[M \mid N,u]] = E_1[Ng(N)] + E_1[uN]p^2 \tag{28}$$

Assuming that λ is small, j<<N, it is easy to obtain an estimate of the bandwidth consumption by ignoring the terms of j^2, j^3, and j^4.

$$\begin{aligned} E_1[C] &= N_0 E_2[M_0] - w\lambda[1 + 2N_0 p - (\frac{3}{2}N_0^2 - N_0)p^2 \\ &+ \frac{1}{6}(4N_0^3 - 9N_0^2 + 4N_0)p^3] - (u_r N_0 + u_0 w)\lambda p^2 \end{aligned} \tag{29}$$

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.

In Fig. 6, we give the results of the average number of transmissions (27) over all topologies for receivers pruning off the network in Fig. 1. Increasing the loss probability p, the total number of transmissions becomes larger. When the pruning probability λ increases, the average number of transmissions decreases. This is because more links may prune from the multicast group and the total number of links decreases for large λ. From Fig. 6, the effect of receiver pruning on E[M] is not significant for small λ. However, Fig. 7 shows that the bandwidth consumption E[C] decreases as λ increases. These results have tendencies similar to those of the above number of transmissions.

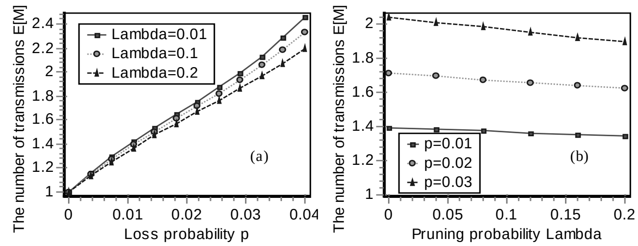

Fig. 6 The average number of transmissions for receiver pruning in the multicast tree of Fig. 1.

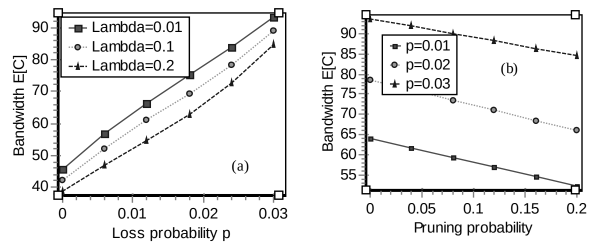

Fig. 7 The average bandwidth consumption for receiver pruning in the multicast tree of Fig. 1.

3.2 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.

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

$$u_0 = 2\sum_{k \in N_0} n_k + N_0 = 2\sum_{k \in N} n_k + N + 2\sum_{k \in N_i} n_k + N_i \tag{30}$$

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

$$u_i = 2\sum_{k\in N_i} n_k + N_i \quad (31)$$

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.

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. 1. 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.

$$u = 2[\sum_{k\in N} n_k - (d_i - 1)N_i] + N \quad (32)$$

From (30), (31) and (32), one has the following equation:

$$u_0 = u + u_i + 2N_i(d_i - 1) \quad (33)$$

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:

$$E[M] = g(N) + up^2 \quad (34)$$

where g(N) is given by (16), u is given by (33).

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 the following:

$$E[M]_s = g(N_s) + u_s p^2, \qquad s = 0, i \quad (35)$$

Substituting (34) and (35) into (33), one obtains

$$E[M]_0 - E[M] - E[M]_i = g(N_0) - g(N) - g(N_i) + 2N_i(d_i - 1)p^2 \quad (36)$$

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.

$$E[M] + E[M]_i + E[M]_j - E[M]_0 = g(N) + g(N_i) + g(N_j) - g(N_0) - 2N_i(d_i - 1)p^2 - 2N_j(d_j - 1)p^2 \quad (37)$$

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

Fig. 8 and Fig. 9 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.

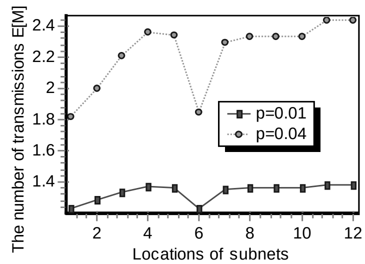

Fig. 8 The number of transmissions after the pruning of one subnet.

Table 2 The values of the x axis corresponding to the locations of two subnets rooted in node i and j shown in Fig. 9.

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

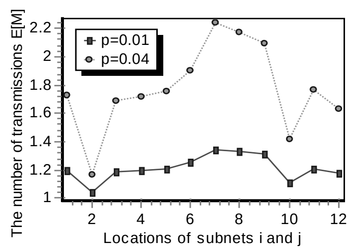

Fig. 9 The number of transmissions after the pruning of 2 subnets. The locations of subnets corresponding to the values of the x axis are shown in Table 2.

4. Optimal RR placements - A New Algorithm for RR routers 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 router) join the session at each router at a rate of λ_J (or λ_s). The initial network topology is shown in Fig. 1.

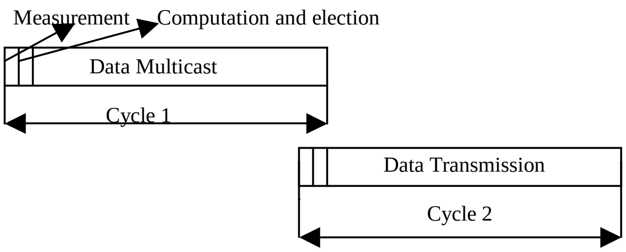

Fig. 10 A typical cycle of the 3-phase algorithm.

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 loss probability.

Worth noting in this paper, many routers may have embedded RR capability but this capability of recovering errors and losses 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. Here, the bandwidth E[C] is the total number of links affected by one multicast packet.

The purpose of the 3-phase algorithm is to optimally appoint a new router to become a RR, an appointment that takes place every cycle. The cycle consists of the measurement, computation and election, and multicast phases, and different cycles are pipelined as in Fig. 10. 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 pipeline operation, as seen in Fig. 10, 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. 1, 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 paper, 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

$$u_k = 2\sum_{k_i \in T_k} n_{k_i} + n_k \qquad (38)$$

where we sum up only the routers under router k. The u_k of node k can be recursively obtained from (38) through the use of the topology parameters of its children nodes.

$$u_k = \sum_{k_1 \in child(node.k)} u_{k_1} + e_k = n_k + \sum_{k_1 \in child(node.k)} (u_{k_1} - n_{k_1}) \qquad (39)$$

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 regards 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. 1 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:

$$E[C] = E[C(S)] + E[C(S_k)] = N(S)E[M(S)] + N(S_k)E[M(S_k)] \qquad (40)$$

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.

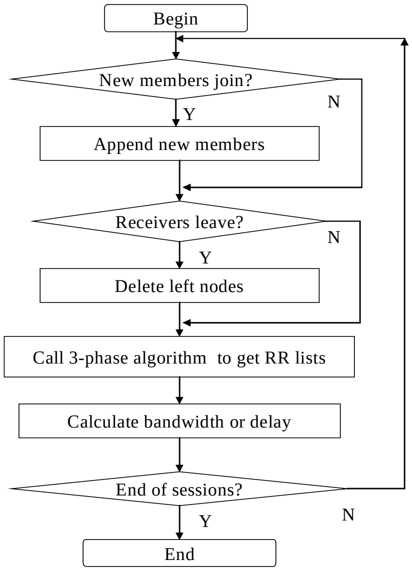

Fig. 11 The flowchart of dynamic networks.

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 $u_k = 2\sum_{k_i \in T_k} n_{k_i} + n_k$), 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 through

$$u = u_0 - u_k - 2n_k d_k \tag{41}$$

Substituting these parameters into (40), each router can calculate the total bandwidth E[C].

$$E[C] = (n_0 - n_k)[g(n_0 - n_k) + (u_0 - u_k - 2n_k d_k)p^2] + n_k[g(n_k) + u_k p^2] \tag{42}$$

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. It is possible that the number of self-appointed RRs will grow, cycle after cycle. However, increasing the length of cycle time (Fig. 10) 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.

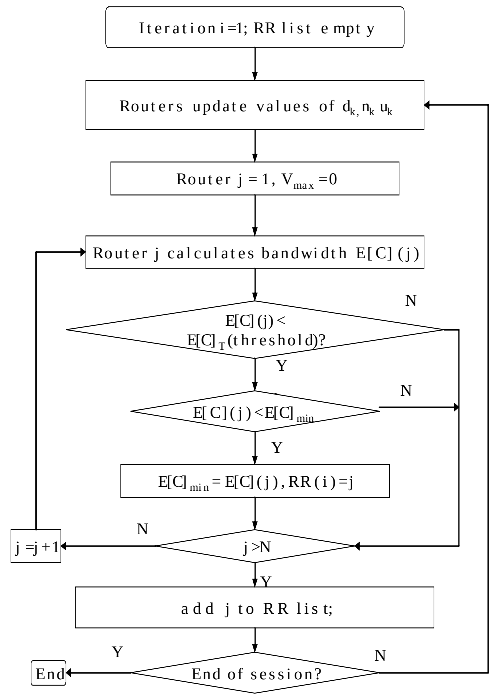

Fig. 12 The flowchart of the 3-phase algorithm.

Once a router becomes a RR, it will compute the value E[C] (42). In every cycle, if pruning causes the E[C] to become less than a certain low threshold, this RR will resign his rule. 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.

5. 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. Different topologies have different optimal RR locations. 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 Fig. 1 is node 1.

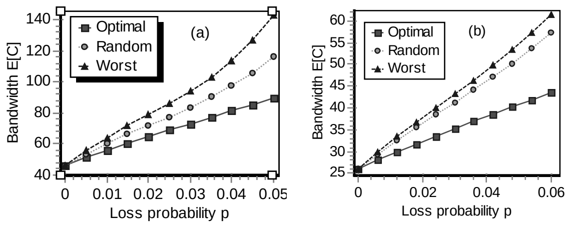

Fig. 13 A comparison of the bandwidth consumption for optimal, random, and the worst RR locations versus the loss probability p in the case of 1 RR appointment. (a) the network in Fig. 1. (b) the network after subnet 6 rooted at node 6 is pruned.

If RRs are not selected optimally, a lot of bandwidth will be wasted. In the following, we will 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 paper to the case where only one RR is appointed, where the sender selects the best RR out of many self-appointed ones. Fig. 13 provides a comparison of 1 RR. From Fig. 13, 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 (116-90)/90 = 29%.

Table 3. The optimal locations of RR in the case of the pruning of one subnet

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

Although the effect of receiver pruning is small in Fig. 6, the burst pruning of subnets has a significant influence on multicast performance in Fig. 8 and Fig. 9. 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 3 shows the optimal locations of the remaining multicast network in Fig. 1, 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 from node 1. 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. 13. 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.

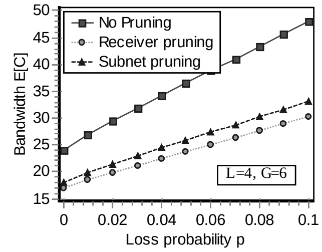

Fig.14 A comparison of the total bandwidth between receiver pruning and the subnet pruning for the topology in Fig. 2.

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.

We compare the results of the receiver pruning with 1 RR to the subnet pruning with 1 RR in the following. Fig.14 shows the results for the topology in Fig. 2, 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.

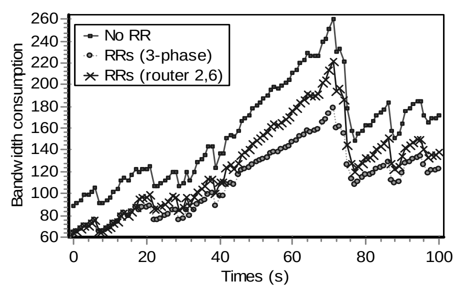

Fig. 15 The bandwidth performance of dynamic networks. $\lambda_P=0.2, \lambda_J=0.8$.

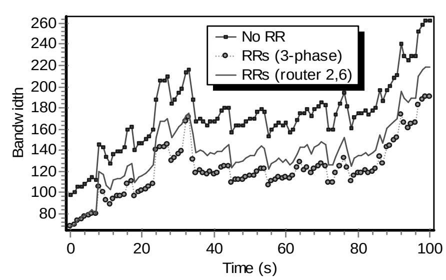

Fig. 16 The bandwidth performance of dynamic networks. $\lambda_P=0.2, \lambda_J=0.7$. $\lambda_S=0.05$. p=0.04

Some simulation results are shown in Fig. 15- Fig. 16 where λ_P is the probability that a receiver prunes from the multicast session and that 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 Fig. 15 and Fig. 16, we compare the bandwidth consumption for different RR locations. 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, previous RR locations may no longer be optimal. The RR locations given by the 3-phase algorithm always yield lower bandwidth consumption.

6. 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 bandwidth consumption increases with the increasing loss probability p and decreases with the increasing pruning probability λ. We have suggested a new algorithm to adapt the optimal RR location to the dynamic change in the network topology based on bandwidth consumption. We have found that the RR locations have great 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 will be wasted if RR is not optimally selected. We have also found that random pruning has less effect on multicast performance compared to burst pruning. Further investigation of the effects of multiple RR appointments, of the bandwidth consumption due to control packets of the 3-phase algorithm, and of the joining effects is underway.

References

[1] John C.Lin, Sanjoy Paul, “RMTP: A Reliable Multicast Transport Protocol”, *Proc. of IEEE INFOCOM*, San Francisco, USA,1996, pp.1414-1424.

[2] Athina.Markopoulou, Fouad.Tobagi, “Hierarchical Reliable Multicast: performance analysis and placement of proxies. *Proceedings of the Second International Workshop on Networked Group Communication (NGC 2000),* Palo Alto, USA, ACM Press, pp.27-35, Nov.2000.

[3] Pravin Bhagwat, Partho P.Mishra, Satish K.Tripathi, “Effect of Topology on Performance of Reliable Multicast Communication”, *Proc. of IEEE INFOCOM*. 1994.

[4] Joerg Nonnenmacher, Ernst W.Biersack, “Performance Modeling of Reliable Multicast Transmission”, *Proc. of IEEE INFOCOM,* Vol. 2, pp. 471 -479, 1997.

[5] MiKi Yamamoto, Makoto Yamaguchi, Takashi Hashimoto, Hiromasa Ikeda, “Performance Evaluation of Reliable Multicast Communication Protocol with Network Support”, *IEEE ICC*, pp,1736-1741,2000.

[6] Christos Papadopoulos, Guru Parulkar and George Varghese, “An Error Control Scheme for Large-Scale Multicast Applications”, *Proc. of IEEE INFOCOM,* San Francisco, USA,1998, pp.1188-1196.

[7] Martin S.Lacher, Joerg Nonnenmacher, and Ernst 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.

[8] 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.

[9] 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.

[10] Hwa-Chun Lin, Kuen-Feng Yang, “Placement of Repair Servers to Support Server-Based Reliable Multicast”, *IEEE ICC*,vol.16, 2001, pp.1802-1806.

[11] 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.

[12] Zuo Wen Wan, “Performance evaluation and optimization of reliable multicast”, *Ph.D dissertation*, Concordia University, 2004.

Ahmed K. Elhakeem received his Ph.D. degree from the Southern Methodist University, Dallas, TX, in 1979. He spent the next two years working as a Visiting Professor in Egypt, after which he moved to Ottawa, Canada, in 1982. He assumed research and teaching positions at Carleton and Manitoba Universities, and later moved to Concordia University, Montreal, Canada, where he has been a Professor in the Electrical and Computer Engineering Department since 1983. He has published numerous papers in IEEE and in international journals in the areas of spread spectrum and networking. He is a well-known expert in these areas and serves as a consultant to various companies. His current research interests include interconnected wireless LANs, Error correction for IP Multicast, wideband networks, switching architectures, CDMA networks, multi protocols, software Radios, and reconfigurable networks. He is a co-author of the book *Fundamentals of Telecommunications Networks* (New-York: Wiley, 1994). He has chaired and organized numerous technical sessions in IEEE conferences, and was the Technical Program Chairman for IEEE Montech ’86 in Montreal, Canada. More recently, he was the Key Guest Editor for four issues of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS on Code Division Multiple Access, namely CDMA I, II, III and IV, appearing in May ‘94, June ’94, October ’96, and December ’96. Dr. Elhakeem was the Communications chair of IEEE Montreal and the TCCC Rep. to ICC ’99. He served as an associate editor for the IEEE Communications Letters Journal (1996-1999). He is a professional engineer in Ontario and a senior member of IEEE. Dr. Elhakeem has served as a consultant to many companies, and to Communication Research Center and Industry Canada.