Turk J Elec Engin, VOL.11, NO.1 2003, © TÜBİTAK

Network Traffic Properties of Bimodal Multicast Protocol

Öznur ÖZKASAP
Koç University, Department of Computer Engineering,
34450 Sarıyer İstanbul-TURKEY
e-mail: oozkasap@ku.edu.tr
Mine ÇAĞLAR
Koç University, Department of Mathematics,
34450 Sarıyer İstanbul-TURKEY
e-mail: mcaglar@ku.edu.tr

Abstract

The popularity of large-scale distributed applications, such as videoconferencing, multimedia dissemination, electronic stock exchange and distributed cooperative work, has grown with the availability of high-speed networks and the expansion of the Internet. The key property of this type of applications is the need to distribute data among multiple participants together with an application-specific quality of service needs. This fact makes scalable multicast protocols an essential underlying communication structure. Although there exist several studies investigating the traffic characteristics of unicast communication, multicast traffic has not been examined extensively in previous studies. It is well known that the aggregate traffic properties of self-similarity and long-range dependence are ubiquitous in wide area networks and lead to adverse consequences in network performance. In this study, we analyze traffic characteristics of a novel scalable, reliable multicast protocol, Bimodal Multicast (Pbcast). In particular, our simulation studies demonstrate that epidemic approach of Bimodal Multicast generates short-range dependent traffic with low overhead traffic and transport delays. We elaborate on the protocol mechanisms as an underlying factor in our empirical results.

Key Words: *Multicast network traffic, self-similarity, long-range dependence, Bimodal Multicast (Pbcast), Scalable Reliable Multicast (SRM), epidemic communication.*

1. Introduction

It is well known that multicast transport protocols offering strong reliability guarantees such as atomicity, virtual synchrony, delivery ordering, and network-partitioning support have limitations in terms of scalability and throughput stability [1]. The main drawback is that in order to obtain strong reliability guarantees, costly protocols are used and the possibility of unstable or unpredictable performance under failure scenarios is accepted. Although the other class of protocols offering support for best-effort reliability in large-scale overcome message loss and failures, they do not guarantee end-to-end reliability. Common failure scenarios such as router overload and system-wide noise can cause these protocols to behave pathologically and hence

lead to negative protocol effects on network performance. Our approach in this article is to focus on both scalable and reliable multicast communication, which we analyze through traffic characteristics.

Many analyses of fine-grained measurements over the last decade have shown that network traffic is often bursty on a wide range of time scales with strong correlations across arbitrarily large time lags. These characteristics, called self-similarity and long-range dependence (LRD) respectively, imply significant queuing delays and degraded network performance. As a result, much of the recent research has focused on investigating the causes and consequences of traffic self-similarity and LRD. The pioneering work of Leland et al. [2] is based on the analysis of massive amounts of aggregate traffic traces from LAN. This has triggered three research streams: one dealing with further traffic characterization, another dealing with modeling issues, and the other concentrating on queuing implications and performance evaluation. In the characterization stream, several studies have verified the ubiquitous presence of self-similarity in a variety of networked environments such as LAN, WAN and ATM. Isolated traffic sources such as VBR video traffic and WWW traffic have been shown to possess self-similarity and LRD as source-intrinsic properties [3]. The research stream on the modeling of network traffic is based on matching the characteristics established by the analysis of traffic traces with the elements of a stochastic model. Incidentally, the collective efforts of researchers in this area have resulted in shedding light on the causes of self-similarity and LRD observed at the link level. See [4] for a review of such models and a specific construction based on compound Poisson packet generation. The research stream on performance evaluation examines the consequences of self-similarity and LRD. Most importantly, the queue lengths under self-similar traffic decay more slowly as compared to short-range dependent traffic. This has been shown analytically for various self-similar models in addition to simulation studies [3]. Another striking result verified with both approaches is that the utilization factor cannot be practically improved by enlarging buffers [5,6]. Instead, increasing link bandwidth has the effect of decreasing queuing delay more drastically under self-similar and long-range dependent traffic conditions.

The position of the network in the three layers of protocol stack with relation to network traffic has been illustrated in Figure 1. Research has revealed the nature of applications and user behavior as the main cause of statistical characteristics of traffic observed at the link layer while establishing the consequences of such characteristics on network performance. It has been shown that transport layer mechanisms are important components in translating heavy-tailed file size distributions at the application layer into link traffic self-similarity [6]. Since the application and user characteristics are static, in order to provide superior network performance, current traffic research has moved towards the analysis of intermediate layers.

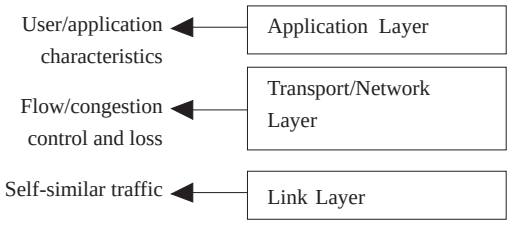

Figure 1. Protocol stack and its relationship to network traffic.

In this article, we analyze traffic characteristics of scalable multicast communication. We focus on a transport level scalable multicast protocol, namely Bimodal Multicast as a promising approach. It is compared to Scalable Reliable Multicast (SRM) protocol, which significantly differs from Bimodal Multicast in loss recovery and has similar mechanisms to TCP prevalent in the Internet. Bimodal Multicast based on epidemic principles in loss recovery emerges as both reliable and scalable, and provides remarkably stable

delivery output [1]. SRM is also scalable, but having best-effort reliability can be problematic in the presence of low levels of system-wide noise or by transient elevated rates of message loss [7,8]. For unicast traffic, in [9] TCP connection arrivals have been studied at the transport layer. However, in previous studies, large-scale multicast traffic has not been examined for LRD characteristics. Our initial study [10] and our further consideration in the present study investigating two multicast protocols show that when one approach does not intrinsically lead to long-range dependent traffic, the other does, under identical network conditions.

The article is organized as follows. In Section 2, Bimodal Multicast is reviewed in comparison to other scalable reliable multicast protocols. Section 3 describes our simulation settings and method of traffic analysis. We report the simulation results in Section 4. The significance of these results in terms of LRD is discussed in Section 5. Finally in Section 6, we state our overall conclusions.

2. Multicast Transport Protocols

2.1. Bimodal Multicast

Bimodal Multicast [1] is a novel option in the spectrum of multicast protocols that is inspired by prior work on epidemic protocols [11], Muse protocol for network news distribution [12], and the lazy transactional replication method [13]. Bimodal Multicast is based on an epidemic loss recovery mechanism. It has been shown to exhibit stable throughput under failure scenarios that are common on real large-scale networks [1]. In contrast, this kind of behavior can cause other reliable multicast protocols to exhibit unstable throughput. Bimodal Multicast consists of two sub-protocols, namely an optimistic dissemination protocol and a two-phase anti-entropy protocol.

Optimistic dissemination: This sub-protocol is a best-effort, hierarchical multicast used to efficiently deliver a multicast message to its destinations. This phase is unreliable and does not attempt to recover a possible message loss. If IP multicast is available in the underlying system, it can be used for this purpose. For instance, the protocol model implemented on the ns-2 network simulator [14] in this study uses IP multicast. Otherwise, a randomized dissemination protocol can play this role.

Two-phase anti-entropy: The second stage of Bimodal Multicast is responsible for message loss recovery. It is based on an anti-entropy protocol that detects and corrects inconsistencies in a system by continuous gossiping. The two-phase anti-entropy protocol progresses through unsynchronized rounds. In each round:

- Every group member randomly selects another group member and sends a digest of its message history. This is called a 'gossip message'.
- The receiving group member compares the digest with its own message history. Then, if it is lacking a message, it requests the message from the gossiping process. This message is called 'solicitation', or retransmission request.
- Upon receiving the solicitation, the gossiping process retransmits the requested message to the process sending this request.

Protocol execution: Figure 2 illustrates the execution of Bimodal Multicast. A, B, C and D are group members, and the time advances from top to bottom. A dashed arrow in the figure denotes a message loss. First, multicast messages M0, M1 and M2 are transmitted unreliably by the dissemination protocol. Because of a process or communication failure, process C fails to receive message M0, and process D fails to

receive M1. Then, the anti-entropy protocol executes. Each process selects another one randomly, and sends its message history digest. Upon receiving a gossip message from process B, process C discovers that it is missing M0 and requests a retransmission from B, and recovers this message loss. Because of the randomness in selecting a process to gossip, a process may not receive a gossip message in a given round. For example, process D does not detect its message loss until the next anti-entropy round. The figure simplifies the execution by showing that the protocol alternates between dissemination and anti-entropy stages. However, in practice, these stages run concurrently.

One of the differences of Bimodal Multicast's anti-entropy protocol from the other gossip protocols is that during message loss recovery, it gives priority to the recent messages. If a process detects that it has lost some messages, it requests retransmissions in reverse order: most recent first. If a message becomes old enough, the protocol gives up and marks the message as lost. By using this mechanism, the protocol avoids failure scenarios where processes suffer transient failures and are unable to catch up with the rest of the system. One of the drawbacks of traditional gossip protocols is that such a failure scenario can slow down the system by causing processes' message buffers to fill up. The duration of each round in the anti-entropy protocol is set to be larger than the typical round-trip time for an RPC over the communication links. The simulations conducted in this study use a round duration of 100 ms. Processes keep buffers for storing data messages that have been received from members of the group. Messages from each sender are delivered in FIFO order to the application. After a process receives a message, it continues to gossip about the message for a fixed number of rounds. Then, the message is garbage collected.

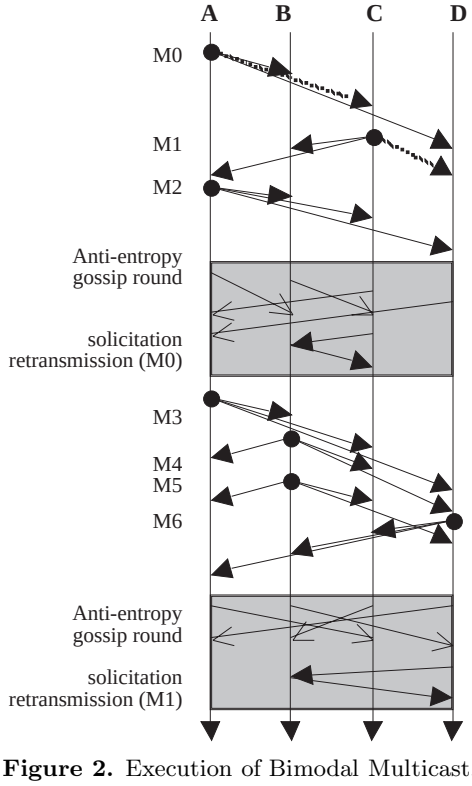

Figure 2. Execution of Bimodal Multicast.

2.2. Other Scalable Reliable Multicast Protocols

Other scalable reliable multicast protocols focus on best-effort reliability in large-scale systems. These protocols overcome message loss and failures, but they do not guarantee end-to-end reliability. For instance,

group members may not have a consistent knowledge of group membership, or a member may leave the group without informing the others. When the message loss probability is very low or uncommon, they can give a very high degree of reliability. However, failure scenarios such as router overload and system-wide noise, which are known to be common in Internet protocols, can cause these protocols to behave pathologically [15,16]. Example systems are the Internet Muse protocol for network news distribution [12], the Scalable Reliable Multicast (SRM) protocol [17], the Pragmatic General Multicast (PGM) protocol [18], and the Reliable Message Transfer Protocol (RMTP) [19].

SRM [17] is a well-known reliable multicast protocol that was first developed to support wb, a distributed whiteboard application. The protocol is inspired by the principles of IP multicasting, application level framing (ALF), and the TCP/IP architecture design. The protocol necessitates the basic IP delivery model and forms reliability on an end-to-end basis. There is no need for modification on the underlying IP network. Similar to TCP that adaptively sets timers or congestion control windows, SRM algorithms dynamically regulate their control parameters based on the observed performance within a session. Unlike Bimodal Multicast, which provides FIFO delivery ordering, SRM does not provide an ordered delivery of messages. The protocol aims to scale well both to large networks and sessions, and exploits a receiver-based reliability mechanism.

In SRM, each group member multicasts low-rate, periodic session messages that report the sequence number state for active sources, or the highest sequence number received from every member. As well as the reception state, the session messages also contain timestamps that are used to estimate the distance from each member to every other. Members utilize session messages in SRM to determine the current participants of the session. In addition to state exchange, receivers use the session messages to estimate the one-way distance between nodes. The session packet timestamps are used to estimate the host-to-host distances needed by loss recovery mechanisms. The random delay before sending a request or repair packet is a function of that member's distance in seconds from the node that triggered the request or repair. Repair requests and retransmissions are multicast to the whole group. A lost packet ideally triggers only a single request from a host just downstream of the point of failure.

Loss recovery: Multicast group members detect lost messages by means of gaps in the sequence number. In order to detect losses of the last messages that are sent, SRM uses session messages. When a group member A detects a message loss, it schedules a retransmission request, and sets a request timer to a value from the uniform distribution on

$$[C_1 * d_{S,A}, (C_1 + C_2) * d_{S,A}] \text{ seconds}$$

where $\mathrm{d}_{S,A}$ is member A's estimate of the one-way delay to the original source S of the missing data and C_1, C_2 are request timer parameters. If a member receives a request for the missing data before its own request timer for that data expires, then the member resets its request timer. When a group member B receives a request from A for a data message that B has a copy, B sets a repair timer to a value from the uniform distribution on

$$[D_1 * d_{A,B}, (D_1 + D_2) * d_{A,B}] \text{ seconds}$$

where $\mathrm{d}_{A,B}$ is the B's estimate of the one-way delay to A, and D_1, D_2 are repair timer parameters. If B receives a repair for the missing data before its repair timer expires, then B cancels its repair timer. As discussed in [17], there is no single setting for the timer parameters that gives optimal performance for all topologies, session memberships and loss patterns. When it is desirable to optimize the tradeoff between

delay and the number of duplicate requests and repairs, an adaptive algorithm can be used. Adaptive SRM adjusts the timer parameters C_1, C_2, D_1, and D_2 in response to the past behavior of the loss recovery algorithms.

Recent studies [1,7,8] have shown that, for the SRM protocol, random packet loss can trigger high rates of overhead messages. In addition, this overhead grows with the size of the system. Related to this scalability problem, some of our previous simulations have explored the behavior of Bimodal Multicast and SRM protocol versions on a large-scale network with high-speed data transfer. In these simulations, we construct large-scale tree topologies consisting of 1000 nodes. Up to 100 of the 1000 nodes were randomly chosen to be group members. We set the message loss rate to 0.1% on each link with the sender located at the root node injecting 100 210-byte multicast messages per second. One set of results, taken from [20], for the background overhead of each protocol in the form of repair message traffic is shown in Figure 3. We include error bars showing minimum and maximum values recorded over the set of runs, using different seeds for the random number generator. The results demonstrate that as the network and process group size scale up, the number of control messages received by group members during loss recovery increases linearly for SRM protocols, an effect previously reported [7,8]. These costs remain almost constant for Bimodal Multicast versions (in the graphs these are labeled as Pbcast and Pbcast-ipmc for short). Pbcast-ipmc is the version of Bimodal Multicast that uses IP multicast for message repairs during loss recovery. Compared to the basic Pbcast, Pbcast-ipmc has a slightly lower overhead in the form of request messages. If multiple receivers miss a message, Pbcast-ipmc increases the probability of rapid convergence during loss recovery.

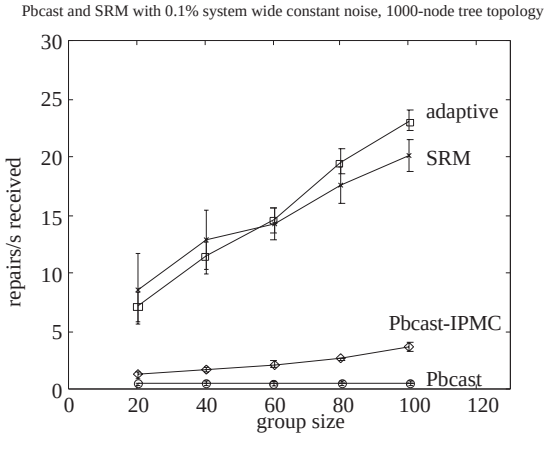

Figure 3. Overhead in the form of repairs per second for Bimodal Multicast and SRM, 1000-node tree topologies with 0.1% system-wide drop rate.

3. Simulation Settings and Method of Analysis

Simulation methods allow gaining control over the parts of the network and lead to a better understanding of protocols. For instance, in a simulation model, link loss probabilities can be set and maintained easily, and several network topologies can be constructed. Many process group applications and scenarios can be built on top of these settings. Our simulation study uses the ns-2 network simulator [14] to model network and protocol behavior. Our The implementation for Bimodal Multicast developed over ns-2 is used in this study [1]. Implementation of SRM available in ns-2 is used for the comparison of some results. For that case, we compare SRM and Bimodal Multicast in the same simulation settings.

Our approach is to consider the delay of packets over the network. Recently, packet delay measurements over the Internet are used to trace the conditions of the network between an origin and destination pair [21-23]. In our simulations, such measurements represent traffic at the transport level. The randomness in delays is due to traffic generated as a result of noise and the control and recovery mechanisms of the transport protocol.

In our simulations, the delay of a data message at a process is measured as the delay between the time that a message is initially multicast to the group by data source and the time the message is first delivered by the process. There are basically two cases:

• The message is not exposed to failure and is delivered at the end of best-effort transmission,

• The message drops because of a failure in the network, and error recovery mechanism takes part in recovering the message and makes sure it reaches the intended destination processes.

A process can also receive duplicate copies of a message, but in our analysis we do not consider duplicate receipts and just use the first receipt time of a message to calculate its delay. Since Bimodal Multicast protocol provides FIFO ordered delivery, in some of our simulations we analyze its delay distribution in two forms: Delay distribution at the node level and delay distribution after FIFO ordering. In contrast, since SRM does not guarantee ordered delivery, we only analyzed its delay distribution at the node level.

3.1. Topologies

Our initial simulations are performed on a 500-node tree topology where a randomly selected 300 nodes are group members. The sender located at the root node sends at a rate of 0.01 (100 multicast messages per second), and on all network links there is a system-wide drop rate of 1%. An example application for this scenario would be a multicast-based distance education session in a wide area setting consisting of multiple local campus networks. In this case, one server would multicast the content to most of the nodes in the network. Another application could be an air traffic control system where the controller consoles are replicated to achieve fault tolerance.

Other scenarios might be LANs connected by long distance links and networks where routers with limited bandwidth connect group members. Such configurations are common in today's networks as well. With these in mind, our second scenario simulates a clustered network with 80 nodes where it consists of two 40-node fully connected clusters, and a single link connects those clusters. All nodes are the members of the multicast group where there is one sender. There is a 1% intracluster drop rate formed in both clusters, and a varying high drop rate is injected on the link connecting the clusters that make it behave like a bottleneck link. A sample cluster topology with 18 nodes is shown in Figure 4b. We vary the operating parameters of the multicast message rate of the sender and intracluster drop rate.

A dense transit stub topology is considered for various group sizes. The Internet can be viewed as a collection of interconnected routing domains where each domain can be classified as either a stub or a transit domain. Stub domains correspond to interconnected LANs and the transit domains model WAN or MANs. We use a gt-itm [24] topology generator for producing transit-stub topologies. A sample transit-stub topology with 128 nodes is shown in Figure 4a. We obtain our results from three runs of simulations, each run consisting of a sequence of at least $2^{15} = 32768$ multicast data messages transmitted by the sender at a rate 50 messages (each with size 210 bytes) per second. We do only three replications, but the topology is generated randomly and differs in each run.

Another scenario consists of large-scale transit-stub topologies (that approximate the structure of the Internet) with 1500 nodes where the sender is located on a central node and receivers are located at randomly

selected nodes on the network. A certain link noise in the form of a drop rate is set on every link forming a system-wide noise. We vary three operating parameters, namely group size, multicast message rate of the sender, and system-wide drop rate. This scenario primarily focuses on the impact of randomized message loss over traffic generated by the Bimodal Multicast. We obtain our results from several runs of ns.

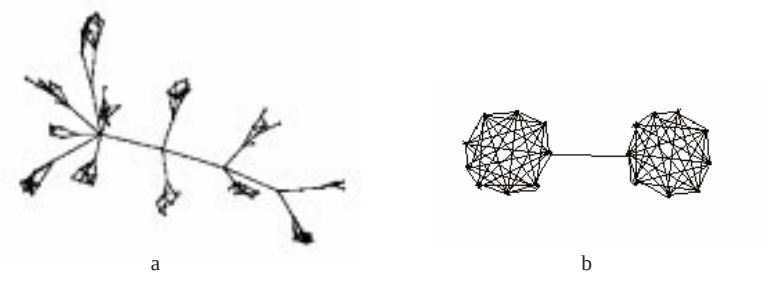

Figure 4. Sample topologies (a) Transit-stub (b) Clusters.

3.2. Hurst Parameter of Message Delay

LRD is defined as the slow, power-law like decrease of the autocovariance function γ of a stationary sequence at large lags k, given by $\gamma(\mathrm{k}) \sim \quad c_\gamma k^{2H-2}$, with $0.5 < \mathrm{H} < 1$. The parameter H is called the Hurst parameter, whose value represents the magnitude of the correlation. The value $H = 0.5$ corresponds to an independent sequence as in Gaussian white noise, and the larger the H, the slower the decay of the function γ at large lags. Hence, we say that there exists more LRD as H increases.

The long-term correlations in the traffic can be characterized through delay process among others [25,22]. There exist traffic models for workload, which corresponds to the message arrival process in multicast traffic studied in the present paper. It is shown in [25] that the Hurst parameter computed from the workload process and the delay process are in agreement. Along these lines, [22] studied LRD through packet delay traces in Internet traffic. Our approach in this paper will be similar. We will concentrate on the delay data obtained from the simulations of multicast message traffic and compute the Hurst parameter H from these data. The delay sequence is stationary as the transient part of the simulations disregarded. Here, the lag k of the function γ has the unit of a number of messages. We estimate H using the wavelet estimation method, as will be described next.

3.3. Wavelet Estimation Method

The wavelet estimation method is known to have very good properties for estimating the Hurst parameter H as opposed to variance-time estimation and other heuristic methods [25-27]. It is an unbiased, consistent, and also computationally efficient method of estimation.

We apply the wavelet estimation method as given in [26], using Daubechies wavelets with two vanishing moments. Let $d(j,k),\ k = 1,\ldots,n_j,\ j = 1,\ldots,J$ denote the 'details' obtained by the discrete wavelet transform of the sequence of message delays $\{x_k,\ k = 1,...,N\}$, where J is such that $2^{J+1} \leq N \leq 2^{J+2}$, and n_j is the number of coefficients available at octave j. The statistic central to the method is given by

$$\mu_j = \frac{1}{n_j}\sum_{k=1}^{n_j} d^2(j,k), \quad j = 1,\ldots,J.$$

Let c_f denote the coefficient in the spectrum of the delay sequence. That is, it is the counterpart of c_γ of the autocovariance function in the spectrum. Then, the Hurst parameter H and the coefficient c_f are estimated through a weighted linear regression of

$$y_j = \log_2(\mu_j) - g_j$$

over $j = j_1, \ldots, j_2$, where j_1 and j_2 are the scales relevant for LRD. Typically, these are the larger scales. The constant

$$g_j = E(\log_2(\mu_j)) - j(2H-1) - \log_2(c_f C)$$

is introduced to ensure that the fundamental hypothesis of regression holds (with C a constant that depends on H). Then, the slope α of the regression line is $(2H-1)$ and the estimate of H is given by

$$\hat{H} = \frac{\alpha + 1}{2}$$

which is unbiased and consistent.

4. Simulation Results

In this section, we report our initial results given in [10] as well as our more detailed simulations of Bimodal Multicast traffic in comparison to SRM. First, a large-scale tree topology is considered where the group size is also large. In another setting, moderate-size clusters connected by a noisy link are simulated. Then, the two protocols are compared in dense transit-stub topology where both the network and group sizes are the same. In view of these results, the effect of a large scale network is analyzed only for Bimodal Multicast in the sparse case as well.

4.1. Randomized Message Loss

In this part of the study, data were gathered on a 500-node tree topology where a randomly selected 300 nodes are group members. The sender located at the root node sends with at a rate of 0.01 (100 multicast messages per second), and on all network links there is a 1% system-wide drop with rate.

Figure 5 shows delay histograms of the protocols for this scenario where the x-axis is the delay in seconds (in increments of 0.1 ms intervals) and the y-axis is the percentage of occurrences. Figures 5a and 5c are the node level delay distribution of Pbcast, and SRM respectively. A typical receiver delivers messages with lower delays when Pbcast protocol is used for group communication. As shown in Figure 5c, SRM has a large tail with a maximum observed delay of nearly 800 ms, and a group of packets delivered at around 400 ms. Overall, SRM has a significant number of packets delivered during the first 100 ms and a second broad distribution containing almost 5% of packets, which arrive with delays of between 300 ms and 800 ms. Notice that the basic SRM distribution is not as tight as the unordered Pbcast distribution, which has more than 90% of its packets arriving at the lowest possible delays. In the case of Pbcast, around 2% of packets are delayed and arrive in the period between 200 ms and 300 ms, with no larger delays observed.

We also investigate message delays of Pbcast after FIFO ordering is accomplished. In that case, depending on the message loss rate experienced by the receiver, some percentage of messages are delivered

with higher delays since messages not in order are buffered prior to delivery in order to guarantee FIFO ordering property (Figure 5b). These higher delays reflect the cost of waiting for messages to be retransmitted and placing them into the correct delivery order.

These results are important at least in settings where the steady delivery of data is required by the application. We observe that as SRM is scaled to larger groups, steadiness of throughput can be expected to degrade. We experimented with a variety of noise levels, and obtained similar results, although the actual number of delayed packets obviously depends on the level of noise in the system.

We have found the Hurst parameter H for Pbcast to be 0.54 whereas for SRM to be 0.65. In this case H being close to 0.5, Pbcast performs very well with no LRD implication. On the other hand, the delays for SRM are long-range dependent, although H is not very high. Hence, Pbcast performs better. Our estimate of H for Pbcast after FIFO is 0.72, which qualifies the delay in this case to be long-range dependent. However, note that this result reflects the effect of the application level, namely the requirement of FIFO ordering, and not the effect of the transport layer.

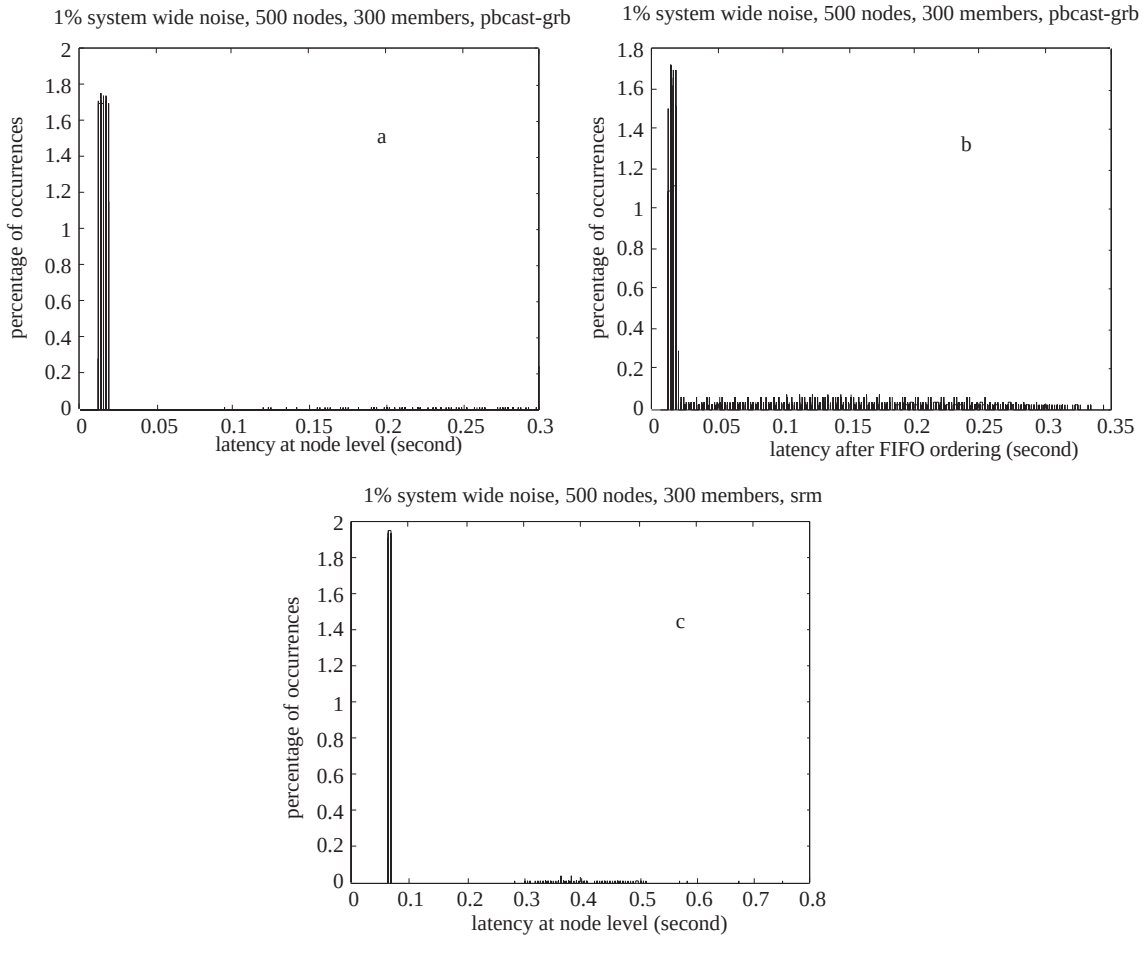

Figure 5. Delay histograms. a) Pbcast at node level, b) Pbcast after FIFO ordering, c) SRM.

4.2. Clusters connected by a noisy link

In the previous subsection, we focused on the impact of randomized message loss on the performance of Pbcast and SRM protocols. In this scenario, we simulate a clustered network with 80 nodes. The network

consists of two 40-node fully connected clusters, and a single link connects those clusters where all nodes form an 80-member process group. The sender is located on the first cluster, and it generates 100 multicast messages per second. There is 1% intra-cluster noise formed in both clusters, and a high drop rate of 20%, 40% or 50%, is injected into the link connecting the clusters. This inter-cluster noise corresponds to the probability that a message transmitted from the first cluster to the second will drop and hence get lost. We then explore the delay characteristics of a receiver on the second cluster.

In this configuration, both SRM and adaptive SRM deliver some messages with very long delays of many seconds. In the adaptive case particularly, about 5% of all data messages are delayed by 5 s or more before delivery. On the other hand, Pbcast delivers all data messages within 1 s and hence can be seen as offering relatively steady data throughput in networks with this configuration.

Table 1. Hurst Parameter of Delay for Clustered Network.

Drop	Pbcast		SRM	
Rate	before FIFO	after FIFO	non-adaptive	adaptive
20%	0.54	0.63	0.52	0.55
40%	0.49	0.61	0.66	0.59
50%	0.54	0.61	0.55	0.65

Table 1 gives the estimates of the Hurst parameter H for the three different noise rates and the two protocols. We see that H is very low for Pbcast before ordering at around 0.5 for all levels of drop rate. In these cases, the delays have very weak dependence among each other. As one would expect, FIFO ordering has an implication towards longer and more correlated delays over the network. This fact is demonstrated with higher values of H, around 0.6, for Pbcast after ordering. On the other hand, the value of H for both adaptive and non-adaptive SRM protocol varies from around 0.5 to 0.65, again with no specific pattern with the noise level. These values leading to only moderate LRD characterization, as in the case of Pbcast after FIFO, do not have adverse implications on the network performance. However, even non-adaptive SRM could have long-range dependence, as a protocol, while Pbcast before FIFO does not.

The histograms obtained from delay data for clusters connected by a noisy link support the LRD analysis. The Pbcast delays are concentrated around low values (all less than 1 s) and the histograms look like normal and exponential distributions and/or their mixtures. Hence, the tails of the histograms decay exponentially. As an example, delay distributions with a 50% drop rate are given in Figure 6. For Pbcast before FIFO, an exponential distribution fits well, with a mean 0.18 s, where H was found to be 0.54. For Pbcast after FIFO, a normal distribution fits well with a mean of 0.43 s, where H was found to be 0.61. In contrast, for the distributions of SRM delay the tails are prominent. Although an exponential distribution fits well for non-adaptive SRM delay as in Figure 6c, the mean delay, 1.22 s, is higher than that for Pbcast even after FIFO, and the distribution has a long right tail. For this case H was found to be 0.55. The adaptive SRM was found to be long-range dependent with H equal to 0.65. This is in agreement with the corresponding distribution in Figure 6d, which has a long right tail with very large observations and amean of 1 s. A Pareto distribution, which is quite common in LRD, fits well in this case.

4.3. Dense Transit-Stub Topology

In the view of previous results, we did a more detailed analysis in this subsection. Initially, several independent runs of ns-2 were obtained to observe the effect of randomness. For the statistical precision of our results for long-range dependence, each run lasts at least $2^{15} = 32768$ messages. With such a long sequence,

independent runs with different seeds showed almost no random variation in the estimated Hurst parameters and other statistics of performance. That is why we have chosen to generate three independent random topologies for each group size and report their average statistics here. This has introduced randomness in our experiments while reinforcing our results as all topologies for a fixed group size showed similar performance characteristics within some variation.

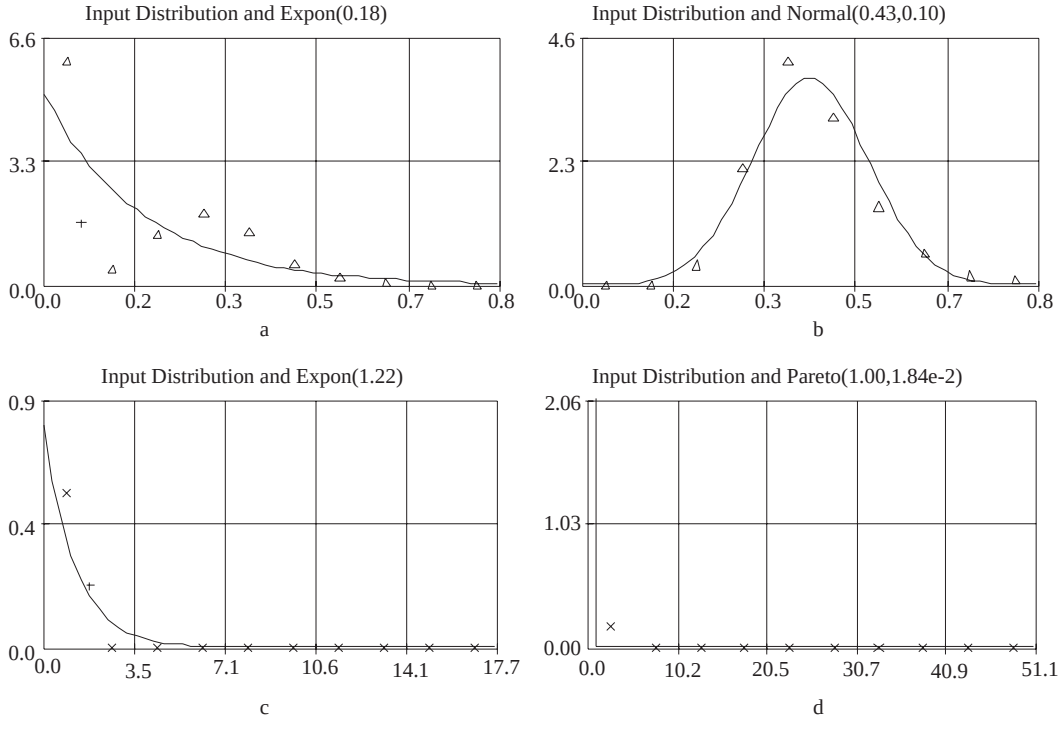

Figure 6. Delay distributions with 50% noise rate (a) Pbcast before FIFO, (b) Pbcast after FIFO, (c) non-adaptive SRM, (d) adaptive SRM.

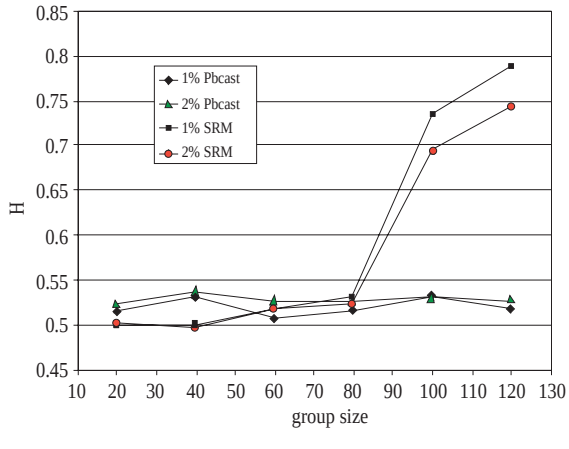

Figure 7. Hurst parameter versus group size for Bimodal Multicast and SRM at two levels of system-wide drop rate.

The most significant comparative result is displayed in Figure 7. We have started with small groups of size 20 and went up to 120 all on a transit-stub topology. The Hurst parameter estimates are given in Figure 7 for Bimodal Multicast and SRM at two levels of system-wide drop rate, namely 1% and 2% . Both protocols behave similarly up to group size 80; they generate short-range dependent traffic with H values around 0.5. When the group size increases to 100 or more, SRM traffic shows long-range dependence with H values statistically significantly greater than 0.7. Bimodal Multicast continues to produce short-range dependent traffic for groups of size 100 and 120. To demonstrate this pattern with respect to bigger group sizes, the scaling diagram is given in Figure 8a for Bimodal Multicast and (b) for SRM. The trivial scaling for Bimodal Multicast is apparent whereas the scaling for SRM indicates $H > 0.5$. In addition to the Hurst parameter, which indicates the strength of correlations across time, the mean of delay is also strikingly different for the two protocols. The mean delay for Bimodal Multicast is in the order of 0.03 s, whereas it is in the order of 0.70 s for SRM.

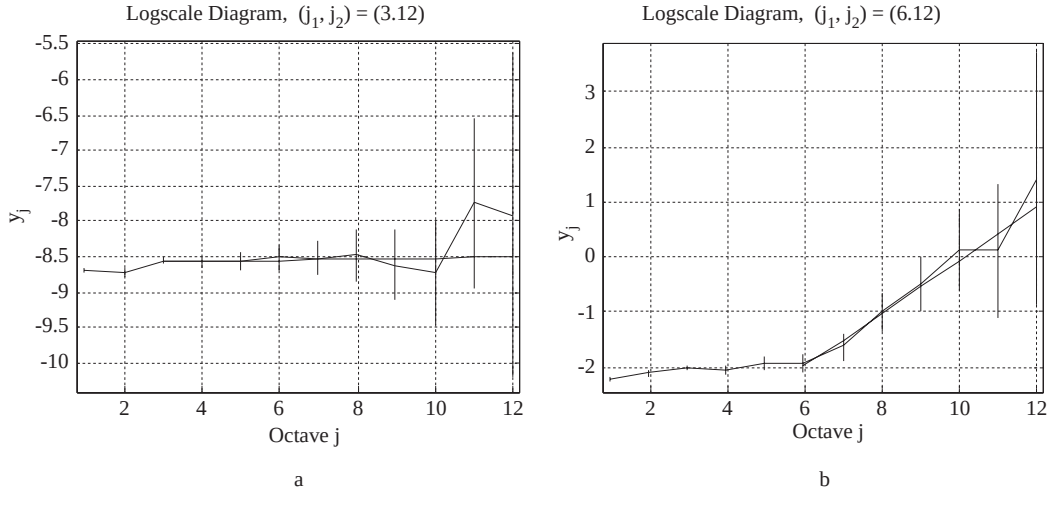

Figure 8. Scaling diagram used for the estimation of H obtained as a result of the wavelet transform of the delay sequence for the farthest receiver in a group of size 120 with 1% system-wide drop rate in the case of (a) Bimodal Multicast and (b) SRM.

Summary statistics of our simulations for large group sizes 60 to 120 are given in Tables 2 and 3. Overall, we see that the interarrival distribution is approximately normally distributed when the network is not pressured. This is true for Bimodal Multicast in all cases, and in smaller size groups for SRM. In the SRM case, as the group size increases, the distribution becomes right-skewed (long right tail) and an exponential distribution fits well. In the tables, the parameters given for the normal distribution are the mean and the standard deviation, and the parameter of the exponential distribution is the mean. The interarrival distributions reflect the performance implications of the traffic patterns. The interarrival distribution of the long-range dependent sequence for SRM is found to be exponentially distributed, whereas it is found to be normally distributed for the corresponding Bimodal Multicast traffic, which is short-range dependent. In Tables 2 and 3, the difference in the standard deviations of normal and exponential distributions is remarkable although they may have the same mean. Recall that the mean and the standard deviation of exponential distribution are the same. There is a similar difference for the throughputs, which is not documented here. Although the mean throughput is the same, the variance is significantly smaller for Bimodal Multicast.

Table 2. Hurst Parameter and Performance Measures on Transit-stub for Bimodal Multicast.

Parameters				Performance		
Drop rate	Message rate (msgs/s)	Group size	H and 95% CI	Loss ratio	Throughput (msgs/s)	Interarrival Distribution
1%	50	60	0.506 [0.500,0.513]	0.0019%	49.9981	N(0.02,0.00555)
1%	50	80	0.514 [0.511,0.518]	0.0019%	49.9986	N(0.02,0.00620)
1%	50	100	0.531 [0.529,0.533]	0.0010%	49.9981	N(0.02,0.00620)
1%	50	120	0.517 [0.498,0.536]	0.0029%	49.9976	N(0.02,0.00619)
2%	50	60	0.525 [0.521,0.530]	0.0086%	49.9948	N(0.02,0.00793)
2%	50	80	0.524 [0.516,0.532]	0.0229%	49.9876	N(0.02,0.00880)
2%	50	100	0.530 [0.519,0.541]	0.0010%	49.9938	N(0.02,0.00849)
2%	50	120	0.526 [0.517,0.535]	0.0210%	49.9871	N(0.02,0.00880)

The scalability of Bimodal Multicast is remarkable, only at a negligible cost in reliability. The throughput decreases and the variance of the interarrival increases only slightly as the group size increases and/or the drop rate increases. The Hurst parameter, on the other hand, is very stable in response to the doubling of the drop rate or the increase of the group size. Bimodal Multicast provides stable throughput in the sense that it has smaller variance than SRM. On the other hand, SRM makes the utmost effort reliability as no losses have been encountered in our simulations. This comes at a cost of longer delays, slightly lower throughput than Bimodal Multicast (significantly lower for $N = 120$), more variable interarrivals and most, importantly, self-similar traffic patterns.

Table 3. Hurst Parameter and Performance Measures on Transit-stub for SRM.

Parameters				Performance	
Drop rate	Message rate (msgs/s)	Group size	H and 95% CI	Throughput (msgs/s)	Interarrival Distribution
1%	50	60	0.517 [0.511,0.523]	49.9990	N(0.02,0.00744)
1%	50	80	0.531 [0.527,0.535]	49.9971	N(0.02,0.0152)
1%	50	100	0.734 [0.702,0.765]	49.9962	Exp(0.02)
1%	50	120	0.787 [0.676,0.792]	49.9962	Exp(0.02)
2%	50	60	0.516 [0.514,0.518]	49.9986	N(0.02,0.0117)
2%	50	80	0.522 [0.516,0.528]	49.9981	Exp(0.02)
2%	50	100	0.695 [0.653,0.738]	49.99381	Exp(0.02)
2%	50	120	0.742 [0.676,0.808]	49.9251	Exp(0.02)

4.4. Bimodal Multicast at Large-scale Sparse Topology

In this subsection, Bimodal Multicast is analyzed in detail for a much larger network. As shown in Table 4, the H values remain essentially around 0.5 even if we increase the drop rate or the group size. In the lower 1% drop rate, H increases slightly due to an increase in the transmission rate. This is slightly more pronounced in the 10% drop rate, where we have an increase to values around 0.55 (0.543 and 0.577) with increased message rate. The protocol is scalable as there is no significant increase due to group size and also network size when compared to the previous subsection.

Loss ratio increases with the drop rate as expected. Since the noise is system wide, an increase in the group size is also effective as the number of hops increases between the sender and the receiver. The interarrival distributions are normal or exponential as expected for such low H values. The mean interarrival times increase and the throughputs decrease slightly as the drop rate or the group size increases.

Table 4. Hurst Parameter and Performance Measures on Transit-stub.

Parameters				Performance		
Drop rate	Message rate (msgs/s)	Group size	H and 95% CI	Loss ratio	Throughput (msgs/s)	Interarrival Distribution
1%	10	10	0.514 [0.508, 0.521]	0.0012%	9.9998	Normal (0.100003, 0.018)
1%	100	10	0.525 [0.518, 0.532]	0.0003%	99.9863	Normal (0.010002, 0.002)
1%	10	50	0.503 [0.496, 0.510]	0.023%	9.9973	Normal (0.10003, 0.028)
1%	100	50	0.516 [0.509, 0.523]	0.0012%	99.9806	Normal (0.010004, 0.003)
10%	10	10	0.511 [0.507, 0.515]	0.209%	9.9787	Exp (0.1002)
10%	100	10	0.577 [0.573, 0.581]	0.049%	99.9253	Exp (0.01001)
10%	10	50	0.523 [0.517, 0.528]	3.87%	9.6127	Exp (0.104)
10%	100	50	0.543 [0.537, 0.549]	3.60%	96.3484	Exp (0.0104)

5. Discussion on Long-Range Dependence

The present article focuses on traffic properties of Bimodal Multicast through several simulation scenarios. More detailed analysis on the comparison of Bimodal Multicast with SRM is given in [28]. In particular, the marginal delay distribution has been analyzed for Bimodal Multicast [28], which complements the earlier results in [1]. We have shown that the delay distribution decays exponentially fast as expected from the Markov property of the epidemic mechanism of Bimodal Multicast. Such behavior can be modeled through an appropriate chain-binomial framework [29]. In [28], the intrinsic relation of transport protocol mechanisms to traffic characteristics is studied. The real MBone traces are also examined in comparison to unicast TCP and UDP traffic as a motivation for the study of the transport layer.

We consider a given message multicast to all group members from a sender. In the worst case, it is possible that none of the other group members receive the multicast. No matter how many processes receive the message, it will be repaired through the epidemic mechanism of the protocol. Let N denote the group size and R_t denote the number of receivers that have not received the message at round t. In the epidemic terminology, these are susceptible processes, and equivalently $N-R_t$ corresponds to the number of infectious processes. In the worst case, R_0 would be N-1 and only the sender would have the original message. At each round, the probability of infection depends on the infectious processes present at that time. Let p denote the probability that a given susceptible process receives a gossip and the following retransmitted message from a given infectious process successfully. Then $q = 1 - p$ is the probability of failure of an infection by that infectious process. A plausible assumption is that processes become infected independent of each other, since each process sends a gossip message to a randomly chosen group member. Also with the assumption that message loss occurs independently for each process, it is possible to evaluate q. Let ε be an upper bound for the probability of message loss for each pair of processes in the network. On the other hand, the probability that a given infectious process gossips to a given susceptible process is $\beta = f/N$ where f is the fanout, that is, the number of processes a process gossips to at each round. For a successful retransmission of the data message, all of the gossip, request and retransmission messages associated with the recovery process must be transmitted successfully. It follows that an upper bound for q is $\beta(1-\varepsilon)^3$, which we use as a pessimistic value for q. Hence, the probability that none of the infectious processes at round t infect a susceptible process is q^{N-R_t}. Therefore, the number of susceptible processes R_{t+1} in the next round is binomially distributed with parameters R_t and q^{N-R_t}. That is, R is a Markov chain with a lower triangular transition matrix and state space $\{0, 1, \ldots, N\text{-}1\}$ where 0 is the absorbing state, in the worst case when

$R_0 = N$-1. In general, it is conditionally a Markov chain on $\{0, 1, \ldots, R_0\}$ given R_0.

By demonstrating the relationship of the delay D of a retransmitted message to the expected values of R_t, t= 1,2, ..., we can show that

$$\bar{F}(t) \equiv P\{D > t\} \leq q^t, \quad t = 1, 2, \ldots$$

Hence, the tail of the marginal delay distribution decays faster than a geometric distribution with parameter q. Geometric distribution is the discrete analogue of exponential distribution, which is indeed found to be a good fit to the empirical distributions of the delay of retransmitted messages in [28]. Both geometric and exponential distributions have light tails, whereas many statistics in network traffic have heavy-tails in the presence of LRD. Although the Hurst parameter is an indicator about the autocorrelation of delay, an intrinsically exponential type marginal delay distribution and a Markovian recovery mechanism show that LRD is not expected for Bimodal Multicast. This substantiates the idea that short-range dependence is intrinsically due to the epidemic mechanism of the protocol.

We have also demonstrated through simulations that while Bimodal Multicast generates desirable network traffic, SRM traffic shows LRD at time scales over 1 s to 1 min. Smaller time scales might represent the effect of SRM's control actions at the granularity of time-to-live, and request and repair timers, spanning time scales smaller than 1 s. In contrast, the larger scales show the overall effect on the network due to congestion caused by the overhead of the protocol and they span time scales from 1 s to a minute or so. This range of time scales is generally sufficient for traffic modeling purposes as shown in [30] due to the finite buffer sizes of real systems. On the other hand, LRD induced by the application layer, which is excluded in this study, is relevant at time scales of minutes, and hours. The timer mechanism of SRM depends on the estimations of delay over the network. The adjustment of request and repair timers accordingly can have an effect that propagates self-similarity to scales from seconds to minutes although these timers are in much smaller scales. The longer the delay estimate, the larger the timers are set. As a result, in the presence of heavy tailed delays in the Internet, SRM's behavior is expected to degrade. In the unicast case, TCP's congestion control mechanism is suggested as the cause for the self-similarity induced at time scales of a few milliseconds to tens of seconds and analyzed to an extent in [31]. We believe that a similar analysis is valid also for SRM, which is left as future work.

6. Conclusion

We have studied the traffic characteristics and performance criteria for Bimodal Multicast protocol in tree, transit-stub and connected cluster topologies. Various drop rates, group sizes and data transmission rates have been considered in the transit-stub topology. The Hurst parameter is estimated from the delay sequence as a representative measure of LRD. We established that the protocol generates well-behaved traffic with no signs of LRD in realistic network topologies and parameters. These results reinforce the scalability results reported in [1]. As the group size increases, overhead traffic, throughput, loss ratio and the Hurst parameter, which is around 0.5, all remain stable proving the scalability of Bimodal Multicast. We also conclude that Bimodal Multicast protocol does not intrinsically lead to long-range dependent traffic when the network topology is tree, clusters or transit-stub. On the other hand, SRM protocol shows LRD although at a moderate level in clusters. Since LRD behavior of the traffic leads to adverse implications for network performance, we conclude that Bimodal Multicast is a superior protocol in this sense.

As future work, Bimodal Multicast deserves more research in terms of traffic characteristics for various

parameters such as gossip rate and buffer sizes. We aim to provide a stochastic model involving the parameters and the mechanisms of Bimodal Multicast. Comparative studies with other scalable multicast approaches will help to identify efficient protocol mechanisms. We emphasize multicast communication as a prominent paradigm for future applications. In addition, concrete results to be obtained for multicast communication mechanisms would also provide a basis in examining unicast and wireless data communication mechanisms in terms of their effects on network performance. What happens when Bimodal Multicast mixes with self-similar traffic of WAN/Internet is an open question. We expect that performance results for the multiplexing of short-range and long-range dependent traffic will be valid. In multiservice networks, the control of the hybrid traffic with different QoS requirements will be important. Modeling such a control on the basis of a stochastic framework also remains as future work.

References

[1] Birman, K.P., Hayden, M., Ozkasap, O., Xiao, Z., Budiu, M. and Minsky, Y., "Bimodal Multicast", ACM Transactions on Computer Systems, 17(2): 41-88, 1999.

[2] Leland, W.E., Taqqu, M.S., Willinger, W., Wilson, D.V., On the Self-Similar Nature of Ethernet Traffic (Extended version), IEEE/ACM Transactions on Networking, 2(1): 1-15, 1994.

[3] Willinger, W., Paxon, V., Reidi, R., Taqqu M., Long-Range Dependence and Data Network Traffic. Long-Range Dependence: Theory and Applications, P. Doukhan, G. Oppenheim and M. S. Taqqu, eds., Birkhauser, 2001.

[4] Caglar, M., A Long-Range Dependent Workload Model for Packet Data Traffic. Submitted to Mathematics of Operations Research, 2001.

[5] Norros, I., On the Use of Fractional Brownian Motion in the Theory of Connectionless Networks. IEEE Journal on Selected Areas in Communications, 13: 953-62, 1995.

[6] Park, K., Kim, G.T., Crovella, M.E., On the Effect of Traffic Self-Similarity on Network Performance, Proceedings of the SPIE International Conference on Performance and Control of Network Systems, November, 1997.

[7] Liu, C., Error Recovery in Scalable Reliable Multicast, Ph.D. dissertation, University of Southern California, 1997.

[8] Lucas, M., Efficient Data Distribution in Large-Scale Multicast Networks, Ph.D. dissertation, Department of Computer Science, University of Virginia, 1998.

[9] Feldmann, A., Gilbert, C., Willinger, W., and Kurtz, T.G., "The Changing Nature of Network Traffic: Scaling Phenomena". Computer Communication Review, 28: 5-29, 1998.

[10] Ozkasap, O., Caglar, M., Multicast Network Traffic and Long-Range Dependence. Proceedings, IASTED, International Conference on Advances in Communications, July, 2001.

[11] Demers, A., Greene, D., Hauser, C., Irish, W., Larson, J., Shenker, S., Sturgis, H., Swinehart, D. and Terry, D., 1987, Epidemic Algorithms for Replicated Database Maintenance, Proceedings of the Sixth ACM Symposium on Principles of Distributed Computing, Vancouver, British Columbia, pp. 1-12.

[12] Lidl, K., Osborne, J. and Malcome, J., 1994, Drinking from the Firehose: Multicast USENET News, USENIX Winter 1994, pp. 33-45.

[13] Ladin, R., Lishov, B., Shrira, L. and Ghemawat, S., 1992, Providing Availability using Lazy Replication, ACM Transactions on Computer Systems, 10(4): 360-391.

[14] Bajaj, S., Breslau, L., Estrin, D., et al., Improving Simulation for Network Research, 1999, USC Computer Science Dept. Technical Report 99-702.

[15] Labovitz, C., Malan, G.R. and Jahanian, F., 1997, Internet Routing Instability, Proceedings of SIGCOMM '97, pp. 115-126.

[16] Paxson, V., 1997, End-to-End Internet Packet Dynamics, Proceedings of SIGCOMM '97, pp. 139-154.

[17] Floyd, S., Jacobson, V., Liu, C., McCanne, S. and Zhang, L., A Reliable Multicast Framework for Light-Weight Sessions and Application Level Framing, IEEE/ACM Transactions on Networking, 5(6): 784-803, 1997.

[18] Speakman, T., Farinacci, D., Lin, S. and Tweedly, A., 1998, PGM Reliable Transport Protocol, Internet-Draft.

[19] Paul, S., Sabnani, K., Lin, J. C. and Bhattacharyya, S., 1997, Reliable Multicast Transport Protocol (RMTP), IEEE Journal on Selected Areas in Communications, special issue on Network Support for Multipoint Communication, 15(3),

[20] Ozkasap, O., Scalability, Throughput Stability and Efficient Buffering in Reliable Multicast Protocols, Technical Report, TR2000-1827, Department of Computer Science, Cornell University.

[21] Andren, J.; Hilding, M.; Veitch, D., Understanding End-to-End Internet Traffic Dynamics, GLOBECOM 1998. The Bridge to Global Integration, Volume 2, 1118 -1122, 1998.

[22] Borella, M.S., Brewster G.B., Measurement and Analysis of Long-Range Dependent Behavior of Internet Packet Delay, Proceedings of IEEE INFOCOM 1998, Volume 2, pp. 497-504.

[23] Li, Q., Mills, D.L., On the long-range dependence of packet round-trip delays in Internet, ICC 98. Volume 2, 1185–1191, 1998.

[24] GT-ITM topology generator. http://www.isi.edu/nsnam/ns/ns-topogen.html

[25] Abry, P., Veitch, D., Wavelet analysis of long-range-dependent traffic, IEEE Transactions on Information Theory 44(1): p.2-15, January 1998.

[26] Veitch, D. and Abry, P., "A Wavelet Based Joint Estimator of the Parameters of Long-Range Dependence", IEEE Trans. on Information Theory, 45: 878-897, 1999

[27] Caglar, M., 2000, Simulation of Fractional Brownian Motion with Micropulses, Advances in Performance Analysis, 3: 43-69p.

[28] Caglar, M. and Ozkasap, O., Traffic Properties of Scalable Multicast Communication: Comparison of Bimodal Multicast and SRM, submitted, 2002.

[29] Bailey, N.T.J., The Mathematical Theory of Infectious Diseases and its Applications, Charles Griffin and Company, London, 1975.

[30] Grossglauser, M., Bolot, J-C., On the relevance of Long-Range Dependence in Network Traffic. IEEE/ACM Trans. On Networking, 7: 629-640, 1999.

[31] Sikdar, B., Vastola, K., "On the Contribution of TCP to the Self-Similarity of Network Traffic", Proceedings of the 2001 Tyrrhenian International Workshop on Digital Communications: Evolutionary Trends of the Internet, Taormina, Italy, September 17-20, 2001.