A bandwidth analysis of tree-based reliable multicast protocols

Christian Maihöfer and Kurt Rothermel
University of Stuttgart

ABSTRACT

In this paper, an analytical bandwidth evaluation of generic tree-based reliable multicast protocols is presented. Our analysis is based on a realistic system model, including data packet and control packet loss, asynchronous local clocks and imperfect scope-limited local groups. Two of the four considered protocol classes use aggregated acknowledgments (AAKs). With AAKs they are able to provide reliability even in case of node failures. All tree-based protocols provide good scalability for large receiver groups. Relating to protocols with aggregated acknowledgments, the analysis shows only little additional bandwidth overhead and therefore high throughput rates. Finally, we have analyzed the influence of the branching factor on a protocol's performance. Our results show that the optimal branching factor depends mainly on the probability for receiving messages from other local groups. If local groups are assigned to a separate multicast address, the optimal branching factor is 2. On the other hand, if TTL scoping is used as in RMTP or TMTP and therefore the probability for receiving messages from other local groups is greater than zero, larger local groups provide better performance.

Keywords: branching factor, aggregated acknowledgments, local groups, scope overlapping, TTL scoping

1. INTRODUCTION

A number of reliable multicast transport protocols have been proposed in the literature, which are based on the acknowledgment scheme. Reliability is ensured by replying acknowledgment messages from the receivers to the sender, either to confirm correct data packet delivery or to ask for a retransmission.

Reliable multicast protocols that send acknowledgment messages directly to the sender can overwhelm the sender with too many positive (ACK) or negative acknowledgment (NAK) messages. This problem is the well-known acknowledgment implosion problem, which is a vital challenge for the design of reliable multicast protocols, since it limits the scalability for large receiver groups. Tree-based approaches promise to be scalable even for a large number of receivers, since they arrange receivers into a hierarchy, called ACK tree.[1] Leaf node receivers send their positive or negative acknowledgments to their parent node in the ACK tree, called group leader. Each group leader is responsible for collecting ACKs or NAKs only from their direct child nodes in the hierarchy, which we will call a local group. Since the maximum number of child nodes and therefore the local group size is limited, no node is overwhelmed with messages and scalability for a large receiver group is ensured. The maximum number of child nodes can be determined according to the processing performance of a node, its available network bandwidth, its memory equipment, and its reliability.

In this paper we present a throughput analysis based on bandwidth requirements as well as the overall bandwidth consumption of all group members, which refer to the data transfer costs. One characteristic of multicast transmissions is that the component with the weakest performance may determine the transmission speed. This means, a group member with a low bandwidth connection, low processing power, high packet loss rate or high packet delay may prevent high transmission rates. Therefore, it is very useful to be able to quantify the necessary requirements for a given multicast protocol.

Regarding the branching factor, our results show that the optimal factor mainly depends on the used reliable multicast protocol and the probability for receiving retransmissions destined to other local groups, which we will denote as scope overlapping probability. If the scope overlapping probability is low, a small branching factor results in the highest throughput and lowest bandwidth consumption. On the other hand, if the scope overlapping probability grows, the optimal branching factor increases also.

The remainder of this paper is structured as follows. In Section 2 we discuss the background of our analysis and take a look at related work. In Section 3 we briefly classify the analyzed protocols. Our bandwidth evaluation in

{christian.maihoefer|kurt.rothermel}@informatik.uni-stuttgart.de, University of Stuttgart, Institute of Parallel and Distributed High-Performance Systems (IPVR), Breitwiesenstr. 20-22, D-70565 Stuttgart, Germany

Section 4 starts with a definition of the assumed system model before the various protocol classes are analyzed in detail. To illustrate the results, some numerical evaluations are presented in Section 5 before we conclude with a brief summary.

2. RELATED WORK

Reliable multicast protocols were already analyzed in previous work. The first processing requirements analysis of generic reliable multicast protocols was presented by Pingali et al..[2] They compared the class of sender- and receiver-initiated protocols. Following analytical papers are often based on the model and analytical methods introduced by Pingali et al.[2] Levine et al.[3] have extended the analysis to the class of ring- and tree-based approaches. In Maihöfer et al.[4] protocols with aggregated acknowledgments are considered.

A bandwidth analysis of generic reliable multicast protocols was done by Kasera et al.,[5] Nonnenmacher et al.[6] and Poo et al.[7] Kasera et al.[5] studied and compared local recovery techniques. The system model is based on a special topology structure consisting of a source link from the sender to the backbone, backbone links and finally tail links from the backbone to the receivers. In Nonnenmacher et al.[6] a similar topology structure is used. They studied the performance gain of protocols using parity packets to recover from transmission errors. The protocols use receiver-based loss detection with multicasted NAKs and NAK avoidance. In Poo et al.,[7] non-hierarchical protocols are compared. In contrast to previous work, not only stop-and-wait error recovery is considered in the analysis but also go-back-N and selective-repeat schemes.

Our paper differs from previous work in the following ways. First, we consider the loss of data packets *and* control packets. Second, we assume that local clocks are not synchronized which affects the NAK-avoidance scheme. NAK-avoidance works less efficiently with this more realistic assumption. Third, our analysis considers that local groups may not be confined perfectly, so that local data or control packets may reach nodes in other local groups. Fourth, our work extends previous analysis by two new tree-based protocol classes. They are based on aggregated ACKs to be able to cope with node failures. Finally, in contrast to our previous work we focus in this work on the protocols' optimal branching factor.[8]

3. CLASSIFICATION OF RELIABLE MULTICAST PROTOCOLS

In this section we briefly classify the reliable multicast protocols analyzed in this paper. A more detailed and more general description for some of these classes can be found in the literature.[2,3,6] Tree-based approaches organize the receivers into a tree structure called ACK tree, which is responsible for collecting acknowledgments and sending retransmissions. We assume that the sender is the root of the tree. If a receiver needs a retransmission, the parent node in the ACK tree is informed rather than the sender. The parent nodes are called group leaders for their children, which form a local group. Note that a group leader may also be a child of another local group. A child that is only a receiver rather than a group leader is called leaf node.

The first considered scheme of this class (H1) uses ACKs sent by the receivers to their group leaders to indicate correctly received packets. Each group leader that is not the root node also sends an ACK to its parent group leader until the root node is reached. If a timeout for an ACK occurs at a group leader or the root, a multicast retransmission is invoked. An example of a protocol similar to our definition of (H1) is RMTP.[9] The second scheme (H2) is based on NAKs with NAK suppression. A NAK is delayed a random time and sent to the local group using multicast. If a NAK from another unsuccessful member is received, the own NAK is suppressed. For deciding deterministically when packets can be removed from memory selective ACKs (SAKs) are used. A SAK is sent to the parent node after a certain number of packets are received or after a certain time period has expired, to propagate the state of a receiver to its group leader. TMTP[10] is an example for class (H2).

Before the next scheme will be introduced, it is necessary to understand that (H1) and (H2) can guarantee reliable delivery only if no group member fails in the system. Assume for example that a group leader G_1 fails after it has acknowledged correct reception of a packet to its group leader G_0 which is the root node. If a receiver of G_1's local group needs a retransmission, neither G_1 nor G_0 can resend the data packet since G_1 has failed and G_0 has removed the packet from memory. This problem is solved by aggregated hierarchical ACKs (AAKs) of the third scheme (H3). A group leader sends an AAK to its parent group leader after all children have acknowledged correct reception. After a group leader or the root node has received an AAK, it can remove the corresponding data from memory because all members in this subhierarchy have already received it correctly. Lorax[11] and RMTP-II[12] are examples for AAK protocols. Our definition of (H3)'s generic behavior is as follows:

1. Group leaders send a local ACK after the data packet is received correctly.
2. Leaf node receivers send an AAK after the data packet is received correctly.
3. The root node and group leaders wait a certain time to receive local ACKs from their children. If a timeout occurs, the packet is retransmitted to all children or selective to those whose ACK is missing. Since leaf node receivers send only AAKs rather than local ACKs, a received AAK from a receiver is also allowed to prevent the retransmission.
4. The root node and group leaders wait to receive AAKs from their children. Upon reception of all AAKs, the corresponding packet can be removed from memory and a group leader sends an AAK to its parent group leader. If a timeout occurs while waiting for AAKs, a unicast AAK query is sent to the affected nodes.
5. If a group leader or leaf node receiver receives further retransmissions after an AAK has been sent or the prerequisites for sending an AAK are met, these data packets are acknowledged by AAKs rather than ACKs. The same applies for receiving an AAK query that is replied with an AAK if the prerequisites are met.

In summary, ACKs are used for fast error recovery in case of message loss and AAKs to clear buffer space. Besides the AAK scheme, we consider in our analysis of (H3) a threshold scheme to decide whether a retransmission is performed using unicast or multicast. The sender or group leader compares the number of missing ACKs with a threshold parameter. If the number of missing ACKs is smaller than this threshold, the data packets are retransmitted using unicast. Otherwise, if the number of missing ACKs exceeds the threshold, the overall network and node load is assumed to be lower using multicast retransmission.

Our next protocol will be denoted as (H4) and is a combination of the negative acknowledgment with NAK suppression scheme (H2) and aggregated acknowledgments (H3). Similar to (H2), NAKs are used to start a retransmission. Instead of selective periodical ACKs, aggregated ACKs are used to announce the receivers' state and allow group leaders and the sender to remove data from memory. Like SAKs, we assume that AAKs are sent periodically. We define the generic behavior of (H4) as follows:

1. Upon detection of a missing or corrupted data packet, receivers send a NAK per multicast scheduled at a random time in the future and provided that not already a NAK for this data packet is received before the scheduled time. If no retransmission arrives within a certain time period, the NAK sending scheme is repeated.
2. Group leaders and the sender retransmit a packet per multicast if a NAK has been received.
3. After a certain number of correctly received data packets, leaf node receivers send an AAK to its group leader in the ACK tree. A group leader forwards this AAK to its parent group leader or sender, respectively, as soon as the same data packets are correctly received and the corresponding AAKs from all child nodes are received.
4. The sender and group leaders initiate a timer to wait for all AAKs to be received. If the timer expires, an AAK query is sent to those child nodes whose AAK is missing.
5. If a group leader or leaf node receiver gets an AAK query and the prerequisites for sending an AAK are met, the query is acknowledged with an AAK.

4. BANDWIDTH ANALYSIS

4.1. Model

Our model is similar to the one used by Pingali et al.[2] and Levine et al..[3] A single sender is assumed, multicasting to R identical receivers. In case of tree-based protocols, the sender is the root of the ACK tree. We assume that nodes do not fail and that the network is not partitioned, i.e. retransmissions are finally successful. In contrast to previous work, packet loss can occur on both, data packets *and* control packets. Multicast packet loss probability is given by q and unicast packet loss probability by p for any node.

We assume that losses at different nodes are independent events. In fact, since receivers share parts of the multicast routing tree, this assumption does not hold in real networks. However, if all classes have similar trees no protocol class is privileged relative to another one by this assumption.

4.2. Protocol independent methods

The main issue for our analysis is to obtain the number of necessary transmissions, M, to deliver a data packet correctly to all receivers. Many other quantities, like the number of ACK or NAK packets and the number of timeouts that have to be processed depend on M.

Analogous to M, which is the total number of data packet transmissions for all receivers, M_r denotes the number of necessary data packet transmissions for a single receiver r. M_r depends on the probability $\tilde{p}$ that a retransmission is necessary. $\tilde{p}$ is the failure or retransmission probability for a single receiver and is made up of the data and control packet loss probabilities (see following sections). With $\tilde{p}$, the probability that the number of necessary transmissions M_r for receiver r is smaller or equal to m (m=1, 2, ...) is:

$$P(M_r \leq m) = 1 - \tilde{p}^m. \tag{1}$$

As the packet losses at different receivers are assumed to be independent from each other, the following holds,[2] where B is the branching factor:

$$P(M \leq m) = \prod_{r=1}^{B} P(M_r \leq m) = (1 - \tilde{p}^m)^B = \sum_{i=0}^{B} \binom{B}{i} (-1)^i \tilde{p}^{im} \tag{2}$$

$$P(M = m) = P(M \leq m) - P(M \leq m - 1)$$

$$E(M) = \sum_{m=1}^{\infty} mP(M = m) = \sum_{i=1}^{B} \binom{B}{i} (-1)^{i+1} \frac{1}{1 - \tilde{p}^i}. \tag{3}$$

$E(M)$ is the expected number of necessary transmissions to receive the data packet correctly at all receivers.

The necessary number of transmissions for a single receiver follows from the Bernoulli distribution. This means, M_r counts the number of trials until the first success occurs. The probability for the first success in a Bernoulli experiment at trial k with probability for success $(1 - \bar{p})$ is:

$$P(X = k) = (1 - \bar{p})\bar{p}^{k-1}. \tag{4}$$

The expectation follows to (see[2]):

$$E(M_r) = \frac{1}{1 - \bar{p}} \tag{5}$$

$$E(M_r | M_r > x) = \frac{x + 1 - x\bar{p}}{1 - \bar{p}} \tag{6}$$

$$P(M_r > x)[E(M_r | M_r > x) - x] = E(M_r) - x. \tag{7}$$

Finally, we have to obtain the number of group leaders. The number of nodes R in a complete tree with branching factor B and height h is:

$$R = \sum_{i=0}^{h-1} B^i = \frac{1 - B^h}{1 - B} \Rightarrow h = \log_B \Big(R(B - 1) + 1\Big). \tag{8}$$

G, the number of group leaders follows to:

$$G = \textstyle\sum_{i=0}^{\log_B [R(B-1)+1]-2} B^i. \tag{9}$$

4.3. Tree-based protocol (H1)

Our analysis distinguishes between the three different kinds of nodes in the ACK tree, the sender at the root of the tree, the receivers that form the leaves of the ACK tree and the receivers that are inner nodes. We will call these inner receivers group leaders. Group leaders are sender and receiver as well.

Our analysis of all tree-based protocols is based on the assumption that each local group consists of exactly B members and one group leader. We assume further, that when a group leader has to send a retransmission, the group leader has already received this packet correctly. The following subsections analyze the bandwidth requirements at the sender W_S^{H1}, receivers W_R^{H1} and group leaders W_H^{H1}.

4.3.1. Sender (root node)

$$W_S^{H1} = W_d(1) + \sum_{m=2}^{M^{H1}} W_d(m) + \sum_{i=1}^{\widetilde{L}^{H1}} W_a(i) \tag{10}$$

$$E(W_S^{H1}) = E(M^{H1})E(W_d) + E(\widetilde{L}^{H1})E(W_a) \tag{11}$$

W_d and W_a are the necessary bandwidth requirements for a data packet or ACK packet, respectively. M^{H1} is the number of necessary transmissions until all members of a local group have received a packet correctly. $\widetilde{L}^{H1}$ is the total number of ACKs received for this packet.

$E(M^{H1})$, the expected number of necessary transmissions, is determined by the probability for a retransmission:

$$\tilde{p} = q_D + (1 - q_D)p_A, \tag{12}$$

i. e. either a data packet is lost (q_D) or the data packet is received correctly and the ACK is lost ($(1 - q_D)p_A$). Now, $E(M^{H1})$ can be determined with Eq. 3:

$$E(M^{H1}) = \sum_{i=1}^{B} \binom{B}{i} (-1)^{i+1} \frac{1}{1 - \tilde{p}^i}. \tag{13}$$

Group leaders receive an ACK from each child node for every data transmission provided that the data packet and ACK packet was not lost. The mean number of ACKs $E(\widetilde{L}^{H1})$ is therefore:

$$E(\widetilde{L}^{H1}) = BE(M^{H1})(1 - q_D)(1 - p_A). \tag{14}$$

4.3.2. Receiver (leaf node)

$E(N_{r,t}^{H1})$ is the total number of received transmissions at receiver r and consists mainly of the sent messages from the parent $E(N_r^{H1})$, provided that each local group has its own multicast address. However, if the whole multicast group has only one multicast address, retransmissions may reach members outside of this local group. The probability for receiving a retransmission from another local group is assumed to be p_l for any receiver. Such received transmissions from other local groups increase the load of a node. In our analysis we assume that transmissions from other local groups do not decrease the necessary number of local retransmissions, since in many cases they are received after a local retransmission have already been triggered.

The number of received transmissions $E(N_r^{H1})$ from the parent node at receiver r is:

$$E(N_r^{H1}) = E(M^{H1})(1 - q_D). \tag{15}$$

The total number of received transmissions $E(N_{r,t}^{H1})$ at receiver r is now (for G see Eq. 9):

$$E(N_{r,t}^{H1}) = E(M^{H1})(1 - q_D) + (G - 1)E(M^{H1})(1 - q_D)p_l. \tag{16}$$

Finally, the bandwidth requirement W_R^{H1} for a receiver is:

$$W_R^{H1} = \sum_{i=1}^{N_{r,t}^{H1}} W_d(i) + \sum_{j=1}^{N_r^{H1}} W_a(j) \tag{17}$$

$$E(W_R^{H1}) = E(N_{r,t}^{H1})E(W_d) + E(N_r^{H1})E(W_a). \tag{18}$$

4.3.3. Group leader (inner node)

Since a group leader is a sender and receiver as well, the bandwidth requirement is the sum of the sender and receiver bandwidth requirements. However, $W_d(1)$ is not considered here, since the initial transmission is sent using the multicast routing tree rather than the ACK tree. Furthermore, a group leader may receive additional retransmissions only from $G-2$ group leaders, since its parent group leader and this group leader itself have to be subtracted.

$$W_H^{H1} = \underbrace{\sum_{m=2}^{M^{H1}} W_d(m) + \sum_{k=1}^{\widetilde{L}^{H1}} W_a(k)}_{\text{as sender}} + \underbrace{\sum_{i=1}^{N_g^{H1}} W_d(i) + \sum_{j=1}^{N_r^{H1}} + W_a(j)}_{\text{as receiver}}$$

$$E(N_g^{H1}) = E(M^{H1})(1-q_D) + (G-2)E(M^{H1})(1-q_D)p_l \tag{19}$$

$$E(W_H^{H1}) = \big(E(M^{H1})-1\big)E(W_d) + E(\widetilde{L}^{H1})E(W_a) + E(N_g^{H1})E(W_d) + E(N_r^{H1})E(W_a) \tag{20}$$

$$= E(W_S^{H1}) + E(W_R^{H1}) - E(W_d(1)) - E(M^{H1})(1-q_D)p_l E(W_d). \tag{21}$$

The maximum throughput rates Λ_S^{H1}, Λ_R^{H1}, Λ_H^{H1} for the sender, receiver and group leader are:

$$\Lambda_S^{H1} = \frac{1}{E(W_S^{H1})}, \quad \Lambda_R^{H1} = \frac{1}{E(W_R^{H1})}, \quad \Lambda_H^{H1} = \frac{1}{E(W_H^{H1})}. \tag{22}$$

Overall system throughput Λ^{H1} is given by the minimum of the throughput rates for the sender, receiver and group leader:

$$\Lambda^{H1} = min\{\Lambda_S^{H1}, \Lambda_H^{H1}, \Lambda_R^{H1}\}. \tag{23}$$

The total bandwidth consumption of protocol (H1) is then the sum of the sender's, leaf node receivers' and group leaders' bandwidth consumptions:

$$W^{H1} = W_S^{H1} + (R-G+1)W_R^{H1} + (G-1)W_H^{H1}. \tag{24}$$

4.4. Tree-based protocol (H2)

(H2) uses selective periodical ACKs (SAKs) and NAKs with NAK avoidance. The sender and group leaders collect all NAKs belonging to one round and send a retransmission if the waiting time has expired and at least one NAK has been received. We have to distinguish between the number of rounds and the number of transmissions. The number of rounds is equal or greater than the number of retransmissions, since if a sender or receiver receives no NAK within one round, no retransmission is invoked.

A SAK is sent by the receiver to announce its state, i.e. its received and missed packets, after a sequence of data packets have been received. We assume that a SAK is sent after a certain period of time. Therefore, when analyzing the processing requirements for a *single* packet, only the proportionate requirements for sending and receiving a SAK (W_Φ) is considered. S is assumed to be the number of SAKs received by the sender in the presence of possible SAK loss: $E(S) = (1-p_A)B$.

4.4.1. Sender (root node)

$$W_S^{H2} = \sum_{i=1}^{M^{H2}} W_d(i) + \sum_{j=1}^{\widetilde{L}^{H2}} W_n(j) + SW_\Phi \tag{25}$$

$$E(W_S^{H2}) = E(M^{H2})E(W_d) + E(\widetilde{L}^{H2})E(W_n) + E(S)E(W_\Phi) \tag{26}$$

$E(M^{H2})$ is determined by Eq. 3 with loss probability $\tilde{p} = q_D$. A round starts with the sending of a data packet and ends with the expiration of a timeout at the sender. Usually, there will be one data transmission in each round. However, if the sender receives no NAKs due to NAK losses, no retransmission is made and new NAKs must be sent by the receivers in the next round. O_r^{H2} is the number of rounds for receiver r. The number of rounds is the sum of the number of necessary rounds for sending transmissions M_r^{H2} and the number of empty rounds $O_{e,r}^{H2}$ in which all NAKs are lost and therefore no retransmission is made:

$$O_r^{H2} = M_r^{H2} + O_{e,r}^{H2}. \tag{27}$$

$E(M_r^{H2})$ is given in Eq. 5 with failure probability $\bar{p} = q_D$. The expected number of empty rounds $E(O_{e,r}^{H2})$ is the expected number of empty rounds after the first transmission plus the expected number of empty rounds after the second transmission and so on:

$$E(O_{e,r}^{H2}) = \sum_{k=1}^{E(M_r^{H2})-1} \left(\frac{1}{1-p_k} - 1\right). \tag{28}$$

$(1/1 - p_k)$ is the expectation for the number of empty rounds plus the last successful NAK reception at the sender which is subtracted (see Eq. 5). The number of empty rounds after transmission k is determined by the failure probability p_k, i.e. the probability that all sent NAKs in round k are lost:

$$p_k = q_N^{N_k}. \tag{29}$$

N_k, the number of NAKs sent in round k, is obtained as follows. The first receiver that did not receive the data packet sends a NAK. The probability for packet loss in round k is q_D^k which is equal to $N_{k,1}$, the probability for the first receiver to send a NAK. Then a second receiver sends a NAK provided that it has received no data packet and no NAK packet. Either the first receiver has sent no NAK (with probability $1 - N_{k,1}$) or the NAK was lost or sent simultaneously (with probability $N_{k,1}(q_N + p_s - q_N p_s)$). As we assume a system model in which local clocks are not synchronized, it is possible that NAKs are sent simultaneously. This probability is given by p_s. Now, N_k can be expressed as follows:

$$N_k = \sum_{i=1}^{B} N_{k,i} \tag{30}$$

$$N_{k,1} = {q_D}^k \tag{31}$$

$$\begin{aligned} N_{k,2} &= {q_D}^k(1 - N_{k,1} + N_{k,1}(q_N + p_s - q_N p_s)) \\ &= N_{k,1} - N_{k,1}^2 + N_{k,1}^2(q_N + p_s - q_N p_s) \end{aligned} \tag{32}$$

$$N_{k,n} = N_{k,n-1} - N_{k,n-1}^2 + N_{k,n-1}^2(q_N + p_s - q_N p_s), n > 1. \tag{33}$$

The total number of rounds O^{H2} for all receivers can be defined analogous to O_r^{H2}:

$$O^{H2} = M^{H2} + O_e^{H2} \tag{34}$$

$$E(O_e^{H2}) = \sum_{k=1}^{E(M^{H2})-1} \left(\frac{1}{1-p_k} - 1\right). \tag{35}$$

To determine $E(\widetilde{L}^{H2})$ we must take into account that NAKs are not only received from members of this local group but may also be received from other local groups with scope overlapping probability p_l (see Eq. 16):

$$E(\widetilde{L}^{H2}) = \vartheta_1(1 - q_N) + (G - 1)\vartheta_1 p_l \tag{36}$$

$$\vartheta_1 = \sum_{k=1}^{E(M^{H2})} N_k \frac{1}{1-p_k}. \tag{37}$$

ϑ_1 is the total number of NAKs sent within a local group. The number of group leaders (G), is obtained with Eq. 9.

4.4.2. Receiver (leaf node)

Retransmissions are received mainly from the parent node, but may also be received from other group leaders. Analogous, NAKs are mainly received from other receivers in the same local group but may also be received from receivers in other local groups. The bandwidth requirement for a receiver is:

$$\begin{aligned} E(W_R^{H2}) = {} & E(M^{H2})(1-q_D)E(W_d) + E(W_\Phi) + P(O_r^{H2} > 1)[E(O_r^{H2}|O_r^{H2} > 1) - 1]\frac{\vartheta_2}{\vartheta_3}E(W_n) \\ & + \underbrace{\Big[P(O^{H2} > 1)[E(O^{H2}|O^{H2} > 1) - 1]\vartheta_2 - P(O_r^{H2} > 1)[E(O_r^{H2}|O_r^{H2} > 1) - 1]\frac{\vartheta_2}{\vartheta_3}\Big](1-q_N)E(W_n)}_{\text{from this local group}} \\ & \underbrace{+ (G-1)p_l E(M^{H2})(1-q_D)E(W_d) + (G-1)p_l P(O^{H2} > 1)[E(O^{H2}|O^{H2} > 1) - 1]\vartheta_2 E(W_n)}_{\text{from other local groups}}. \end{aligned} \tag{38}$$

ϑ_2 is the average number of NAKs sent in each round and ϑ_3 is the mean number of receivers that did not receive a data packet and therefore are supposed to send a NAK:

$$\vartheta_2 = \frac{1}{E(O^{H2})}\sum_{k=1}^{E(M^{H2})} N_k \frac{1}{1-p_k} \tag{39}$$

$$\vartheta_3 = \frac{1}{E(O^{H2})}\sum_{k=1}^{E(M^{H2})} {q_D}^k B \frac{1}{1-p_k}, \tag{40}$$

where $(1/1-p_k)$ is the number of empty rounds plus the last successful NAK sent (see Eq. 5 and 37).

4.4.3. Group leader (inner node)

As the group leader role contains the sender role and the receiver role as well, the processing requirements are:

$$E(W_H^{H2}) = E(W_S^{H2}) + E(W_R^{H2}) - E(W_d(1)) - p_l\Big(E(M^{H2})(1-q_D)E(W_d) + P(O^{H2} > 1)[E(O^{H2}|O^{H2} > 1) - 1]\vartheta_2 E(W_n)\Big). \tag{41}$$

The processing requirements for one other local group have to be subtracted because in contrast to the sender or receivers, a group leader has a local parent group *and* local child group, which are already considered for the normal operations.

Finally, the maximum throughput rates are:

$$\Lambda_S^{H2} = \frac{1}{E(W_S^{H2})}, \quad \Lambda_R^{H2} = \frac{1}{E(W_R^{H2})}, \quad \Lambda_H^{H2} = \frac{1}{E(W_H^{H2})} \tag{42}$$

$$\Lambda^{H2} = min\{\Lambda_S^{H2}, \Lambda_H^{H2}, \Lambda_R^{H2}\}. \tag{43}$$

The total bandwidth consumption of protocol (H2) is:

$$W^{H2} = W_S^{H2} + (R - G + 1)W_R^{H2} + (G-1)W_H^{H2}. \tag{44}$$

4.5. Tree-based protocol (H3)

We assume that the correct transmission of a data packet consists of two phases. In the first phase, the data is transmitted and ACKs are collected until all ACKs are received, i.e. until all nodes have received the data packet. Then the second phase starts, in which the missing AAKs are collected. Note that most AAKs are already received in phase one, since AAKs are sent from group leaders as soon as all children have sent their AAKs. In this case, a retransmission is acknowledged with an AAK rather than an ACK. So, only nodes whose AAK is missing must be queried in phase two.

4.5.1. Sender (root node)

The bandwidth requirement of a sender is:

$$W_S^{H3} = \sum_{j=1}^{M_m^{H3}} W_{d,m}(j) + \sum_{k=1}^{M_u^{H3}} N_u W_{d,u}(k) + \sum_{i=1}^{\widetilde{L}_a^{H3}} W_a(i) + \sum_{w=1}^{L_{aaq}^{H3}} W_{aaq}(w) + \sum_{z=1}^{\widetilde{L}_{aa}^{H3}} W_{aa}(z). \tag{45}$$

M_m^{H3} and M_u^{H3} are the number of necessary multicast or unicast transmissions, respectively. $W_{d,m}$ and $W_{d,u}$ determine the bandwidth requirements for a multicast or unicast packet transmission. W_{aa} is the necessary bandwidth for an AAK and $\widetilde{L}_{aa}^{H3}$ is the number of received AAKs. The processing of AAKs is similar to the processing of data packets and ACKs. If AAKs are missing after a timeout has occurred, the sender or group leader sends unicast AAK query messages (W_{aaq}) to the corresponding child nodes. Note that this processing is started after all ACKs have been received and no further retransmissions due to lost data packets are necessary. L_{aaq}^{H3} is the number of necessary unicast AAK queries in the presence of message loss.

With probability p_t that unicast is used for retransmissions, the number of unicast and multicast transmissions are:

$$M_u^{H3} = p_t(M^{H3} - 1) \tag{46}$$

$$M_m^{H3} = (1 - p_t)(M^{H3} - 1) + 1. \tag{47}$$

Please note that the first transmission is always sent with multicast. The probability for a retransmission due to data or ACK loss is given by:

$$\tau = \underbrace{\overbrace{p_t p_D}^{\text{unicast}} + \overbrace{(1-p_t)q_D}^{\text{multicast}}}_{\text{data loss}} + \underbrace{\Big[1 - \Big(\overbrace{p_t p_D}^{\text{unicast}} + \overbrace{(1-p_t)q_D}^{\text{multicast}}\Big)\Big] p_A}_{\text{no data loss but ACK loss}}. \tag{48}$$

$E(M^{H3})$ is determined by τ instead of $\tilde{p}$ and B instead of R analogous to Eq. 3. ϕ is the threshold for unicast or multicast retransmissions. If the current number of nodes n_k, which need a retransmission is smaller than the threshold ϕ, then unicast is used for the retransmission. p_t is the probability that the current number of nodes n_k is smaller than the threshold ϕ:

$$p_t = \frac{1}{M^{H3}} \sum_{k=1}^{M^{H3}} \begin{cases} 1, & n_k < \phi \\ 0, & n_k \geq \phi \end{cases} \tag{49}$$

Since p_t is used to obtain M^{H3}, p_t can only be determined if $q_D = p_D$. In this case, parameter p_t is unnecessary to determine M^{H3}. N_u is the mean number of receivers per round for which a unicast retransmission is invoked:

$$N_u = \frac{1}{M_u^{H3}} \sum_{k=1}^{M^{H3}} \begin{cases} n_k, & n_k < \phi \\ 0, & n_k \geq \phi \end{cases} \tag{50}$$

$E(N_r^{H3})$ is the total number of transmissions that reach receiver r with unicast and multicast from its parent node in the ACK tree:

$$E(N_r^{H3}) = \frac{N_u}{B} E(M_u^{H3})(1 - p_D) + E(M_m^{H3})(1 - q_D). \tag{51}$$

The number of ACKs that reach the sender or group leader in the presence of ACK loss is given by:

$$E(\widetilde{L}_a^{H3}) = B E(N_r^{H3})(1 - p_A) p_c. \tag{52}$$

p_c is the probability that no AAK can be sent due to missing AAKs of child nodes. The number of AAK query rounds L_1, is determined by the probability $\hat{p}$ that a query fails:

$$\hat{p} = p_q + (1 - p_q)p_{AA}. \tag{53}$$

$E(L_1)$ can be determined analogous to Eq. 3 with B_{aa} instead of B and $\hat{p}$ instead of $\tilde{p}$. B_{aa} is the number of receivers, the sender has to query when the first AAK timeout occurs, which is equal to the number of receivers that have not already successfully sent an AAK in the first phase:

$$E(L_1) = \sum_{i=1}^{B_{aa}} \binom{B_{aa}}{i} (-1)^{i+1} \frac{1}{1 - \hat{p}^i} \tag{54}$$

$$B_{aa} = B\Big(p_c + (1 - p_c)p_{AA}\Big)^{E(N_r^{H3})}. \tag{55}$$

$p_c + (1 - p_c)p_{AA}$ is the probability that no AAK can be sent in a round or that the AAK is lost. Queries are sent with unicast to the nodes whose AAK is missing. The total number of queries in all rounds are:

$$E(L_{aaq}^{H3}) = \textstyle\sum_{k=1}^{E(L_1)} B_{aa}\hat{p}^{(k-1)}. \tag{56}$$

The number of AAKs received at the sender is the number of AAKs in the retransmission phase plus the number of AAKs in the AAK query phase, which is exactly one AAK from every receiver in B_{aa} (see Eq. 52).

$$E(\tilde{L}_{aa}^{H3}) = BE(N_r^{H3})(1 - p_{AA})(1 - p_c) + B_{aa}. \tag{57}$$

Now, $E(W_S^{H3})$ is entirely determined by:

$$E(W_S^{H3}) = E(M_u^{H3})N_u E(W_{d,u}) + E(M_m^{H3})E(W_{d,m}) + E(\tilde{L}_a^{H3})E(W_a) + E(L_{aaq}^{H3})E(W_{aaq}) + E(\tilde{L}_{aa}^{H3})E(W_{aa}). \tag{58}$$

4.5.2. Receiver (leaf node)

The bandwidth requirement at the receiver is given by:

$$W_R^{H3} = \sum_{i=1}^{N_{r,t}^{H3}} W_d(i) + \sum_{j=1}^{L_a^{H3}} W_a(j) + \sum_{k=1}^{L_{aa}^{H3}} W_{aa}(k) + \sum_{l=1}^{\tilde{L}_{aaq}^{H3}} \Big(W_{aa}(l) + W_{aaq}(l)\Big). \tag{59}$$

$N_{r,t}^{H3}$ is the total number of transmission that reach receiver r. In contrast to the already obtained N_r^{H3}, additional data retransmissions are considered from other local groups that may be received with probability p_l:

$$E(N_{r,t}^{H3}) = E(N_r^{H3}) + (G - 1)E(N_r^{H3})p_l. \tag{60}$$

The number of transmissions that are acknowledged with an ACK, L_a^{H3}, or with an AAK, L_{aa}^{H3}, are:

$$L_a^{H3} = p_c E(N_r^{H3}) \tag{61}$$

$$L_{aa}^{H3} = (1 - p_c)E(N_r^{H3}). \tag{62}$$

Here we assume that only transmissions from this local group are acknowledged. $\tilde{L}_{aaq}^{H3}$, the number of AAK queries received by an receiver are:

$$\tilde{L}_{aaq}^{H3} = \frac{1}{B_{aa}} E(L_{aaq}^{H3})(1 - p_q), \tag{63}$$

where $1/B_{aa}$ is the probability to be a receiver that gets an AAK query. Finally, the expectation for a receiver's bandwidth requirements is:

$$E(W_R^{H3}) = E(N_{r,t}^{H3})E(W_d) + E(L_a^{H3})E(W_a) + E(L_{aa}^{H3})E(W_{aa}) + E(\tilde{L}_{aaq}^{H3})\Big(E(W_{aa}) + E(W_{aaq})\Big). \tag{64}$$

4.5.3. Group leader (inner node)

The bandwidth requirement at a group leader consists of the sender and receiver bandwidth requirements (see Eq. 21):

$$E(W_H^{H3}) = E(W_S^{H3}) + E(W_R^{H3}) - E(W_{d,m}(1)) - E(N_r^{H3})p_l E(W_d). \tag{65}$$

Finally, the maximum throughput rates are:

$$\Lambda_S^{H3} = \frac{1}{E(W_S^{H3})}\ ,\quad \Lambda_R^{H3} = \frac{1}{E(W_R^{H3})}\ ,\quad \Lambda_H^{H3} = \frac{1}{E(W_H^{H3})} \tag{66}$$

$$\Lambda^{H3} = min\{\Lambda_S^{H3}, \Lambda_H^{H3}, \Lambda_R^{H3}\}. \tag{67}$$

The total bandwidth consumption of protocol (H3) is:

$$W^{H3} = W_S^{H3} + (R - G + 1)W_R^{H3} + (G - 1)W_H^{H3}. \tag{68}$$

4.6. Tree-based protocol (H4)

The generic definition of protocol class (H4) is given in Section 3. As in (H3), the correct transmission of a data packet consists of two phases. In the first phase, the data is transmitted. If NAKs are received by the sender or group leaders, retransmissions are invoked. We assume that the retransmission phase is finished before the second phase starts. In this phase AAKs are sent from receivers to their parent in the ACK tree. Missing AAKs are queried per unicast messages by the sender and group leaders. In a NAK-based protocol this is only reasonable if it is done after a certain number of correct data packet transmissions rather than after every transmission. Therefore, the costs for sending and receiving AAKs ($W_{aa,\phi}$) as well as the costs for querying AAKs ($W_{aaq,\phi}$) can be set to a proportionate cost of the other costs.

4.6.1. Sender (root node)

$$W_S^{H4} = \sum_{i=1}^{M^{H4}} W_d(i) + \sum_{j=1}^{\widetilde{L}^{H4}} W_n(j) + \sum_{w=1}^{L_{aaq}^{H4}} W_{aaq,\phi}(w) + \sum_{z=1}^{\widetilde{L}_{aa}^{H4}} W_{aa,\phi}(z) \tag{69}$$

$$E(W_S^{H4}) = E(M^{H4})E(W_d) + E(\widetilde{L}^{H4})E(W_n) + E(L_{aaq}^{H4})E(W_{aaq,\phi}) + E(\widetilde{L}_{aa}^{H4})E(W_{aa,\phi}) \tag{70}$$

$E(M^{H4})$ and $E(\widetilde{L}^{H4})$ are determined analogous to protocol (H2). The number of AAK queries is determined by the probability $\hat{p}$ that a query fails:

$$\hat{p} = p_q + (1 - p_q)p_{AA}. \tag{71}$$

The number of query rounds $E(L_1)$ can be determined analogous to Eq. 3 with B_{aa} instead of B and $\hat{p}$ instead of $\tilde{p}$. B_{aa} is the number of receivers, the sender has to query when the first AAK timeout at the sender occurs. Since receivers send one AAK autonomously after a certain number of successfully receptions, the number of nodes to query in phase 2 is the number of lost AAKs.

$$E(L_1) = \sum_{i=1}^{B_{aa}} \binom{B_{aa}}{i}(-1)^{i+1}\frac{1}{1-\hat{p}^i} \tag{72}$$

$$B_{aa} = Bp_{AA}. \tag{73}$$

The total number of unicast query messages in all rounds are:

$$E(L_{aaq}^{H4}) = \sum_{k=1}^{E(L_1)} B_{aa}\hat{p}^{(k-1)}. \tag{74}$$

Using unicast, only those nodes are queried whose AAK is missing. So finally, the number of received AAKs at the sender is equal to the number of child nodes in the ACK tree:

$$E(\widetilde{L}_{aa}^{H4}) = B. \tag{75}$$

4.6.2. Receiver (leaf node)

$$E(W_R^{H4}) = E(M^{H4})(1-q_D)E(W_d) + P(O_r^{H4}>1)[E(O_r^{H4}|O_r^{H4}>1)-1]\frac{\vartheta_2}{\vartheta_3}E(W_n) + E(W_{aa,\phi}) + E(\widetilde{L}_{aaq}^{H4})\Big(E(W_{aaq,\phi})+E(W_{aa,\phi})\Big)$$
$$+ \underbrace{\Big[P(O^{H4}>1)[E(O^{H4}|O^{H4}>1)-1]\vartheta_2 - P(O_r^{H4}>1)[E(O_r^{H4}|O_r^{H4}>1)-1]\frac{\vartheta_2}{\vartheta_3}\Big](1-q_N)E(W_n)}_{\text{from this local group}}$$
$$+ \underbrace{(G-1)p_l E(M^{H4})(1-q_D)E(W_d) + (G-1)p_l P(O^{H4}>1)[E(O^{H4}|O^{H4}>1)-1]\vartheta_2 E(W_n)}_{\text{from other local groups}} \quad (76)$$

ϑ_2 and ϑ_3 can be obtained analogous to (H2). $E(\widetilde{L}_{aaq}^{H4})$, the number of received AAK queries and replied AAKs is (see Eq. 63):

$$\widetilde{L}_{aaq}^{H4} = \frac{1}{B_{aa}}E(L_{aaq}^{H4})(1-p_q), \quad (77)$$

and the number of rounds O^{H4} is determined analogous to (H2).

4.6.3. Group leader (inner node)

As the group leader role contains the sender role and the receiver role as well, the processing requirements are:

$$E(W_H^{H4}) = E(W_S^{H4}) + E(W_R^{H4}) - E(W_d(1)) - p_l\Big(E(M^{H4})(1-q_D)E(W_d) + P(O^{H4}>1)[E(O^{H4}|O^{H4}>1)-1]\vartheta_2 E(W_n)\Big). \quad (78)$$

Finally, the maximum throughput rates at the sender, receiver, group leader and overall throughput are:

$$\Lambda_S^{H4} = \frac{1}{E(W_S^{H4})}, \quad \Lambda_R^{H4} = \frac{1}{E(W_R^{H4})}, \quad \Lambda_H^{H4} = \frac{1}{E(W_H^{H4})} \quad (79)$$
$$\Lambda^{H4} = min\{\Lambda_S^{H4}, \Lambda_H^{H4}, \Lambda_R^{H4}\}. \quad (80)$$

The total bandwidth consumption of protocol (H4) is:

$$W^{H4} = W_S^{H4} + (R-G+1)W_R^{H4} + (G-1)W_H^{H4}. \quad (81)$$

5. NUMERICAL RESULTS

We examine the relative performance and bandwidth consumption of the analyzed protocols by means of some numerical examples. The mean bandwidth costs are set equal to 1 for data packets (W_d, $W_{d,u}$, $W_{d,m}$), 0.1 for control packets (W_a, W_n, W_{aa}, W_{aaq}) and 0.01 for periodical control packets (W_ϕ, $W_{aa,\phi}$ and $W_{aaq,\phi}$). The following graphs show the throughput of the various protocol classes relative to the normalized maximum throughput of 1.

Figure 1 shows the numerical results with a varying number of receivers. The number of child nodes is set equal to 10 for all classes and the probability for receiving packets from other local groups, p_l, is set equal to 0.001. (H3) is shown with $\phi = 0$ which corresponds with protocol (H1) except for the additional aggregated ACKs of (H3). $\phi = 0$ means that all retransmission are sent with multicast.

All protocol classes experience a throughput degradation with increasing group sizes although the local group size remains constant. This results from our assumption that a packet is received with probability $p_l = 0.001$ outside the scope of a local group. With increasing number of receivers, the number of groups increase also and therefore more packets from other local groups are received. Note that if each local group is assigned a separate multicast address for retransmissions, no packets from other local groups are received and therefore p_l has to be set equal to 0. In this case, the throughput of all hierarchical approaches remains constant.

The protocols with negative acknowledgments and NAK avoidance provide the best performance. As it can be further seen in the figure, the additional overhead for periodical aggregated acknowledgments is very low, therefore

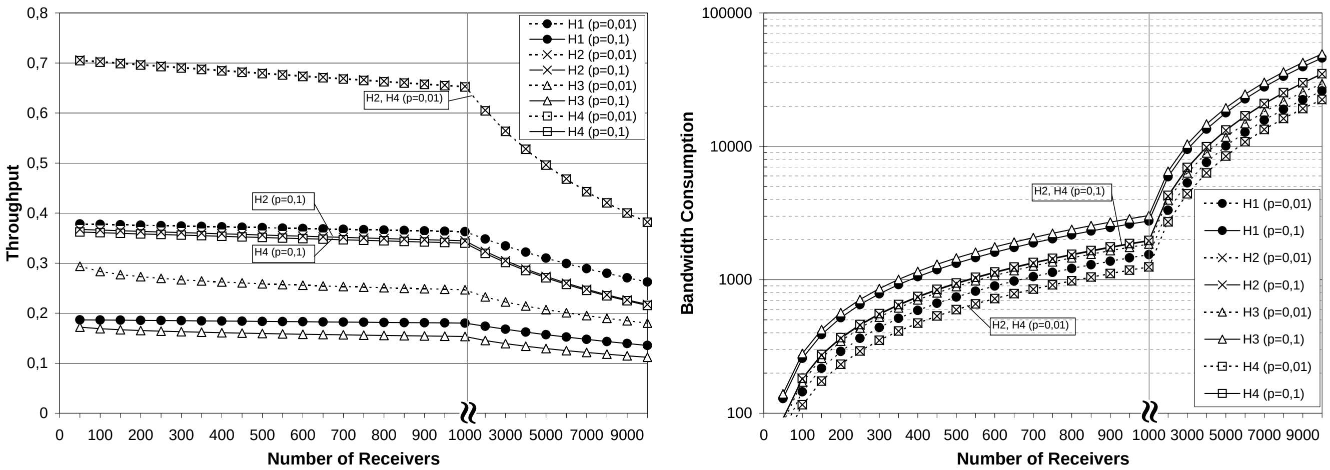

Figure 1. Bandwidth results with respect to the number of receivers for a) throughput and b) bandwidth consumption

(H4) provides almost the same performance as (H2). In case of (H3), the aggregated acknowledgments are sent after every correct message transmission. Therefore, the performance reduction compared to (H1) is more significant than between (H4) and (H2). If (H3)'s aggregated ACKs are also sent periodically as in (H4), the performance would be almost the same as (H1)'s performance. This means that the additional costs for providing reliability even in the presence of node failures are small and therefore acceptable for protocol implementations.

Due to readability, the result for (H3) with $\phi = 2$ is not shown in the figure. With $\phi = 2$, only retransmissions for equal or more than 2 nodes are made using multicast and with unicast otherwise. In this case, (H3) provides better performance especially for a large number of receivers. For a packet loss probability of 0.1 the performance is equal to (H1).

Figure 1.b shows the total bandwidth consumption. The results for (H3) with $\phi = 2$ are not shown in the figure, which are in any case lower compared to (H3) with $\phi = 0$. In case of a packet loss probability of 0.01, (H3) with $\phi = 2$ requires only the bandwidth of (H1). If the packet loss probability is higher than 0.01, the bandwidth consumption is even smaller than that of (H1). For example, with packet loss probability 0.1, (H3)'s bandwidth consumption with $\phi = 2$ is more than 30% below (H1)'s requirements. NAK with NAK avoidance protocols require low bandwidth costs. Therefore, protocols (H2) and (H4) provide the lowest bandwidth consumption.

In the following we will show the impact of the branching factor on the protocols' performance. All displayed results assume a group size of 10000 receivers. For protocol (H3) we have assumed $\phi = B$, i.e. all retransmissions are sent with unicast. Figure 2 shows the throughput with respect to bandwidth requirements. In Figure 2.a it is assumed that local groups are perfectly confined, i.e. messages sent by a group leader are only received by the leader's local group. This can be achieved by assigning a multicast address for each local group. As shown in this figure, small local groups reach the highest throughput. The reason for this result is that less packets must be sent or received at a single inner node if the local group size is small.

In Figure 2.b it is assumed that local groups are not perfectly confined with a scope overlapping probability of $p_l = 0.1$. As the results show, this assumption leads to larger optimal group sizes for most protocols. However, (H1)'s optimal branching factor with loss probability 0.01 is still small. As protocols (H2) and (H4) send not only retransmissions by means of multicast but also NAKs, more messages are received outside the scope of a local group. So, they react more sensitive to not perfectly confined local groups than (H1) and therefore, a larger branching factor and a smaller number of local groups provide better performance.

If the scope overlapping probability p_l is increased, the optimal branching factor increases also for all protocol classes. For example, with $p_l = 0.2$, the optimal branching factor for (H1) with loss probability 0.01 is then 50 and for (H2) over 100 child nodes per group leader. The more local groups exist, the more independent message retransmissions are triggered. If local groups are not perfectly confined in scope, the number of local groups determine the number of received messages from other local groups. Because if more local groups exist, more message retransmissions are triggered and more messages are received outside the scope of the local group. This results in less local groups for

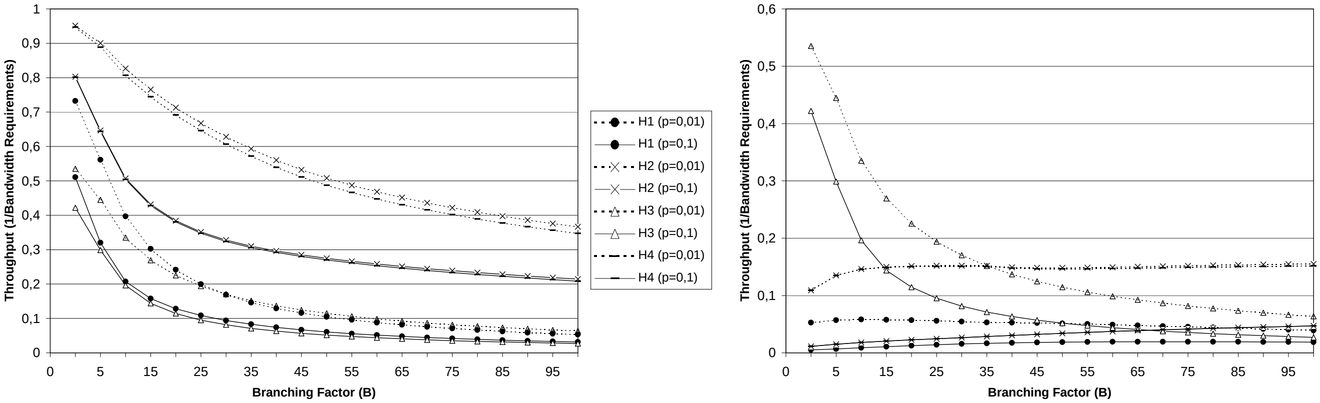

Figure 2. Throughput limited by bandwidth requirements with scope overlapping (a) $p_l = 0$ (left side) and (b) $p_l = 0.1$ (right side)

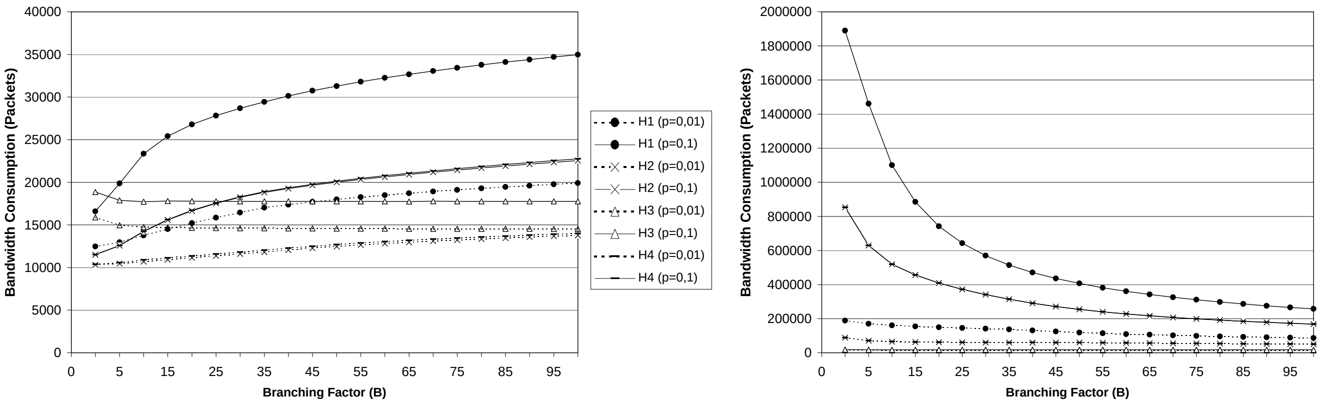

Figure 3. Bandwidth consumption with scope overlapping (a) $p_l = 0$ (left side) and (b) $p_l = 0.1$ (right side)

maximum throughput and therefore in a larger optimal branching factor. If the scope overlapping probability p_l is decreased, the optimal branching factor decreases also. In the extreme case of $p_l = 0$, the optimal branching factor is 2 for all protocols as Figure 2.a shows.

The performance of protocol (H3) is independent of the scope overlapping probability always constant, since retransmissions are always sent with unicast. If the scope overlapping probability p_l exceeds 0.02, (H3) outperforms all other protocol classes.

Finally, Figure 3 shows the total bandwidth consumption of all analyzed protocols in terms of weighted sent and received messages. The results for total bandwidth consumption are similar to the throughput results. With perfectly confined local groups, small local groups result in the lowest bandwidth consumption. In case of imperfect confined local groups, larger local group sizes are preferable. In contrast to the throughput results, we cannot identify in Figure 3.b an optimal value within the displayed range of up to 100 child nodes per group leader. In fact, total bandwidth consumption reacts very sensitive to imperfect confined local groups, so that the optimal group size is larger than 100 nodes. However, we can see for loss probability 0.1 that after an initial decrease, the bandwidth consumption does not decrease significantly as the branching factor is increased. So, a branching factor of 30 or more child nodes would be a reasonable value in this scenario.

6. SUMMARY

We have analyzed the throughput in terms of bandwidth requirements and the overall bandwidth consumption of tree-based multicast protocols assuming a realistic system model with data packet loss, control packet loss and asynchronous clocks. Of particular importance are the analyzed protocol classes with aggregated acknowledgments. In contrast to other hierarchical approaches, these classes provide reliability even in the presence of node failures.

The protocol classes with aggregated acknowledgments lead to only a small throughput decrease and slightly increased overall bandwidth consumption compared to the same classes without aggregated acknowledgments. This means, that the additional costs for providing a reliable multicast service even in the presence of node failures are small and therefore acceptable for reliable multicast protocol implementations.

The impact of the branching factor on the protocols' throughput and bandwidth consumption depends on the probability for receiving messages from other local groups. If local groups are assigned to a separate multicast address and therefore messages are strictly confined to a local group, the optimal branching factor is 2. On the other hand, if TTL scoping is used it can be assumed that messages are not strictly confined to the local group's scope. In this case, larger local groups provide better performance and less bandwidth consumption for most protocols.

REFERENCES

1. K. Rothermel and C. Maihöfer, "A robust and efficient mechanism for constructing multicast acknowledgment trees," in *Proceedings of the Eight International Conference on Computer Communications and Networks*, pp. 139–145, IEEE, (Boston), Oct. 1999.
2. S. Pingali, D. Towsley, and J. F. Kurose, "A comparison of sender-initiated and receiver-initiated reliable multicast protocols," in *Proceedings of the Sigmetrics Conference on Measurement and Modeling of Computer Systems*, pp. 221–230, ACM Press, (New York), May 1994.
3. B. Levine and J. Garcia-Luna-Aceves, "A comparison of reliable multicast protocols," *Multimedia Systems* **6**, pp. 334–348, Sept. 1998.
4. C. Maihöfer, K. Rothermel, and N. Mantei, "A throughput analysis of reliable multicast transport protocols," in *Proceedings of the Ninth International Conference on Computer Communications and Networks*, pp. 250–257, IEEE, (Las Vegas), Oct. 2000.
5. S. Kasera, J. Kurose, and D. Towsley, "A comparison of server-based and receiver-based local recovery approaches for scalable reliable multicast," in *Proceedings of IEEE INFOCOM'98*, pp. 988–995, IEEE, (New York), Apr. 1998.
6. J. Nonnenmacher, M. Lacher, M. Jung, G. Carl, and E. Biersack, "How bad is reliable multicast without local recovery," in *Proceedings of IEEE INFOCOM'98*, pp. 972–979, IEEE, (New York), Apr. 1998.
7. G. Poo and A. Goscinski, "Performance comparison of sender-based and receiver-based reliable multicast protocols," *Computer Communications* **21**, pp. 597–605, June 1998.
8. C. Maihöfer, "A bandwidth analysis of reliable multicast transport protocols," in *Proceedings of the Second International Workshop on Networked Group Communication (NGC 2000)*, pp. 15–26, ACM, (Palo Alto), Nov. 2000.
9. S. Paul, K. Sabnani, J. Lin, and S. Bhattacharyya, "Reliable multicast transport protocol (RMTP)," *IEEE Journal on Selected Areas in Communications, special issue on Network Support for Multipoint Communication* **15**, pp. 407–421, Apr. 1997.
10. R. Yavatkar, J. Griffioen, and M. Sudan, "A reliable dissemination protocol for interactive collaborative applications," in *The Third ACM International Multimedia Conference and Exhibition (MULTIMEDIA '95)*, pp. 333–344, ACM Press, (New York), Nov. 1996.
11. B. Levine, D. Lavo, and J. Garcia-Luna-Aceves, "The case for reliable concurrent multicasting using shared ack trees," in *Proceedings of the Fourth ACM Multimedia Conference (MULTIMEDIA'96)*, pp. 365–376, ACM Press, (New York), Nov. 1996.
12. B. Whetten and G. Taskale, "An overview of the reliable multicast transport protocol II," *IEEE Network* **14**, pp. 37–47, Feb. 2000.