Impact of Virtual Group Structure on Multicast Performance

Markus Hofmann, Manfred Rohrmüller

Institute of Telematics, University of Karlsruhe, 76128 Karlsruhe, Germany
Phone: +49 721 6086413, Fax: +49 721 388097
{hofmann, rohrmuel}@telematik.informatik.uni-karlsruhe.de

Abstract. Scalability will be a key issue in the design and the development of reliable multicast protocols for the Internet. As the geographic span and the size of communication groups increase, efficient connection management schemes including scalable error and congestion control become more and more important. Besides other approaches, several schemes based on subgrouping have been proposed to overcome the well-known implosion problem and to optimize network utilization. However, the performance of these approaches strongly depends on the virtual group structure used for local error recovery and congestion control. While a certain structure may reduce average transfer delay, another one may be suitable to decrease overall network load. This paper discusses the suitability of several metrics for subgrouping of global multicast groups and investigates the impact of virtual group structures on the overall performance of multicast communication.[1]

1 Introduction

Groups present an ubiquitous form of relationship and interaction in human society. People get together in groups to share common interests or to work on collaborative projects. Distributed computer systems are arranged into cooperative groups to master complex problems. Emerging applications, such as collaborative distributed work or information dissemination, rely on group interaction and are expected to require information exchange between a large number of geographically dispersed components. The recent success of applications deployed over the MBone [6] illustrates the enormous potential of group communication and demonstrates the instant need for economic multicast services in the Internet. Recently, multicast data transmission based on Deering's IP multicast extensions [1] has been widely available in the Internet. However, the bearer service provided by IP does not fit the requirements of each individual application. It offers a best-effort service leaving it up to the application to provide the required quality of service. Several error correction schemes have been proposed to improve reliability of multicast communication in the Internet [2]. All of them have to deal with the well-known implosion problem due to feedback messages generated by the receivers.

[1] In Proceedings of 4^{th} COST237 Workshop, December 15-19, Lisboa, Portugal, 1997

The *Scalable Reliable Multicast (SRM)* [3], for example, uses damping and slotting mechanisms to reduce state management overhead. Receivers solely take the responsibility for error correction, which is why SRM achieves a high degree of fault tolerance. SRM is an example of the receiver-based approach for error control. A receiver missing a certain data unit multicasts a repair request to the whole group. Group members that have successfully received the requested packet will multicast it to the entire group. To avoid a flood of repair requests and of retransmission, SRM suppress redundant requests by using timers carefully set and adjusted to the current network load.

Other approaches arrange receivers in a virtual tree hierarchy with the sender at the root. The *Reliable Multicast Transport Protocol (RMTP)* [7] or the *Tree-based Multicast Transport Protocol (TMTP)* [9], for example, represent a multi-level hierarchical approach in which leaf receivers periodically send status messages to the controller of their subgroup. The controllers, in turn, send their status periodically to the higher layer controllers. This scheme continues until the controllers at the highest level send their status directly to the sender. Lost data packets are always requested from a higher level controller, not making use of local retransmissions between neighboring receivers.

The *Local Group based Multicast Protocol (LGMP)*[2] [4] defines a hybrid approach. It supports reliable and semi-reliable transfer of both continuous media and data files. LGMP is based on the principle of subgrouping for local error recovery and local acknowledgment processing. Receivers dynamically organize themselves into subgroups, which are called Local Groups. They dynamically select a Group Controller to coordinate local retransmissions and to handle status reports. The selection of appropriate receivers as Group Controllers is based on the current state of the network and of the receivers themselves. However, the selection of Group Controllers is not a task of a data transfer protocol like LGMP. Instead, we have defined and implemented a separate configuration protocol, which we call *Dynamic Configuration Protocol (DCP)* [5]. Packet errors are firstly recovered inside Local Groups using a receiver-initiated approach. Missing data units are requested from the sender or a higher level Group Controller only if not even a single member of the Local Group holds a copy of the missing data unit. Otherwise, errors will be recovered by local retransmissions. Full reliability and efficient buffer utilization are ensured by a novel, three-state acknowledgment scheme.

All these approaches are based on the principle of subgrouping. Receivers are divided into separate subgroups, each of them represented by a Group Controller. The subgroups are arranged into a multi-level hierarchy defining a so-called *Virtual Group Structure*. The placement of controllers and the arrangement of subgroups will strongly influence the efficiency of multicast data transfer. While a certain Virtual Group Structure may be chosen to reduce network load, another one may increase average throughput.

The paper discusses the impact of group structure on the efficiency of subgroup-based multicast communication. It presents simulation results and illus-

[2] LGMP is based on the Local Group Concept (LGC)

trates in which way receivers should be arranged to optimize average transfer delay and overall network load. Section 2 introduces the MBone scenario all the simulation models are based on. The following subsections investigate the impact of subgroup size and group hierarchy on network load and average transfer delay. Finally, Section 3 presents a protocol for automated establishment of Virtual Group Structures and Section 4 concludes the paper.

2 Performance Evaluation

The multicast algorithms of the Local Group Based Multicast Protocol (LGMP) have been investigated by performing a large number of different simulations. However, the main conclusions are also valid for other subgroup-based or tree-based multicast approaches. Early results have provided feedback to the development and implementation of LGMP. All simulations have been performed using BONeS/Designer, an event-driven network simulation tool by the Alta Group of Cadence Design Systems. Each simulation model comprises a network topology, a protocol state machine and packet formats used for data and control message exchange. The multicast algorithms of LGMP have been compared to a common, sender-based multicast approach. In this approach, receivers address their acknowledgments directly to the multicast transmitter, and necessary retransmissions are performed solely by the sender.

In a first step, several simple scenarios have been modeled to get an idea about the scalability of LGMP. Multicast groups with up to 2000 receivers have been simulated. The simulation results show significant improvements compared to common multicast techniques [4]. Besides the investigation of scaling issues, the effect of different group structures and the placement of Group Controllers is of interest for further development of scalable multicast protocols. Therefore, more complex simulation models based on a real MBone scenario have been developed. The results of these comprehensive simulations allow us to draw conclusions on the suitability of different virtual group hierarchies. A recommendation on how to subgroup the members of a multicast group is given. These recommendations are also valid for tree-based multicast protocols such as RMTP or TMTP.

2.1 The Simulation Scenario

A major claim of the project is to get authentic and realistic statements on the impact of group structure on overall multicast performance. Instead of analyzing simple and abstract network scenarios, all our simulation models are based on packet loss data collected in the MBone by Yajnik, Kurose and Towsley [8]. The MBone scenario described in their paper and all the protocol mechanisms of LGMP are modeled with full details. This results in quite complex simulation models, each of them running more than two days on a SUN SparcStation 20 with two processors.

The scenario used in our simulations consists of a packet source located in California and of 11 receivers $r_{0,0}$ through $r_{10,0}$ distributed all over the world.

The hosts are connected via the MBone. The average packet loss rate of each link, as reported in [8], is inscribed in Figure 1. The bold lines represent the *backbone* links of the network. These links form the base of the multicast tree and traverse most of the distance in the network. All the other branches are on the *edge* of the multicast routing tree and connect receivers to the backbone. However, the edge branches may cross multiple multicast routers before reaching the receivers.

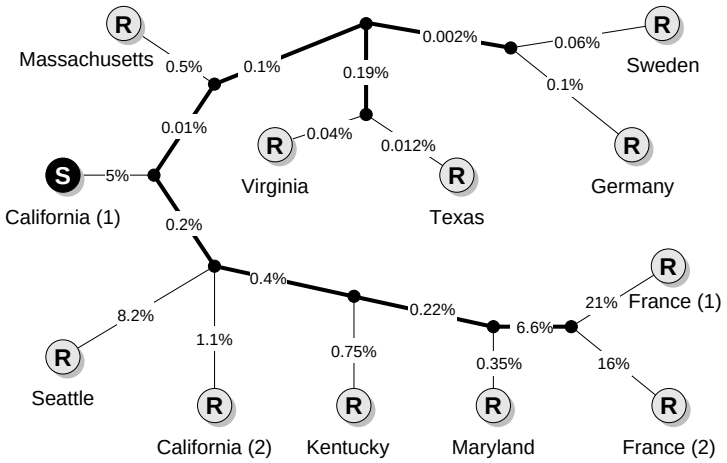

Fig. 1. MBone scenario used for simulations (based on [8])

In order to increase the overall group size, we assume 19 additional group members $r_{i,1}$ through $r_{i,19}$ $(0 \leq i \leq 10)$ in the neighborhood of each (real) receiver $r_{i,0}$. Therefore, the number of simulated receivers is 220. However, the paper always refers to a certain receiving site as the complete set of all 20 simulated receivers. While talking about the receiver in Kentucky, for example, we refer to all the 20 simulated receivers within the area of Kentucky. The additional receivers $r_{i,1}$ through $r_{i,19}$ are modeled using the error probability and the transfer delay given by the receiver $r_{i,0}$. This assumption is quite realistic because the receivers $r_{i,1}$ through $r_{i,19}$ are assumed to be in the same local region and to be connected to the same branch of the multicast transmission path as receiver $r_{i,0}$. The expansion of group size does not effect subgrouping of receivers, because each of the 20 receivers will be assigned to the same Local Group.

All the other parameters are set according to the results presented in [8]. The transmission delay, which is not given in the paper, is set to 40 msec for packets transmitted within the USA and to 100 msec for packets transmitted between hosts in the USA and receivers in Europe. Transmission errors are simulated by Markov models using the average packet loss rate and the average burst length for input. The latter one is set to 2 packets which is in conformity with the results presented in [8].

The packet source transmits 1000 data units per second with a constant size of 2000 Byte per packet. This results in a data rate of about 2 MByte/sec. The simulation time is set to 40 seconds for each simulation. Hence, the sender emits 40000 data packets during the whole simulation. Note that this amount includes all the retransmissions performed by the sender. A status request is sent after each hundredth data packet.

Our simulations examine the average transfer delay and the overall network load. Normally, the *network load* is given by the number of packets traveling across the backbone links. However, the total number of data packets transmitted in our simulations is always set to 40000, as explained before. Therefore, the network load is given relatively to the number of packets originally sent by the transmitter. This relation strongly depends on the number of retransmission performed by the sender. An increase in the number of retransmissions always results in an increase of overall network load. Therefore, the relative value is a good measure for network load. *Transfer delay* is the average time between sending a data packet and its successful delivery to the receiver process. The transfer delay is determined by using a time stamp within each data packet. The time stamp is set by the transmitter on sending a packet for the first time. Because system time is globally synchronized in our simulation models, the transfer delay can easily be calculated by evaluating the time stamp of incoming data packets.

2.2 Impact of Subgroup Size

In order to investigate the impact of group size on multicast performance, subgroups of different size have been defined and evaluated. The global multicast group has been split into separate Local Groups, whereby different subdivisions vary in the maximum distance n allowed between two members of a subgroup. The metrics *Hop Count* and *Loss Probability* have been used to define this distance. The algorithm used to subdivide the receivers and to arrange them into subgroups is the following:

1. Define the maximum distance n between all the members of a subgroup and determine all the possible subdivisions.
2. For a given maximum distance, the solution resulting in a minimum number of subgroups is chosen.
3. If there are multiple subdivisions resulting in the same minimum number of subgroups, choose a solution with homogeneous subgroup sizes.

An example for each of these rules is given to illustrate the algorithm:

1. The receivers in *Massachusetts*, *Virginia*, *Texas*, *Sweden* and *Germany* are combined in a single subgroup if the maximum distance within a subgroup is defined to be four hops. A maximum distance of three hops results in three separate subgroups, namely {*Massachusetts*}, {*Virginia*, *Texas*} and {*Sweden*, *Germany*}.

2. There are two possible subdivisions of the receivers in *Seattle*, *California (2)*, *Kentucky*, *Maryland*, *France (1)* and *France (2)* if a maximum distance of three hops is given:
   - {*Seattle*, *California (2)*, *Kentucky*} and {*Maryland*, *France (1)*, *France (2)*}
   - {*Seattle*, *California (2)*}, {*Kentucky*, *Maryland*} and {*France (1)*, *France (2)*}

   According to the second rule, the first subdivision resulting in two subgroups is chosen.
3. If the maximum distance is set to four hops, the receivers in *Seattle*, *California (2)*, *Kentucky*, *Maryland*, *France (1)* and *France (2)* could be divided into a subgroup of size four and a subgroup of size two. Another solution is to split them into two subgroups of size three. According to the third rule, the latter solution will be chosen.

Table 1 lists the resulting subdivisions according to these rules. The name of each subdivision indicates the metric that has been used to build up the subgroups: Subdivisions including a roman *I* have been built using the hop count metric. Subdivisions including a roman *II* are based on the loss probability. An *f* at the end of the name indicates a flat virtual group structure, whereby all the subgroups are directly attached to the sender. Later on, an *h* indicates a multi-level hierarchy. The paper makes intensive use of these naming conventions in order to explain the simulation results. Table 1 also includes the maximum allowed distance *n* between two members of a subgroup of each subdivision.

The impact of group size on multicast performance has been investigated using flat virtual group structures. All subgroups were directly connected to the sender. Multi-level hierarchies are discussed in the following section. In addition, a standard multicast simulation has been performed to illustrate the benefits of LGMP. In the standard multicast simulation all 220 receivers address their status reports directly to the sender. Furthermore, they request missing data packets always from the sender.

Figure 2 presents the overall number of data packets sent by the transmitter in relation to the number of original data packets. This value allows conclusions concerning the network load caused by reliable multicast transport. The average transfer delay observed by the receivers is given in Figure 3. It illustrates the delay of packets to all receivers within each subgroup as well as the average delay of packets all the members of the global multicast group.

The impact of group size on transfer delay and network load can be derived from the results of simulation *I.a-f* through *I.d-f*. As presented in Figure 2 and 3, the global network load as well as the average transfer delay decrease with an increase of average subgroup size. Especially receivers in *Sweden*, *Germany*, *France (1)* and *France (2)* observe a higher transfer delay in smaller subgroups.

Note that the existence of small subgroups results in a higher transfer delay than for the standard multicast communication. This can be seen in the simulation *I.d-f* and is due to the mechanism of local retransmission and local

Table 1. Simulated subdivisions to examine the impact of group size

Scenario	n	#LGs	Subdivision
I.a-f	5	2	$LG1$: { *Massachusetts*, *Virginia*, *Texas*, *Sweden*, *Germany* } $LG2$: { *Seattle*, *California (2)*, *Kentucky*, *Maryland*, *France (1)*, *France (2)* }
I.b-f	4	3	$LG1$: { *Massachusetts*, *Virginia*, *Texas*, *Sweden*, *Germany* } $LG2$: { *Seattle*, *California (2)*, *Kentucky* } $LG3$: { *Maryland*, *France (1)*, *France (2)* }
I.c-f	3	5	$LG1$: { *Massachusetts* } $LG2$: { *Virginia*, *Texas* } $LG3$: { *Sweden*, *Germany* } $LG4$: { *Seattle*, *California (2)*, *Kentucky* } $LG5$: { *Maryland*, *France (1)*, *France (2)* }
I.d-f	2	7	$LG1$: { *Massachusetts* } $LG2$: { *Virginia*, *Texas* } $LG3$: { *Sweden*, *Germany* } $LG4$: { *Seattle*, *California (2)* } $LG5$: { *Kentucky* } $LG6$: { *Maryland* } $LG7$: { *France (1)*, *France (2)* }
II.a-f	1,5%	6	$LG1$: { *Massachusetts*, *Virginia*, *Texas*, *Sweden*, *Germany* } $LG2$: { *Seattle* } $LG3$: { *California (2)* } $LG4$: { *Kentucky*, *Maryland* } $LG5$: { *France (1)* } $LG6$: { *France (2)* }

acknowledgment processing. In large subgroups, the ability to recover from errors by local retransmissions is quite high. It is likely that one of the members has correctly received the missing data packet. Therefore, it is not necessary to request it from the sender. Especially receivers far away from the sender, such as *Sweden* and *Germany*, gain by local error recovery.

On the other hand, it is not likely that errors can be recovered locally in small subgroups. This is especially true if all the members of a subgroup are connected to a single subnet. In this case, packet errors are likely to be spatially correlated making it impossible to perform local error recovery. Instead, the controller of such a subgroup has to request missing data packets from the sender or its higher level group controller. This effect can be seen evaluating the average transfer delay of receivers *France(1)* and *France (2)* in simulation *II.a-f* (see Figure 3).

One may wonder that subgroup-based multicast communication can even be slower than standard multicasting. However, this drawback is due to the

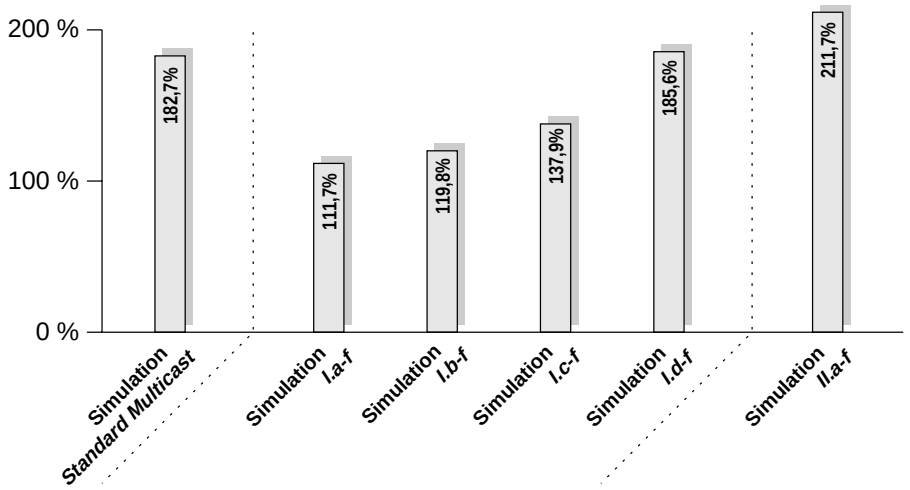

Fig. 2. Global network load for flat group structures

necessity to collect and process all the status reports of subgroup members. Therefore, retransmission requests directed to the sender are delayed until all the members of a subgroup have negatively acknowledged the missing data. To overcome this drawback, a new version of LGMP incorporates periodic status push instead of controller-initiated status poll [5].

The results show that small subgroups should be avoided in flat hierarchies. Multicast performance improves with an increasing number of subgroup members. However, there is always a trade-off between the size of a subgroup and the processing capacity required to handle it. Too many members will result in a controller implosion and will degrade multicast performance. The optimal size of a subgroup strongly depends on the processing capacity of its controller and of the error characteristics of its members. A controller should handle as many receivers as possible. Therefore, a mechanism is required to monitor the load of each controller and to dynamically adapt the virtual group structure to the current network load and receiver state. The Dynamic Configuration Protocol (DCP), presented in [5] and in Section 3, provides such a mechanism.

2.3 Examination of Different Group Hierarchies

To examine the benefits of multi-level hierarchies additional simulations have been performed. They have been based on the subdivisions presented in the former section. However, the subgroups have now been arranged in a multi-level hierarchy according to the given network topology (see Figure 1) and depending on the distance metric (Hop Count or Loss Probability). The resulting group hierarchies are given in Table 2.

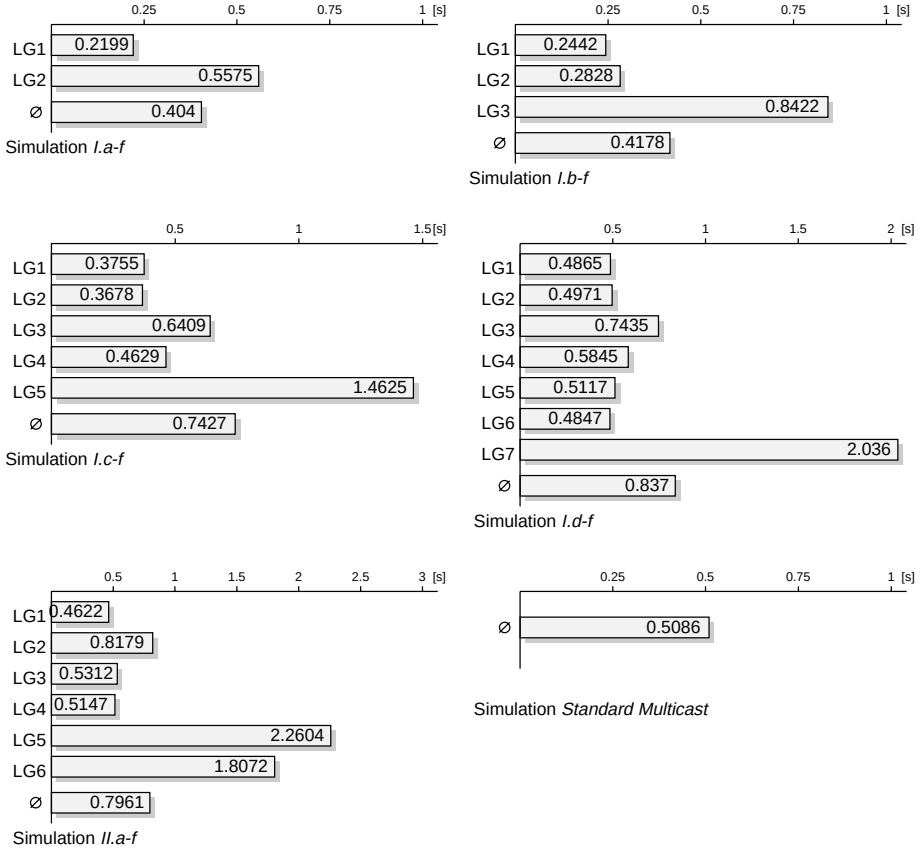

Fig. 3. Average transfer delay for flat group structures

Figure 4 presents the measured network load. As previously explained, it is given relatively to the number of packets originally sent by the transmitter. The average transfer delay is presented in Figure 5.

A comparison of simulation *I.c-f/h* and *II.a-f/h* demonstrates that multi-level hierarchies yield better performance than flat group structures. The evaluation of average transfer delays in group *LG2* and *LG3* shows better performance for the hierarchical group structure in *I.c-h* compared to the corresponding flat structure in *I.c-f*. This is also valid for the groups *LG5* and *LG6* in the simulations *II.a-f/h*. It is remarkable that the mentioned subgroups are exactly those placed in the second level of the hierarchy. These subgroups are not directly attached to the sender. Instead, they address their status reports to a higher level controller.

It is also interesting that in simulation *II.a-h* the receivers in Europe (members of *LG1*, *LG5* and *LG6*) observe nearly the same average transfer delay than receivers in the USA. This illustrates the benefits gained by multi-level hierarchies. Instead of addressing requests for retransmissions directly to the sender in

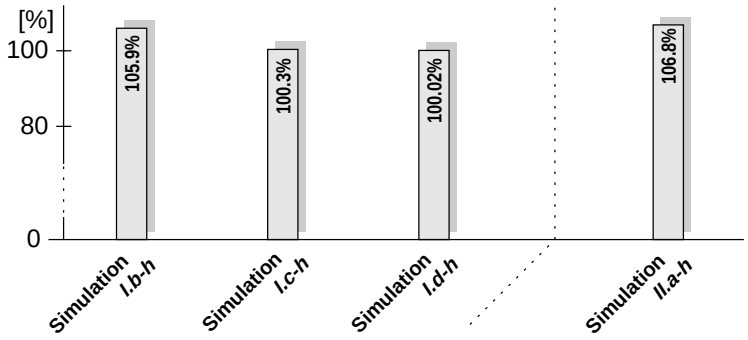

Fig. 4. Global network load for multi-level hierarchies

California, controllers in a multi-level hierarchy get missing data packets from a nearby, higher-level controller.

Figure 5 illustrates that the average transfer delays observed by the groups *LG1* and *LG4* in simulation *I.c-f/h* and by the groups *LG1*, *LG2*, *LG3* and *LG4* in simulation *II.a-f/h* decrease if the subgroups are arranged in a hierarchy. Furthermore, there is an assimilation of average transfer delay in those subgroups that are arranged in a multi-level hierarchy (such as the groups *LG1*, *LG2* and *LG3* in *I.c-h*).

A look at the global network load shown in Figure 4 reveals similar results as explained in the former section. The definition of multi-level hierarchies results in a decrease of network load. Retransmissions between controllers at different

Table 2. Group structures used to examine the impact of group hierarchy

Scenario	I.b-h	I.c-h	I.d-h	II.a-h
Parent of LG 1	California(1)	California(1)	California(1)	California(1)
Parent of LG 2	California(1)	LG1	LG1	LG3
Parent of LG 3	LG2	LG1	LG1	California(1)
Parent of LG 4		California(1)	California(1)	California(1)
Parent of LG 5		LG4	LG4	LG4
Parent of LG 6			LG4	LG4
Parent of LG 7			LG4	

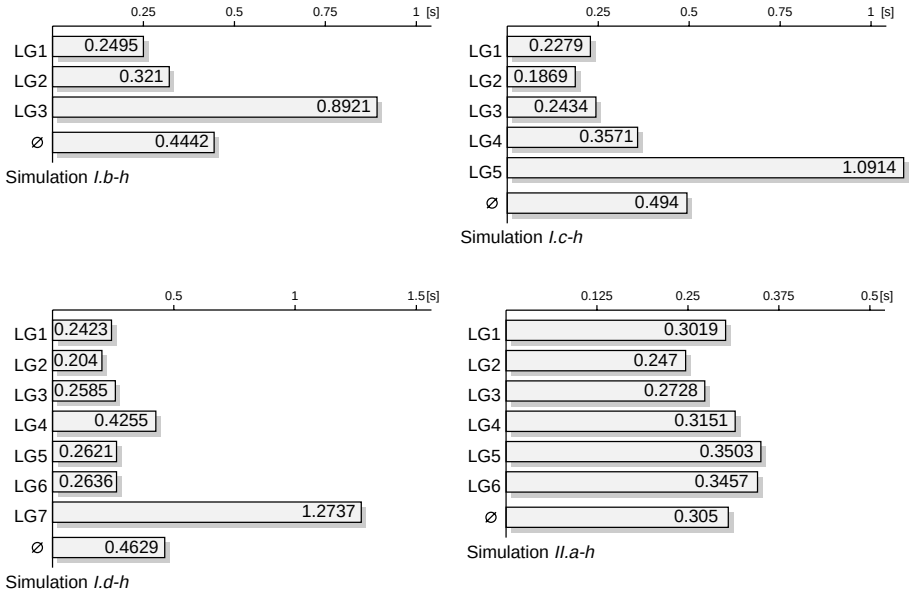

Fig. 5. Average transfer delay for multi-level hierarchies

levels reduce the number of data packets to be retransmitted from the far away sender in California.

The simulation results exhibit two essential conclusions on multi-level group structures:

- There is an assimilation of multicast performance in hierarchically arranged subgroups.
- The arrangement of a few subgroups in a hierarchy results in a better performance for all receivers.

These results are due to local error recovery and local acknowledgment processing. Only those packets that are lost by all the receivers of a subgroup will be requested from the sender. In a multi-level hierarchy, members of a subgroup could also be controllers of a lower-level subgroup. Therefore, the mechanism of local error recovery continues recursively. Only those packets missed by all the members of a subgroup and all the members of its children have to be retransmitted by the sender (or a higher level controller). This results in an assimilation of the performance and decreases the total number of packets to be retransmitted by the sender. Network load and average transfer delay are much less than in flat virtual group structures.

A closer look on the group *LG1* in *I.b-f* and in *II.a-f* (which contain both the same set of receivers) exhibits another interesting fact. Subgroups with relative high packet loss rates (such as the groups *LG5* and *LG6* in *II.a-f*) make the performance in all the other groups worse. Performance is decreased due to the

involved error control and congestion avoidance mechanisms. LGMP, as well as other multicast protocols, implements a rate-based congestion control with multiplicative decrease and additive increase of transfer rate. According to the defined algorithm, a sender will reduce its transfer rate due to status reports from subgroups with high packet loss rate. In addition, performance is decreased due to high load at the sender. While the sender performs retransmissions requested by subgroups suffering from high loss rates, all the other receivers are waiting for new data to arrive. This argument is confirmed by the network load measured for both simulations. Therefore, subgroups with a high loss probability should be attached to another subgroup rather than directly to the sender.

3 The Dynamic Configuration Protocol (DCP)

The results of our simulations provided feedback to the design and the implementation of a protocol named *Dynamic Configuration Protocol (DCP)* [5]. DCP provides mechanisms for an automated establishment of virtual group structures and for dynamic reconfiguration in accordance with the current network load and group membership. No manual administration is necessary. The definition of subgroups is based on a combination of multiple metrics depending on the QoS requirements of the user. DCP is self-organizing and tolerant with respect to failing controllers.

3.1 Expanded Ring Advertisement

Each Group Controller periodically sends packets of type LG_ADVERTISE to announce its existence. These messages are sent using a separate multicast address. Communication participants listen to the group-specific DCP address and use received advertise messages to identify existing Group Controllers. Advertise messages contain information that allows receivers to select the most appropriate Group Controller according to their requirements. By default, an advertise message includes the smoothed error probability of a Group Controller, the number of receivers currently controlled by the Group Controller as well as the multicast address of the represented Local Group. Optional fields have been defined to include additional information, for example a timestamp for the calculation of transfer delay between Group Controller and receiver.

There is some kind of trade-off between network load caused by advertise messages and the time required to react upon dynamic changes in group structure. Frequent sending of advertise messages ensures short reaction times while increasing the network load. Moreover, the visibility of Group Controllers is determined by the scope of their advertise messages. The larger the TTL value of outgoing advertise messages, the more receivers will be able to identify a Group Controller. On the other hand, advertise messages should preferably be limited to a local scope in order to avoid a global flood of control traffic.

To deal with these contrary requirements, we have designed a new mechanism called *Expanded Ring Advertisement*. Group Controllers send their advertise messages with dynamically changing TTL values according to Table 3. The

first message is sent with a scope of 15 (scope 'Site'), the second one with a value of 31 (scope 'Region'), etc.

Table 3. TTL values used to send advertise messages

Interval No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
TTL	15	31	15	63	15	31	15	127	15	31	15	63	15	31	15	254	15	31	15

Receivers within a scope of 15 will get each of the advertise messages. If the distance of a host is between 16 and 31 hops, it will receive every second advertisement. This scheme continues in a way that every 16^{th} advertise message will be distributed worldwide. The Expanded Ring Advertisement ensures that the frequency of advertise messages exponentially decreases with increasing scope. Therefore, the scheme reduces network load while allowing short reaction times upon changes within the local scope of a receiver. The Expanded Ring Advertisement could also be used to estimate the number of hops between a receiver and a Group Controller.

3.2 Selection and Placement of Group Controllers

Once the service user has issued a listen request, a receiver initializes an association control block. Each of these blocks contains an entry named redirect, which is undefined at startup. This entry will identify the controller to which receivers should deliver their status reports. While the value of redirect is undefined, LGMP will address all status reports to the data source.

With the establishment of an association control block, the receiver activates an initialization timer named INIT-TIMER and changes from the inactive to the pending state. It joins the group-specific DCP group and, therefore, stimulates the transmission of an IGMP Host Membership Report. Now, the host is set up to receive packets addressed to the global DCP group, and it buffers all the information obtained from received advertise messages. After expiration of timer INIT-TIMER, a receiving DCP instance evaluates the buffered information, selects one of the discovered Group Controllers, sets the redirect entry of the association control block to the address of the chosen Group Controller, and changes to the active state. While being in active state, a receiver will continue to process advertise messages and to update the redirect entry dynamically.

If no appropriate Group Controller could be found according to the application requirements, the joining receiver has two possibilities. On one hand, it could attach itself directly to the Local Group represented by the multicast transmitter. In this case, the redirect entry of its association control block will be undefined and LGMP will address all status reports to the multicast transmitter. On the other hand, it could establish a new Local Group and appoint itself as Group Controller of the new subgroup. One of the identified Group Controllers

or the multicast transmitter itself will be defined to be the parent of the newly established subgroup. All reports about the status of the new Local Group will be addressed to the parent Group Controller, thus building a group hierarchy.

Initially, it is the founder of a Local Group which will become the Group Controller. Due to the joining and dropping out of receivers, the group structure has to be reconfigured dynamically during the lifetime of an association. It might be beneficial to split a growing Local Group or to merge several waning subgroups. In addition, a joining receiver might be a better Group Controller than the current one due to its network connection or its processing capacity. The following section describes the scheme used to perform such a dynamic reconfiguration of the global group structure.

3.3 Dynamic Reconfiguration of Local Groups

As receivers and Group Controllers may join and leave during the lifetime of a connection, it is necessary to adjust the placement of Group Controllers dynamically according to the current group status, the current network load, and the current characteristics of each communication participant. For example, it could be advantageous to place the Group Controller in the center of a Local Group. Various schemes based on different criteria could be used to determine the optimal Group Controller among all the members of a Local Group.

Besides the move of Group Controllers, the splitting and merging of Local Groups might become necessary due to changes in group membership. The burden of acknowledgment processing and of doing local retransmissions has to be distributed fairly among all the Group Controllers, thus, resulting in a well-balanced group structure.

Receivers use information contained in LG_ADVERTISE messages to maintain a table of reachable Group Controllers. On receiving an advertise message, a host will add a new entry to the table or update an existing one. Each entry represents a Group Controller and indicates its error probability, the estimated number of hops between receiver and Group Controller as well as size and multicast address of the Local Group. In case there is more information about the characteristic of a Group Controller included in its advertise messages, this information will also be added to the table (e.g. transfer delay or carrier fees). While updating their table, Group Controllers ignore their own advertise messages.

Each entry is valid for a time interval T_{val}. When the timer expires and no further advertise message of a certain Group Controller has been received within the last time interval, receivers will delete the corresponding entry in the table. Therefore, each host has an up-to-date view on active Group Controllers, their identity, and their current status. There is no additional information exchange necessary to keep the table valid. If a Group Controller fails or leaves, the corresponding table entry will time out and be deleted. To ensure correctness of this mechanism, the expiration time T_{val} must be longer than the time T_{adv} between two successive advertise messages of the Group Controller concerned.[3]

[3] The time T_{val} depends on the distance between a receiver and the Group Controller.

We propose to choose $T_{val} > (3 \cdot T_{adv}) + \epsilon$ to counterbalance the loss of two successive advertise messages. If three successive announce messages are lost, a receiver will erroneously delete the corresponding entry. However, on receiving the next advertise message the receiver will add the Group Controller again.

Receivers periodically rate the suitability of their current Group Controller GC_i. If the rating r_j of another Group Controller GC_j is better than the rating r_i, the redirect entry will be set to the address of GC_j. However, the difference between r_i and r_j should be higher than a given threshold to avoid oscillatory changing between Local Groups. A problem might also occur in case a large number of receivers decide to assign themselves to a newly defined Group Controller. Due to an overwhelming number of new members, the new Group Controller might get under heavy load, thus decreasing its rating. This would probably result in a further reconfiguration. Therefore, receivers delay spontaneous reconfiguration for a random time to check the rating of a newly detected Group Controller again.

In addition, a receiver R_i periodically calculates its own rating r_i. If r_i is better than the rating r_j of its current Group Controller by some non-negligible amount, the receiver will establish a new Local Group claiming itself to be a Group Controller. It will start to send LG_ADVERTISE messages to advertise its existence and its current status. Nearby receivers may now join the newly established Local Group, thus relieving their previous Group Controller.

4 Conclusions

Subgroup-based multicast protocols, such as RMTP, TMTP or LGMP, are designed to provide efficient and low-cost error control for multicast applications in wide-area networks. The simulation results presented in this paper give evidence that these schemes succeed in keeping retransmissions relatively local within the wide-area topology and in reducing sender implosion. These properties will allow scaling to large receiver sets in large-scale networks. However, the performance of subgroup-based multicast strongly depends on the virtual group structure used for local error recovery and acknowledgment handling. The paper does not define an exact algorithm for subgrouping multicast receivers, but presents some guidelines to do so:

- *Small subgroups should be avoided in flat hierarchies*: In small subgroups, the probability of being able to recover from errors by local retransmissions is quite low. The more members a subgroup includes, the higher the probability that one of them correctly receives a certain data packet. Instead of introducing overhead for management of small subgroups, it is preferable to combine them into a larger group managed by just a single controller. Nevertheless, the size of the subgroups must not cross a certain threshold due to controller implosion.
- *Multiple subgroups should be arranged in a multi-level hierarchy*: The arrangement of multiple subgroups into a multi-level hierarchy relieves the

sender from acknowledgment processing and reduces the number of retransmissions performed by the sender. While the establishment of multi-level hierarchies comes at the expense of additional complexity, the benefits in multicast performance will justify it.
- *Bad receivers should be fairly distributed among all the subgroups*: A single subgroup that permanently requests missing data packets from the sender degrades overall multicast performance. Therefore, multicast receivers should be arranged in a way that each subgroup includes at least one receiver with low packet loss rate.

The results exhibit the importance of packet loss rate for the establishment of virtual group structures. Rather than solely using a hop count metric, the packet loss rate should be taken into account for the arrangement of receivers into subgroups.

Currently, further experiments are performed in the MBone to find an optimal setting for the various parameters of DCP. An implementation of DCP as well as of LGMP is available for Digital Unix, SunOS and Linux. More information on the project, related research activities and future work could be found at http://www.telematik.informatik.uni-karlsruhe.de/~hofmann/LocalGroups.html.

References

1. S. Deering: *Host Extensions for IP Multicasting.* Internet Request for Comments RFC 1112, August 1989.
2. C. Diot, W. Dabbous, J. Crowcroft: *Multipoint Communication: A Survey of Protocols, Functions, and Mechanisms.* IEEE Journal on Selected Areas in Communications, Vol. 15, No. 3, Pages 277-290, April 1997.
3. S. Floyd, V. Jacobson, S. McCanne, C.-G. Liu, L. Zhang: *A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing.* Computer Communication Review, Vol. 25, No. 4, Proc. of ACM SIGCOMM'95, August 1995.
4. M. Hofmann: *A Generic Concept for Large-Scale Multicast.* International Zurich Seminar on Digital Communication, February 21-23, 1996, Zurich, Switzerland, Ed.: B. Plattner, Lecture Notes in Computer Science, No. 1044, Pages 95-106, Springer Verlag, 1996.
5. M. Hofmann: *Enabling Group Communication in Global Networks.* Proceedings of Global Networking'97, Volume II, Pages 321-330, Calgary, Alberta, Canada, June 1997.
6. V. Kumar: *MBone - Interactive Multimedia on the Internet.* New Riders Publishing, Indianapolis, Indiana, USA, 1996.
7. S. Paul, K.K. Sabnani, J.C.-H. Lin, S. Bhattacharyya: *Reliable Multicast Transport Protocol (RMTP).* IEEE Journal on Selected Areas in Communications, Vol. 15, No. 3, Pages 407-421, April 1997.
8. M. Yajnik, J. Kurose, D. Towsley: *Packet Loss Correlation in the MBone Multicast Network.* IEEE Global Internet '96, London, England, November 20-21, 1996.
9. R. Yavatkar, J. Griffioen, M. Sudan: *A Reliable Protocol for Interactive Collaboration Applications.* Proceedings of ACM Multimedia'95, 1995.