Active Reliable Multicast Strategies for Internet-based Grid Computing Infrastructures

M. Maimour
Laboratoire RESAM, Université Lyon 1
ENS, Bât. LR5, 46 allée d'Italie
69364 Lyon Cedex 07, France

C. Pham
Laboratoire RESAM, Université Lyon 1
ENS, Bât. LR5, 46 allée d'Italie
69364 Lyon Cedex 07, France

Abstract *Reliable multicast requires that all packets are safely delivered to the destinations. Such communication patterns are intensively used in the so-called grid computing infrastructure where many computing resources are spread over the Internet to offer to the users the access to a huge amount of computational power. However, meeting the objectives of reliable multicast is not an easy task and this paper presents active solutions that aim to reduce the end-to-end latency and to increase the achievable throughput of reliable multicast protocols for grid computing over the Internet. More specifically, the performances of several active mechanisms such as NACK aggregation and subcasting are investigated.*

Keywords: Reliable multicast, active networks, grid computing

1 Introduction

Multicast is the process of sending every single packet to multiple destinations. Motivations behind multicast facilities are to handle one-to-many communications in a wide-area network with the lowest network and end-system overheads. In contrast to best-effort multicast, that typically tolerates some data losses and is more suited for real-time audio or video for instance, reliable multicast requires that all packets are safely delivered to the destinations. Desirable features of reliable multicast include, in addition to reliability, low end-to-end delays, high throughput and scalability.

These characteristics fit perfectly the need of the grid computing and distributed computing communities as communications in a computing grid make an intensive usage of data distribution and collective operations (submissions of jobs to computing farms, program and data distribution between computing resources, gather and synchronization barrier operations…). In the past few years, many software that propose grid environments for gaining access to very large distributed computing resources have been made available (e.g. Condor [1], Globus [2], Legion [3], Netsolve [4] to name a few). They all implicitly rely on an efficient underlying data distribution mechanism. In the example of a very simple grid session, an initiator typically sends data and control programs to a pool of computing resources; waits for some results, iterates this process several time and eventually ends the session. Therefore an efficient multicast mechanism dramatically reduce the end-to-end latency for running applications on an Internet-based grid (especially for fine-grained applications) and to minimize the overhead at the source (the source itself may need to gather results and build data for the next computing step). More complex grid sessions put higher demands on the network resources and on the multicast/broadcast communication facilities (cooperation among the receivers, receivers acting as sources for the other receivers, …)

Meeting the objectives of reliable multicast is not an easy task. In the past, there have been a number of propositions for reliable multicast protocols that rely on complex exchanges of feedback messages (ACK or NACK) [5, 6, 7, 8]. These multicast protocols usually take the end-to-end solution to perform loss recoveries. Most of them fall into one of the following classes: sender-initiated, receiver-initiated and receiver-initiated with local recovery protocols. In sender-initiated protocols, the sender is responsible for both the loss detection and the recovery (XTP [5]). These protocols do not scale well to a large number of receivers due to the ACK implosion problem in the source. Receiver-initiated protocols move the loss detection responsibility to the receivers. They use NACKs instead of ACKs. However they still suffer from the NACK implosion problem when a large number of receivers have subscribed to the multicast session. In receiver-initiated protocols with local recovery, the retransmission of a lost packet can be performed by any receiver (SRM [6]) in the neighborhood or by a designated receiver in a hierarchical structure (RMTP [7]). All of the above schemes do not provide exact solutions to all the loss recovery problems. This is mainly due to the lack of topology information at the end hosts.

In this paper, we show the benefits a computing grid can draw from an underlying active reliable multicast service by comparing the performances (mainly the achievable throughput) of several active mechanisms with the non-active case. We assume that the computing resources are distributed across an Internet-based network with a high-speed backbone network in the core (typically the one provided by the telecommunication companies) and several lower-speed (up to 1Gbits/s), with respect to the throughput range found in the backbone, access networks at the edge as depicted by figure 1. For simplicity we represented an access network by a router but practically such networks would contain several routers.

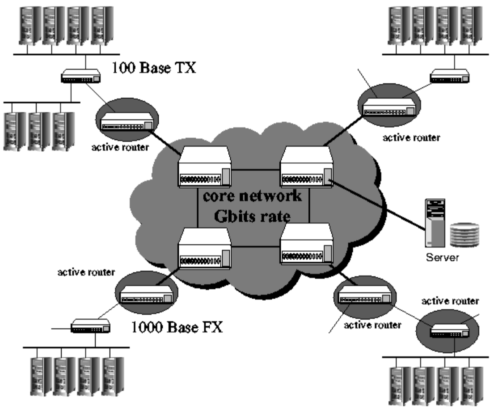


Figure 1. Network model for a simple multicast on a grid.

The rest of the paper is organized as follows. Section 2 presents the active reliable multicast concepts and mechanisms. Section 3 describes the generic protocols used in this paper to compare the various active mechanisms involved in reliable multicasting. Then Section 4 presents the performance study and Section 5 concludes.

2 Active reliable multicast

In active networking, routers themselves play an active role by executing application-dependent services on incoming packets. Recently, the use of active network concepts [9] where routers themselves could contribute to enhance the network services by customized functionalities have been proposed in the multicast research community [10, 11]. These results can be very beneficial to the grid community.

2.1 Active reliable multicast services

Contributing mainly on feedback implosion problems, retransmission scoping and cache of data, these active reliable multicast protocols open new perspectives for achieving high throughput and low latency on wide-area networks:

- the cache of data packets allows for local recoveries of loss packets and reduces the recovery latency.
- the global or the local suppression of NACKs reduces the NACK implosion problem.
- the subcast (partial multicast) of repair packets to a set of receivers limits both the retransmission scope and the bandwidth usage.

For instance, ARM (Active Reliable Multicast) [10] and AER (Active Error Recovery) [11] are two protocols that use a *best-effort* cache of data packets to permit local recoveries. ARM adopts a *global* suppression strategy: a receiver experiencing a packet loss sends immediately a NACK to the source. Active services in routers then consist in the aggregation of the multiple NACKs. In contrast, AER uses a *local* suppression strategy inspired from the one used by SRM and based on local timers at the receivers. In addition, an active router in ARM would send the repair packet only to the set of receivers that have sent a NACK packet (subcast). In AER, the active router simply multicasts the repair packet to all its associated receivers.

2.2 Defining and modeling an active reliable multicast grid session on the Internet

In this study, we only consider active routers at the edge of the core network. This is for two reasons:

- The core network is reliable. It has been shown in [12] that the losses occurred mainly at the edges of the core network: at the source link and at the tail links.
- The core network is a very high-speed network. Adding complex processing functions inside the core network will slow down the packet forwarding functions.

The lower-speed access networks of figure 1 may contain a complete hierarchy of routers that involves several routers. In principle, the nearest router to the backbone is a good location for installing the active services: the cache of packets can serve for several local computing resources for instance. However, it is also possible to have several active routers in the access network (lower right access network in figure 1) and closer to the computing resources to allow for a hierarchical cache of packets or for an efficient, tree-based, subcast mechanism or for a very early filtering of the redundant NACK packets. . . .

A simple grid session usually involves an initiator (source) and a pool of computing resources (receivers).

Communication from the source to the receivers are multicast communications. A receiver usually sends back the results to the source with an unicast communication and generally no communications between receivers are needed. The network model depicted in figure 2 can represent the previously described simple grid session: one source multicasts data packets to R receivers through a packet network composed of a fast core network and several low- to medium performance edge access networks (typical of the Internet). We will call *source link* the set of point-to-point links and traditional routers that connects the source to the core network. Similarly, a *tail link* is composed of point-to-point links and routers connecting a receiver to the core network.

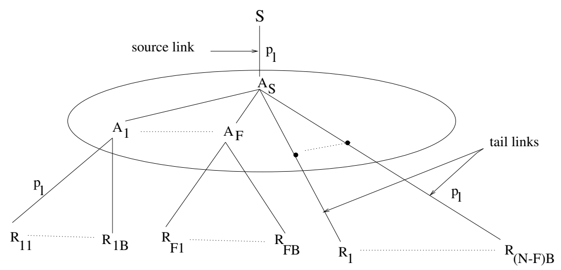


Figure 2. Network model for a simple multicast session on a grid.

Active routers are associated to the tail links (the low- to medium-performance Internet links). However, it is possible that not all routers implement active services and we will consider that F routers among the N possible are active routers, $0 \leq F \leq N$ (Fig. 2). The source has an associated active router, noted A_S. Each active router A_i is responsible of B receivers $R_{i1}, \cdots, R_{iB}$ forming a local group (a local computing resource pool). A receiver associated with an active router is said *linked*. The other receivers are said *free*. For the loss model, we will consider that the core network is reliable. For the other links (the source link or the tail links), the loss probability is noted p_l (this probability includes the losses due to congestion). Therefore, the end-to-end probability of a packet loss perceived by a receiver is $p = 1-(1-p_l)^2$. The losses at the source link are assumed to be temporally independent and those at the tail links are assumed to be mutually independent.

3 Generic active reliable multicast protocols

In this section, we describe 5 generic reliable multicast protocols. These protocols are called S_1, S_2, S_2^s, S_3 and S_3^s. All these protocols benefit from the cache of packets at the active routers (local recovery). However, they differ from each other in the strategy for the NACK suppression and whether subcast is used or not. We introduce them in order to compare the performance of the various active mechanisms for data distribution on an Internet-based grid.

3.1 Description of protocol S_1

S_1 uses the global suppression of NACK packets. A receiver experiencing a packet loss sends immediately a NACK to the source. The active routers have in charge the aggregation of NACK packets in order to forward only one NACK to the source. S_1 has the following properties:

- the source multicasts data packets at the multicast address subscribed to by all the receivers.
- upon reception of a data packet, an active router stores the packet in its cache, if possible, and forward it downstream in the multicast tree.
- upon detection of a packet loss, a receiver sends immediately a NACK towards the source and sets a timer.
- upon reception of a NACK packet, an active router sends the corresponding repair packet, if available, to all the receivers composing its local group. Otherwise, it sends the NACK to the source.
- upon reception of a NACK packet, the source multicasts the repair packet to all receivers at the multicast address.

3.2 Description of protocols S_2 and S_2^s

S_2 uses the NACK local suppression strategy. The receivers wait for a random amount of time prior to send a NACK to the source. The goal is to generate only one NACK per group of receivers in the multicast tree. S_2 works as follows:

- the source multicasts data packets at the multicast address subscribed to by all the receivers.
- upon reception of a data packet, an active router stores the packet in its cache, if possible, and forward it downstream in the multicast tree.
- upon detection of a packet loss, a receiver waits for a random amount of time. Only then it would send a NACK packet to the source and sets a timer.
- when a receiver is waiting to send a NACK, the reception of a similar NACK, from its active router, would make the receiver to cancel its NACK, set a timer and behave as if it has sent the NACK itself. If during the waiting time it receives the repair packet, then the receiver quits the recovery process.

- upon reception of a NACK from downstream, an active router multicasts the repair packet, if available, to its local group. Otherwise, it multicasts the NACK to both the source and its local group, excepting on the NACK incoming link. The desire behavior is to perform the NACK local suppression.

- upon reception of a NACK packet, an active router sends the corresponding repair packet, if available, to all the receivers composing its local group.

- upon reception of a NACK packet, the source multicasts the repair packet to all the receivers.

S_2^s is identical to S_2 but also benefits from the subcast facility from the source. In this case, subcast refers to the ability to send the repair packet only to the set of receivers that have experienced a packet loss (multicast with a TTL scoping is usually used but is less efficient). Since it is difficult for active routers that already perform the local NACK suppression to also implement a subcast service (they are unable to know the identity of the receivers that have experienced a loss since they do not receive all the generated NACKs in their local group), only the free receivers will benefit from the subcast directly from the source.

3.3 Description of protocols S_3 and S_3^s

Active routers that globally suppress NACKs as in S_1 can easily implement a subcast service because they already know the identity of the receivers that have experienced a loss (they received all the corresponding NACKs). We define S_3, which is similar to S_1 in performing a global NACK suppression strategy, that also implements the subcast service within active routers in addition to the NACK suppression service. We also define S_3^s that behaves as S_3 but also benefits from the subcast facility from the source.

4 Analysis of active reliable multicast

In this section, we present the analytical results for the generic protocols S_1, S_2, S_2^s, S_3 and S_3^s. Several parameters have an impact on the performances of an active reliable multicast protocol on a grid. The most important are: (i) the density of active routers, (ii) the processing power of active routers and, (iii) the amount of cache memory at active routers. We will not consider the third item in this study and will always assume that there is enough memory or disk space for caching the packets in the active routers. For the density of active routers, it is also interesting to look at particular cases where $F = 0$ and $F = N$.

We will first consider that active routers have the same processing time than receivers, then we will study the impact of varying the processing time of active routers. The benefit of the subcast and the influence of active routers density in the model will be considered too. In this first attempt the achievable throughput derived from the processing time within each network component will be used as the performance criteria. For the throughput computation the packet size is set to 1024 bytes for all protocols.

The whole analytical models are beyond the scope of this paper but the reader can refer to [13] for the mathematical details. We will only describe below the analysis of the processing requirements at the source for S_1 to show the general methodology we used with each network component. Table 1 shows the useful notation used in the example; a more complete description can be found in [13].

4.1 Example for the analysis of S_1

X^{S_j}	the total processing time per packet at the source for S_j protocols, $j = 1..3$.
X_p, X_n	respectively, the processing time for sending a data packet and to receive a NACK packet.
M_S	number of retransmission of a packet at the source until all active routers and all free receivers have correctly received the packet.
M_a	number of retransmission of a packet at the source until an active router has correctly received the packet.
M_r	number of retransmission of a packet at the source until a receiver (linked or free) has correctly received the packet.

Table 1. Notations used in the example.

We begin by writing that a data packet is sent M_S time until all active routers and free receivers have correctly received it. The source receives for this packet $F\ (M_a - 1)$ NACKs from active routers and $(N-F)B\ (M_r-1)$ NACKs from free receivers. The processing time per packet at the source can therefore be written:

$$E[X^{S_1}] = E[M_S]\, E[X_p] + (F(E[M_a]-1) + (N-F)B\ (E[M_r]-1))\, E[X_n] \quad (1)$$

Replacing $E[M_a]$ and $E[M_r]$ by their expressions [13] gives:

$$E[X^{S_1}] = E[M_S]\, E[X_p] + \left(F\, \frac{p_l}{1-p_l} + (N-F)B\, \frac{p}{1-p}\right) E[X_n] \quad (2)$$

By setting $F = 0$ in eq. (2) we have:

$$E[X^{S_1}] = E[M_R]\, E[X_p] + R\, \frac{p}{1-p}\, E[X_n] \quad (3)$$

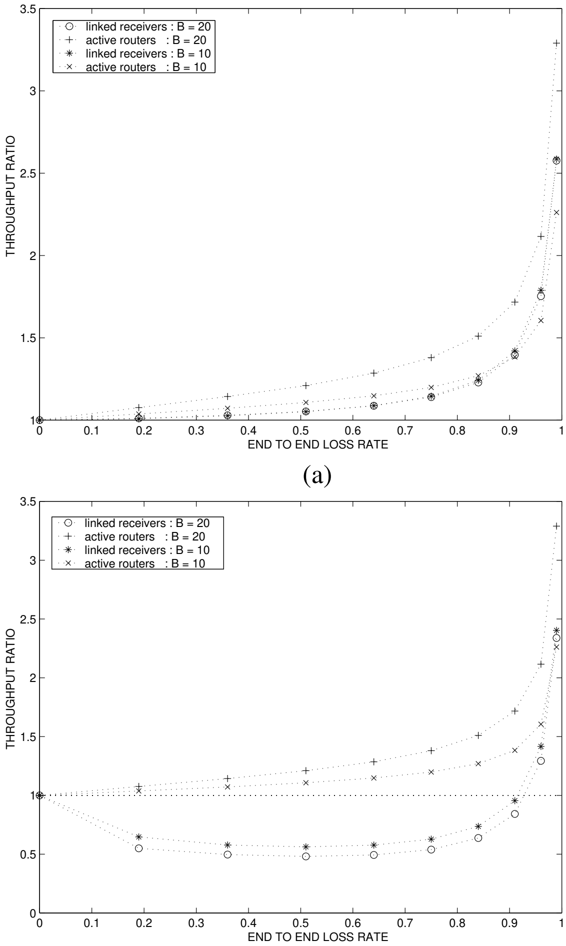
When all routers have active services, we find:

$$E[X^{S_1}] = E[M_A]\, E[X_p] + N\, \frac{p_l}{1-p_l}\, E[X_n] \quad (4)$$

With the processing requirements we can easily derive the achievable throughput in order to compare the performances of the various network components under several strategies.

4.2 Local vs global suppression

In order to deeply compare the two suppression strategies, figure 3 plots the ratio of linked receivers and active routers throughput as a function of the loss probability. Different local group sizes are used.



(a)

(b)

Figure 3. Linked receivers and active routers' throughput ratio: (a) S_2/S_1 (b) S_2/S_3.

Figure 3a plots the throughput ratio for S_2 and S_1. It shows that S_2 (local NACK suppression) always performs better than S_1 (at linked receivers and at the active routers), mainly for high loss rates. In the local suppression strategy, NACKs are suppressed before they reach an active router. Thus justifying the benefit of this strategy for high loss probabilities. Figure 3b plots the throughput ratio for S_2 and S_3. For reasonable loss probabilities, S_3 performs better than S_2 at the linked receivers end. This is because the linked receivers under S_3 benefits from the subcast service. In S_3, a linked receiver receives only once a data packet in contrast with S_2 where a linked receiver could receive more than one copy of the same data packet. Moreover, in S_2, a linked receiver can continue to receive NACKs from its active router every time a receiver in its local group has experienced a loss.

4.3 Benefit of the subcast

The subcast facility has the advantage of unloading the receivers and/or the active routers depending on whether we benefit from this facility from the source or not. To see the benefit of performing the subcast from the active routers associated to the linked receivers, figure 4 plots the throughput ratio at a linked receiver in S_3 and S_1. We can see that the subcast permits a higher throughput at the linked receivers in S_3. The gain obtained with the subcast depends on the local group size and the loss rate. These two parameters gives an idea on the number of receivers that have experienced a loss. Therefore, it is more beneficial to perform the subcast when the local group size is big.

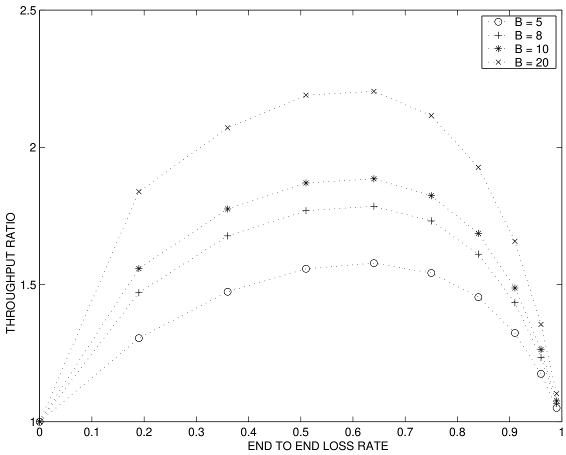
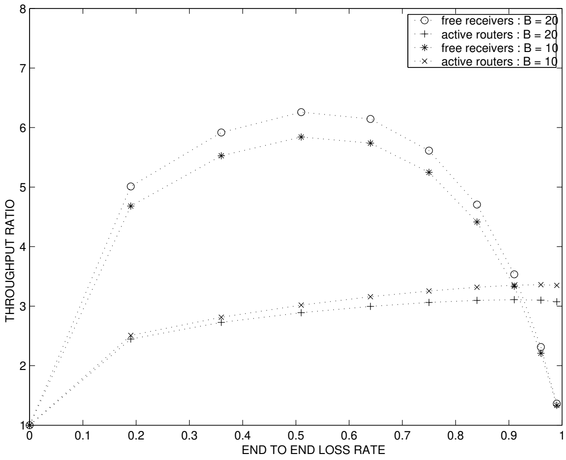
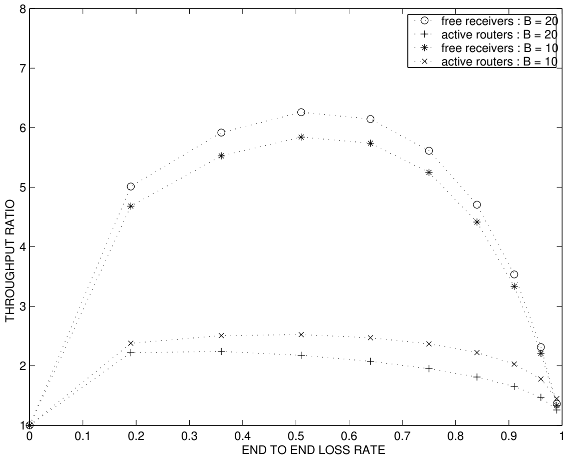


Figure 4. Benefit of the subcast for the linked receivers (S_3/S_2 ratio).

In the case of very low and high loss rates, the subcast does not allow a remarkable gain. For very low loss probabilities (p close to 0) the number of recoveries performed by an active router is very small. Therefore a receiver that does not benefit from the subcast facility is not overwhelmed by receiving several copies of a data packet. For very high loss rates (p close to 1) the number of receivers that have experienced a loss within a local group increases. Thus there are few receivers that are not concerned by the repair packet and the subcast does not show a noticeable gain. In addition to the linked receivers, performing the subcast from the source can unload the active routers and the free receivers (computing resources that does not benefit from active services). The results show that protocols with the subcast from the source perform better in term of the overall throughput than those that do not benefit from this functionality.



(a)



(b)

Figure 5. Benefit of the subcast from the source (a) S_2^s/S_2 and (b) S_3^s/S_3.

To show the benefit of the subcast from the source, Fig. 5a and 5b plot the throughput gain at the free receivers and the active routers as a function of the loss probability for S_2^s and S_2, and for S_3^s and S_3 respectively. Two different different local group sizes are used. It is worth to mention that the number of free receivers ($(N - F)B$) is proportional to the local group size (B). At the free receivers side, we can achieve a gain of 5 and even 6 times higher for a loss rate of 20 % and 50 % respectively. We notice also that similarly to the case of the linked receivers in Fig. 4, the subcast is more beneficial in the presence of a large number of free receivers. Moreover, for very low and high loss rates, the subcast does not permit a noticeable gain. For the active routers, we can see that unlike S_3^s and S_3, S_2^s performs better than S_2 even for high loss rates. This is due to the fact that active routers in S_3^s receive all the NACKs generated in the local group which considerably increases with p. However, S_2 suppresses NACKs before they reach the active routers.

4.4 Active routers density

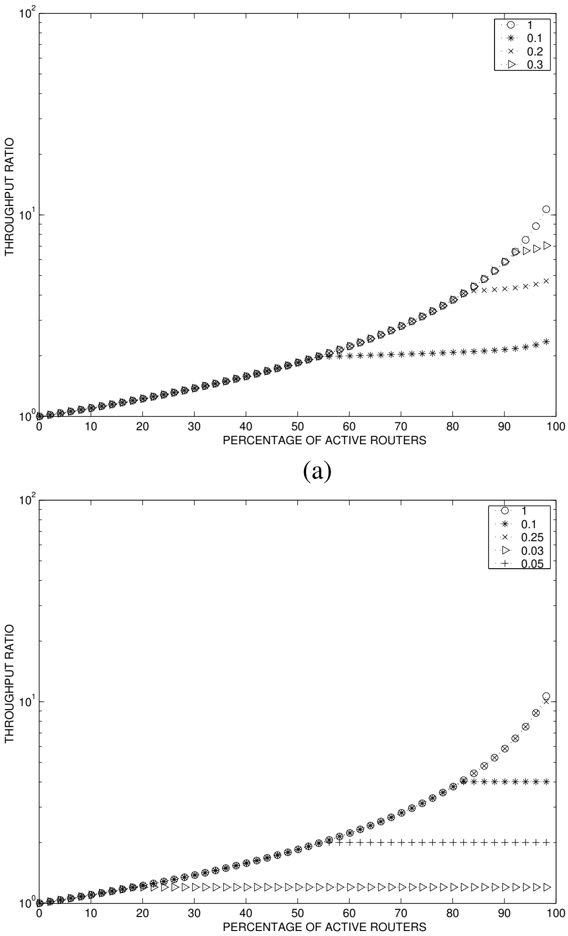
Figure 6 shows the impact of the active routers density on the protocol's performances in term of the overall throughput. The figure plots the overall throughput gain as the number of active routers is increased compared to the no active routers case. Several multiplicating factors to the active routers' processing power are applied. We can see that with the same processing time at the active routers and the receivers, the overall throughput can be an order of magnitude higher if all the receivers are linked. Most interestingly, if the active router's processing power is divided by 10 in S_3 (figure 6a), we can still double the overall throughput provided that 55 % of routers are active. Figure 6b shows that in S_3^s, 55 % of active routers are sufficient to double the performances even when dividing the processing power by 20. Although not shown S_1 and S_2 behave as S_3. S_2^s presents a slightly higher gain than S_3^s for low loss probabilities (eg. 0.05).

5 Conclusions

Distributed computing with expensive computing resources spread over the Internet is a challenging concept. In these Internet-based computing grids, an efficient reliable multicast service can really be a big win for reducing the latency of data and program distribution. Contributions of active routers within the multicast tree can be used for performing additional functionalities such as cache of data, feedback aggregation or subcast. In order to evaluate the potential of these mechanisms for data distribution for the computing grids over the Internet, we proposed five generic protocols: S_1 that uses the global NACK suppression strategy, S_2 that uses local suppression instead, and S_3 that differs from S_1 by the subcast facility. S_2^s and S_3^s behave as S_2 and S_3 respectively but benefit from a subcast facility from the source. All five benefit from the cache of data and use active services located at the edge of the backbone network.

It appeared that local suppression, under the assumption that the random waiting time at a receiver is well-chosen, requires less processing power than global suppression mainly for high loss rates. Therefore a dynamic scheme that would adapt to the loss rate may show interesting results. We have also shown the benefit of the subcast feature which allows a higher and constant throughput.

Thus unloading nodes that benefit from this facility, especially when it is applied from the source. For grid computing, the subcast facility (from the source) appears to be very beneficial, both for linked and free computing resources. Regarding the impact of the active router density on performances, all protocols have the same behavior: the achievable throughput increases as the number of active routers increases. Most interestingly, even with slower processing power at the active routers, increasing their number allows for more performances.



(a)

(b)

Figure 6. Gain in term of the overall throughput achieved by (a) S_3, (b) S_3^s when varying the active routers density. $p = 0.05$, $B = 10$ **and** $N = 1000$**.**

To summarize, active services can really bring an enhanced service to data distribution over the Internet. The grid community can benefit from this technology by incorporating in the network infrastructure specialized active components for efficient reliable multicast services. We are currently analyzing the memory requirements within active routers to propose efficient cache strategies and are implementing these active services on a test-bed.

References

[1] M. Litzkow and M. Livny. Experience With The Condor Distributed Batch System. *Proc. of the IEEE Workshop on Experimental Distributed Systems, 1990.*

[2] I. Foster and C. Kesselman. Globus: A metacomputing infrastructure toolkit. *Intl J. Supercomputing Applications, 11(2):115-128, 1997*

[3] A. Grimshaw, A. Ferrari, F. Knabe and M. Humphrey. Legion: An Operating System for Wide-area computing. *IEEE Computer, 32(5):29-37, May 1999*

[4] M. Beck, T. Moore, J. Plank and M. Swany. Logistical Quality of Service in Netsolve. *Computer Communication, 22(11):1034-1044, July 1999*

[5] XTP Forum. *Xpress Transport Protocol Specification*, March 1995.

[6] S. Floyd, V. Jacobson, and Liu C. G. A reliable multicast framework for light weight session and application level framing. In *ACM SIGCOMM'95*, pages 342–356.

[7] S. Paul and K. K. Sabnani. Reliable multicast transport protocol (rmtp). *IEEE Journal of Selected Areas in Communications*, 15(3):407–421, April 1997.

[8] R. Yavatkar, J. Griffioen, and M. Sudan. A reliable dissemination protocol for interactive collaborative applications. In *ACM Multimedia'95*, November 1995.

[9] D. L. Tennehouse et al. A survey of active network research. *IEEE Communications Magazine*, pages 80–86, January 1997.

[10] L. Wei, H. Lehman, S. J. Garland, and D. L. Tennenhouse. Active reliable multicast. In *IEEE INFOCOM'98*, March 1998.

[11] S. K. Kasera et al. Scalable fair reliable multicast using active services. *IEEE Network Magazine, Special Issue on Multicast*, 2000.

[12] M. Yajnik, J. Kurose, and D. Towsley. Packet Loss Correlation in the MBone Multicast Network. *In Proc. of Global Internet Conference, November 1996.*

[13] M. Maimour and C. Pham. A Throughput Analysis of Reliable Multicast Protocols in an Active Networking Environment. Technical report, http://resam.univ-lyon1.fr/˜mmaimour/Paper/TR/TR01-2001.ps.gz.