

Where is Multicast Today?

(An introduction to the Editorial Note by Neumann, Roca and Walsh on “Large Scale Content Distribution Protocols”)

Ernst W. Biersack
Institut Eurecom
Sophia Antipolis, France
erbi@eurecom.fr

1. INTRODUCTION

Multicast has been a very active area of research in the nineties. Multicast is uniquely qualified to deliver data in a scalable and efficient manner to a large number of receivers. Steve Deering in his thesis showed how to support multicast routing [1]. Since multicast routing was not supported directly by the routers, the first experiments were carried out building overlays such as the Mbone [2]. In the following, innovative applications using multicast were developed. For a survey of the various aspects of multicast and its applications see the book by S. Paul [3].

In multicast communication, the sender sends data to potentially a large number of receivers, as compared to unicast where there is only one receiver. The fact that there are multiple receivers that may experience heterogeneous transmission conditions with respect to bandwidth, loss and delay poses a number of challenges for the design of multicast error control and congestion control schemes.

1.1 Reliable Multicast Data Transfer

Reliable data transfer to a multicast group is difficult since different receivers may lose different packets and need to indicate their loss to the sender. Any reliable data transfer protocol must address the following issues:

- Loss detection: Who detects the loss, the sender or the receivers
- Type of feedback: Positive acknowledgments (ACKs), or negative acknowledgments (NAKs)
- How to send the feedback: via unicast to the sender or via multicast to all participants of the multicast session
- Who retransmits the missing data: In the case of multicast, not only the original sender can retransmit the missing data but also any other receiver or intermediate network node that has a copy of the missing data
- What to retransmit for error recovery: If the sender sends a sequence of packets to a multicast group, different receivers typically lose different packets. For this reason, the retransmission of the *original (lost) data* is often very inefficient. The use of FEC (forward error control) that consists in sending so called *parity packets* for loss repair is much more efficient [4]. Take the simple case where the sender sends 3 packets D1,

D2, D3 to a multicast group consisting of 3 receivers. When each of the receiver loses a single but different packet, the retransmission of original data would require the sender to retransmit all 3 packets D1–D3. Using FEC, the sender transmits a *single parity packet P* that consists of the bit-wise Exclusive OR of the data portion of packets D1–D3. The successful reception of *P* will allow all 3 receivers to reconstruct their missing data packet.

Given these different elements of error control, the designer of a reliable multicast error control protocol has a large number of possibilities. As a consequence, the research community has come up with a variety of reliable multicast error control protocols. My favorite protocol, call it RMC, is one that does (i) loss detection at the receiver, (ii) uses NAKs with probabilistic feedback suppression [5], and has (iii) the sender retransmit parity packets for loss recovery. RMC has a couple of interesting and useful properties: It is purely endsystem based, i.e. it can operate on top of any network that supports multicast or broadcast. In particular, RMC does not require any help from the network for multicast error recovery. RMC also avoids feedback implosion using NAKs and probabilistic NAK suppression and assures efficient data recovery using parities for loss repair.

1.2 Multicast Congestion Control

Besides reliable multicast transmission, the other burning issue to be addressed is multicast congestion control. The congestion control mechanisms built into TCP allow for a fair sharing of the limited bandwidth among the different sessions. Any multicast transmission protocol that competes with TCP sessions for the same bandwidth resources must avoid driving the TCP sessions into starvation. It was therefore required that any multicast transmission should behave “TCP-friendly”.

Many proposals to multicast congestion control are receiver-based and partition the data for transmission in multiple layers, typically each layer on a different multicast address. Each receiver dynamically decides how many layers to subscribe to. A well known protocol in this class is Receiver-driven Layered Multicast (RLM) [6]. However, RLM suffers from a number of problems such as slow convergence, transient periods of high loss, and high load on the routers due to frequent join and leave operations of the receivers.

Despite numerous attempts, the issue of multicast congestion control could not be solved in a satisfactory way. In

my opinion, this may be due to the fact that the problem as originally posed has no solution in a network of routers with FIFO scheduling and tail drop buffer management. For instance, the work of Legout [7] has shown that in a network where each router implements general processor sharing as scheduling mechanism, layered multicast congestion control protocols can be made to work, avoiding the problems observed with RLM in FIFO networks.

In the context of multicast congestion control, Baccelli et al. made some important contributions.

- For an end-to-end congestion control [8] that uses a TCP-like congestion protocol for a one-to-many transmission, the throughput of the multicast session decreases with the logarithm of the number of users even under very optimistic assumptions (homogeneous setting) such as i.i.d. packet service times and light tailed random queuing delays. This result indicates that there are clear limits in terms of group size beyond which multicast with end-to-end congestion control will not scale.
- For the case of reliable multicast communication using *overlays*, where the transmission between two adjacent overlay nodes of the multicast distribution tree is done via TCP and data packets can be buffered in the overlay nodes, Baccelli et al. showed [9] that the overall throughput, independent of the group size, remains strictly positive and converges to the minimum throughput achieved for any of the TCP connections between two adjacent overlay nodes.

Intuitively, this result is due to the fact that for the case of multicast overlays, the different transmissions between adjacent overlay nodes are *decoupled* and can progress at a speed that is only determined by the local characteristics of the path between the two adjacent overlay nodes.

1.3 Context for the Paper by Neumann et al.

The situation with respect to multicast at the end of the nineties was such that (i) The deployment of IP multicast had progressed much slower than initially expected and was limited to a few ISPs and research networks. For the various reasons that hampered the deployment of IP multicast see [10]. (ii) Also, no satisfactory solution for multicast congestion control existed.

In recent years, overlay networks and peer-to-peer based systems started to receive a lot attention and many solutions to overlay multicast tree construction and multicast data distribution using overlays were developed. In fact, as the results of Baccelli indicate, overlays may be the only feasible way to provide multicast to a large number of receivers reachable over the Internet. However, end-to-end multicast transmission is definitely most appropriate for the transmission over broadcast-type networks, where the transmission characteristics to the different receivers are fairly homogeneous.

Over the last ten years, a new broadcast system called Digital Video Broadcast (DVB) has been developed. DVB allows for two types of receivers: Terrestrial TV sets, referred to as DVB-T, or handheld mobile devices, referred to as DVB-H. DVB-H offers reception rates of as high as 15 Mbit/sec and allows to deliver popular content to a large number of users at low cost, which is not the case for today's

cellar GPRS/3G based systems. DVB cannot only be used for TV distribution but also offers a so called "IP Datacast" service where IP packets are encapsulated for transmission over DVB. DVB-H was formally adopted as an ETSI standard in November 2004. Currently, DVB-H trials are carried out in various countries and prototype DVB-H/GPRS handsets exist. For more details see the DVB Web site at www.dvb.org.

In the area of FEC and error correcting codes, exiting developments have happened in the last decade. LDPC (low density parity check) codes, originally invented by Gallager in the sixties and rediscovered in 1995 as well as other sparse graph codes were developed (for a nice survey see chapter 47 of [11]). As explained in the paper by Neumann, Roca and Walsh, sparse graph codes allow for much larger block sizes and much higher encoding and decoding speeds than the well-known Reed Solomon codes.

While the research community turned its attention to overlays, the standardization of building blocks and protocols for reliable end-to-end multicast was pursued within the IETF. The authors of the paper were actively involved in the standardization effort and have also contributed an open-source implementation of some of the building blocks, most notably an LDPC en-/decoder. In the following paper, they give a first hand account of these activities and show how reliable multicast can be used for large scale content delivery.

2. REFERENCES

- [1] S. E. Deering, "Multicast Routing in Internetworks and Extended LANs", *Computer Communications Review*, 18(4):55-64, 1988.
- [2] H. Eriksson, "MBONE: The Multicast Backbone", *Communications of the ACM*, 37(8):54-60, August 1994.
- [3] S. Paul, *Multicasting on the Internet and its Applications*, Kluwer Academic Publishers, 1998.
- [4] J. Nonnenmacher, E. W. Biersack, and D. Towsley, "Parity-Based Loss Recovery for Reliable Multicast Transmission", *IEEE/ACM Transactions on Networking*, 6(4):349-361, August 1998.
- [5] J. Nonnenmacher and E. W. Biersack, "Scalable Feedback for Large Groups", *IEEE/ACM Transactions on Networking*, 7(3):375-386, June 1999.
- [6] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven Layered Multicast", *ACM SIGCOMM*, pp. 117-130, August 1996.
- [7] A. Legout and E. W. Biersack, "PLM: Fast Convergence for Cumulative Layered Multicast Transmission Schemes", *Proc. of ACM SIGMETRICS'2000*, pp. 13-22, Santa Clara, CA, USA, June 2000.
- [8] A. Chaintreau, F. Baccelli, and C. Diot, "Impact of Network Delay Variation on Multicast Sessions Performance with TCP-like Congestion Control", *IEEE Transactions on Networking*, pp. 500-512, 2002.
- [9] F. Baccelli et al., "The One-to-Many TCP Overlay: A Scalable and Reliable Multicast Architecture", *Proc. Infocom 2005*, April 2005.
- [10] C. Diot, B. N. Levine, B. Lyles, H. Kassem, and D. Balensiefen, "Deployment Issues for the IP Multicast Service and Architecture", *IEEE Network magazine special issue on Multicasting*, 14(1):78-88, January/February 2000.
- [11] D. J. C. MacKay, *Information Theory, Inference, and Learning Algorithms*, Cambridge University Press, 2003.