A Clustering Model of the Application-Level Multicast

Gábor Hosszú
Budapest University of Technology and Economics, Budapest

Raymond Pardede
Budapest University of Technology and Economics, Budapest

INTRODUCTION

This chapter reviews the most important fact of the application-level multicast (ALM) and then describes a novel concept of modeling relative density of members called bunched mode and a proposed host-end multicast routing protocol called shortest tunnel first (STF). The bunched mode is based on the thematic multicast concept (TMC), which means that it is a typical multicast scenario where there are a lot of interested hosts in certain institutions, relatively far from each other. This situation is called bunched mode, in which the members of a multicast group are locally in the dense mode, and globally their situation is similar to the sparse mode because these spots are far from each other. The developed analysis tool, NetSim, and the implementation of the TMC, PardedeCAST, are also presented as the tools of this research.

OVERVIEW OF THE APPLICATION LEVEL MULTICAST

Currently there is a fast increasing need for scalable and efficient group communication technology. The multicast is theoretically optimal for such purposes. It can be realized in the Data-link Level, IP level, and Transport/Application level (Hosszú, 2005). However, the IP Multicast has a slow deployment; it has been implemented in the most Operating Systems (OS) and routers, but not widely enabled. That is why the end-host based multicast is emerging, in which each member host duplicates and forwards packets. That solution is called application-level multicast (ALM). ALM is easy to deploy, but less efficient. In the following, the various solutions for the ALM are listed.

The special type of the ALM is the host multicast, which is a hybrid approach. Its goal is to reach a ubiquitous multicast. One of its design requirements is that it should be deployable on the current Internet, which means that the installation of a user-space program is done at end hosts and there is no support required from OS, routers, or servers to enable multicast services. The client applications can create a virtual network called overlay network (shortly overlay) on the top of the Internet. However, the hybrid approach has another design requirement, which is the compatibility with IP multicast to the furthest extent. For that reason, it should use the IP multicast where available, keep the IP Multicast service model, and provide incentive to the future deployment.

Another type of the ALM is the mesh-based protocol. This type creates a mesh for the control plane at first with a redundant topology of the connections between members. After creating the mesh, the algorithm starts to construct a multicast tree. Such protocols are the Narada (Chu et al., 2000), or the Gossamer (Chawathe, 2000).

The opposite of the mesh-based type are the tree-based protocols, where the multicast delivery tree is formed first and then each member discovers others that are not neighboring members and creates control links to these hosts. This solution is suitable for data transferring applications, which need high bandwidth, but are not efficient for real-time purposes. Such protocols are the Yoid (Francis, 2000) and the host multicast tree protocol (HMTP) from Zhang et al. (2002).

APPLICATION-LEVEL MULTICAST PROPERTIES

The goodness of the ALM system can be measured by some parameters, such as control overhead, robustness of the overlay, stress, and stretch.
**Control overhead** means the ratio of the necessary control messages sent by the clients to each other and the amount of the data traffic on the ALM system. In other words, the control overhead is a metric to examine the scalability of the overlay to large groups. Each member on the overlay exchanges refresh messages with all its peers on the overlay. Those messages build the control overheads at different routers, different links, and different members of the multicast group. For efficient use of network resources, the control overhead should be low (Banerjee et al., 2002).

**Robustness of the overlay** of the ALM protocols is measured by quantifying the extent of the disruption in data delivery when various members fail, and the time it takes for the protocol to restore delivery to the other members. Since hosts are potentially less stable than routers, it is important for ALM protocols to decrease the effect of host failures.

**Stress** means the number of identical packets sent through a certain link. In the case of IP-Multicast, the stress is one, but in the case of the ALM, the stress is possibly higher than one. The total amount of the necessary network resources of an ALM system can be measured by the following expression:

\[
\sum_{i=1}^{L} d_i \cdot s_i,
\]

where \( L \) is the number of active links in the data or control message transmission, \( d_i \) is the propagation delay of the link, and \( s_i \) is the stress of the link.

**Stretch** measures the ratio of length of the path from the sender to the receiver in the case of the ALM, and path length in the case of the pure unicast transmission for each receiver. For the IP-Multicast, the stretch equals to one for every receiver; for ALM the stretch is possibly higher than one.

**AD-HOC MULTICAST**

There are a lot of various protocols and implementations of the Application Layer Multicasting. Not only that, the communication over the wireless networks also enhances the importance of the ALM. The reason is because that in the case of mobile devices, the importance of the *ad-hoc networks* is increasing. Ad-hoc is a network that does not need any infrastructure. Such networks are the *Bluetooth* (Haartsen, 1998) and *mobile ad hoc network* (MANET), which comprise a set of wireless devices that can move around freely and communicate in relaying packets on behalf of one another (Mohapatra et al., 2004).

In computer networking there is a weaker definition of this ad-hoc network. It says that ad-hoc is a computer network that does not need routing infrastructure. It means that the mobile devices that use base stations can create an *ad-hoc computer network*. In such situations, instead of *IP-multicast*, the usage of application-level networking (ALN) technology is more practical. In order to support this group communication, various multicast routing protocols are developed for the mobile environment. The multicast routing protocols for ad-hoc networks differ in terms of state maintenance, route topology, and other attributes.

The simplest ad-hoc multicast routing methods are *flooding* and *tree-based routing*. *Flooding* is very simple, which offers the lowest control overhead at the expense of generating high data traffic. This situation is similar to the traditional IP-Multicast routing. However, in the wireless ad-hoc environment, the *tree-based routing* fundamentally differs from the situation in the wired IP-Multicast, where the tree-based multicast routing algorithms are obviously the most efficient ones, such as in the *multicast open shortest path first* (MOSPF) routing protocol (Moy, 1994). Though tree-based routing generates optimally small data traffic on the overlay in the wireless ad-hoc network, the tree maintenance and updates need a lot of control traffic. That is why the simplest methods are not scalable for large groups.

A more sophisticated ad-hoc multicast routing protocol is the *core-assisted mesh protocol* (CAMP), which belongs to the mesh-based multicast routing protocols (García-Luna-Aceves & Madruga, 1999). It uses a shared mesh to support multicast routing in a dynamic ad-hoc environment. This method uses cores to limit the control traffic needed to create multicast meshes. Unlike the core-based multicast routing protocol as the traditional protocol independent multicast-sparse mode (PIM-SM), multicast routing protocol (Deering et al., 1996), CAMP does not require that all traffic flow through the core nodes. CAMP uses a receiver-initiated method for routers to join a multicast group. If a node wishes to join to the group, it uses a standard
Related Content

A Unifying Framework Design for the Management of Autonomic Network Functions
[www.igi-global.com/chapter/a-unifying-framework-design-for-the-management-of-autonomic-network-functions/214427?camid=4v1a](www.igi-global.com/chapter/a-unifying-framework-design-for-the-management-of-autonomic-network-functions/214427?camid=4v1a)

Aspect-Oriented Programming and Aspect.NET as Security and Privacy Tool for Web and 3D Web Programming
[www.igi-global.com/chapter/aspect-oriented-programming-aspect-net/49524?camid=4v1a](www.igi-global.com/chapter/aspect-oriented-programming-aspect-net/49524?camid=4v1a)

A New Process Model for IoT-Based Software Engineering
[www.igi-global.com/chapter/a-new-process-model-for-iot-based-software-engineering/220758?camid=4v1a](www.igi-global.com/chapter/a-new-process-model-for-iot-based-software-engineering/220758?camid=4v1a)

Cloud-Based Predictive Intelligence and Its Security Model
[www.igi-global.com/chapter/cloud-based-predictive-intelligence-and-its-security-model/219121?camid=4v1a](www.igi-global.com/chapter/cloud-based-predictive-intelligence-and-its-security-model/219121?camid=4v1a)