Adopting Minimum Spanning Tree Algorithm for Application-Layer Reliable Mutlicast in Global Mutli-Gigabit Networks

Kirill Karpov[1], Dmitry Kachan[1], Nikolai Mareev[1], Veronika Kirova[1], Dmytro Syzov[1], Eduard Siemens[1] and Viatcheslav Shuvalov[2]

1Future Internet Lab Anhalt, Anhalt University of Applied Sciences, Bernburger Str. 57, 06366 Köthen, Germany
2Department of Transmission of Discrete Data and Metrology,
Siberian State University of Telecommunications and Information Sciences, Kirova Str. 86, Novosibirsk, Russia
{kirill.karpov, dmitry.kachan, nikolai.mareev, dmytro.syzov, veronika.kirova, eduard.siemens}@hs-anhalt.de,
shvp04@mail.ru

Keywords: Application Layer Multicast, Point-to-Multipoint, RMDT, Cascaded Data Transmission, Minimum Spanning Tree, DCMST, Networking, High Bandwidth.

Abstract: Data transmission over the Wide Area Networks (WAN) is a common practice in nowadays Internet, however, it has its limitations. One of them is that IP multicast data transmission rarely can be applied outside of Local Area Networks (LAN). Due to its vulnerability, multicast traffic is blocked by most Internet Service Providers' (ISP) edge equipment. To overcome this limitation, an Application Layer Multicast (ALM) is proposed, where multicast functionality is implemented on the end-hosts, instead of network equipment. For the application of ALM no changes in the network are needed, what significantly facilitate deployment of multicast services. The key point of this work is to implement ALM for reliable high-speed data transmission over WANs using RMDT transport protocol and Minimum Spanning Tree (MST) algorithm, which shall improve bandwidth utilization and provide a higher data rate for data propagation across multiple sites.

1 INTRODUCTION

Transmission of big data chunks over WANs to multiple sides can be implemented using point-to-point approach – when sender host simply initiates data transmission to several destinations in parallel or one by one in the queue. In the first case, data flows share the same link, at least on sender's last mile, and the TCP protocol doesn't share the network resources evenly [5]. Moreover, higher usage of shared bandwidth in that case means lower bandwidth per individual connection. Another approach will provide entire available bandwidth for the receivers, however, each of them will receive data only in its turn. Both solutions will cause unnecessary use of bandwidth since each data set will be sent separately to each receiver.

Usually, LAN connections and connections on short distances have higher bandwidth, fewer impairments and lower latency. In contrary, WAN connections have cross traffic between the data endpoints, many intermediate network devices which cause additional network impairments and a higher level of latency. Moreover, bandwidth in WAN connections has usually a higher price than in LANS.

Using an MST algorithm, it is possible to employ metrics, which will evaluate connections between involved hosts to create optimal ALM topology for data propagation, which will send data over LANs and short distance connections in parallel, and send data in an ad-hoc manner over WAN links. To get the benefit of multicast service, the ad-hoc connection will not completely receive the data before forwarding it further, instead, it will pass it to the next host alongside with confirmation of successful reception of each consequent data chunk. This allows usage of e.g. file-based video transmission, when all users may start to process the file without waiting till end of data transmission.

To make high-speed data transmissions over WAN possible, the RMDT [2] transport protocol was used, since it satisfies all necessary conditions described above:

- The protocol provides WAN acceleration service, which makes network impairments and latency up to 1 second nearly negligible.
- It can serve up to 10 receivers in parallel within a single session natively – means no fairness issues will be among receivers and available bandwidth will be shared evenly. Moreover, it has a centralized congestion control, which allows the coexistence with the cross traffic in IP WANs.
- RMDT is a pure user-space software library which makes it possible to create network applications capable of forwarding the received data chunks further to the next receiver.

The remainder of this paper is structured as follows: Section 2 reviews related work. Section 3 describes the developed application algorithm. Section 4 is devoted to the experimental setup, describes hardware and software equipment, testing environment, measurement and representation methods for the current research. The results of conducted experiments is presented in Section 5. Finally, conclusion discusses the results, followed by the future work.

2 RELATED WORK

A detailed survey of existing tree-based application layer multicast algorithms has been made by Computing Laboratory, University of Kent [6]. However, the efficiency of observed protocols have been investigated only in terms of tree cost and delay optimization.

S. Banerjee and B. Bhattacharjee [7] have also analyzed various application layer multicast algorithms and determine the fields of applicability for them. They substantiated that tree-first application layer multicast approach is useful for high-bandwidth data transfers, however it is less suited for real-time applications.

The Narada performance study [8] provides several useful performance metrics such as latency, bandwidth, stress, resource usage, etc.

The given paper describes and study performance of application layer multicast in combination with high-bandwidth data transport applications.

3 ALGORITHM DESCRIPTION

The key part of developed application layer multicast system is minimum spanning tree algorithm, which is supposed to construct an optimal tree, based on the chosen metrics. In the given research, RTT is the optimization metric. It has been chosen, because RTT is one of the basic characteristics of a network, which is easy to obtain, unlike the available bandwidth, which might cause undesirable effects to the network operation while being measured.

The given ALM realization uses tree-first approach, therefore the first step of application workflow is to recognize the network environment among all hosts which are involved in transmission process. With the chosen metrics the protocol forms an adjacency matrix. This matrix represents a complete directed weighted graph, where the weights are the values of the chosen metric e.g. RTT, available bandwidth, air distance, etc. The result of MST operation will be an adjacency matrix with zeroed non-optimal paths which represents the optimal spanning tree without loops.

The given tree is the directive map for multipoint transmission application, in this case – Data Clone a point-to-multipoint data copy application based on RMDT.

4 EXPERIMENTAL SETUP

4.1 Testing Environment

As an experimental environment, Amazon AWS has been chosen. It provides virtual infrastructure in selected continents and regions. Cascade network transmission infrastructure based on c5.xlarge virtual instance with an Ubuntu 18.04 operating system, 4 vCPU, 8 Gb RAM, and up to 10 Gbps available network access bandwidth have been choosen.

In order to minimize disk I/O operations overhead of getting data from the disk storage, a RAM disk as data storage has been configured on each host.

The instances are distributed all over the world in the following AWS regions: US West (Oregon), EU (Frankfurt), EU (London), Asia Pacific (Singapore), Canada (Central). Using a geo IP service, it has been found out that the hosts from Canada (Central) region are located in Montreal, and the US West (Oregon) data center is located in Boardman. In each

region 3 c5.xlarge virtual instances have been deployed. The regions have been chosen to get different variations of network conditions, such as long and short distances, international and intercontinental links. The air distances between AWS data center locations are shown in Table 1. The minimum spanning tree obtained based on the air distances between AWS regions is shown on Figure 1.

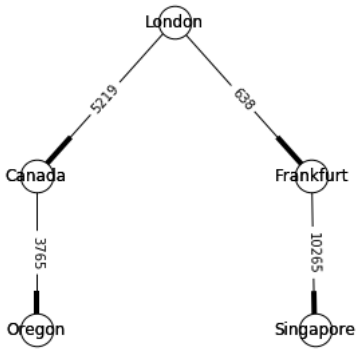

Figure 1: The tree generated by minimum spanning tree algorithm based on the air distance between AWS instance locations.

Table 1: Air distance between virtual instances locations in kilometers.

AWS regions	Oregon	Frank-furt	Lon-don	Singa-pore	Cana-da
Oregon	0	8393	7906	13094	3765
Frankfurt	8393	0	638	10265	5842
London	7906	638	0	10854	5219
Singapore	13094	10265	10854	0	14803
Canada	3765	5842	5219	14803	0

4.2 Software Equipment

For the experiments, the following software and technologies have been used.

1) **Dataclone** – RMDT-based software, which provides point to multipoint data transport functionality [2]. It uses BQL congestion control [3] which is tolerant to big delays and dramatic packet loss rates. In the experiments it will allocate 100 MB of RAM for both send and receive buffers.
2) **Multipoint sender** – a TCP-based application, developed by us to implement point-to-multipoint data transport capability and cascading functionality as Dataclone is doing. It has been created as TCP reference in multipoint field. It uses different threads for simultaneous multi-destination transmission and barrier type of synchronization.

5 EXPERIMENTAL RESULTS

As mentioned in Section 3, the first step of tree-first application layer multicast is the investigation of the given network environment. The result of RTT measurements between the deployed AWS regions is shown in Table 2.

Table 2: Packet RTT delays between virtual instances, in milliseconds.

AWS regions	Ore-gon	Frank-furt	Lon-don	Singa-pore	Canada
Oregon	0	100	141	162	65
Frankfurt	100	0	13	174	100
London	141	13	0	173	87
Singapore	162	174	173	0	219
Canada	65	100	87	219	0

As can be seen from the tables, the RTT values between regions are corresponding to distance metrics, however, the dependency between delay and distance is not linear due to the physical network paths [9] and other factors, such as cross-traffic, configurations and types of intermediate devices, their number, etc.

Based on obtained metrics, the minimum spanning tree for the given set of hosts has been constructed. It is shown on Figure 2.

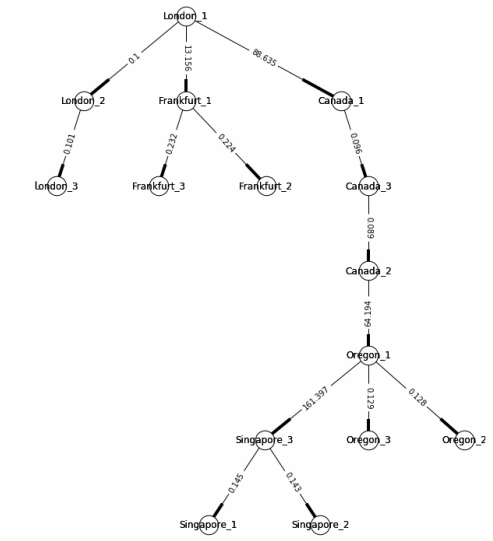

Figure 2: The tree generated by minimum spanning tree algorithm in the AWS network cloud infrastructure using RTT as weights (in milliseconds).

The multipoint TCP realization, which has been described in Section 4.2 produced a constant data rate during all transmission time, which was about 76 Mbps for each edge of the tree.

With Dataclone application as a carrier, the data rates are distributed less uniformly across the transmission tree, in comparison with the TCP multipoint experiment, as shown on Figure 3.

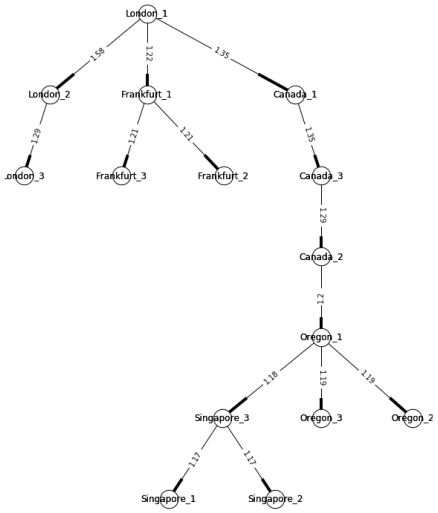

Figure 3: The data transmission tree with average data rates with edges weighted in Gbps.

During the test that lasted 145 seconds, 20 GB of data has been transmitted. The average data rate for the whole tree was 1.14 Gbps, which is 15 times higher than during TCP multipoint experiment. Hereby the average data rate was calculated as the size of the transmitted data chung (20 GB) divided by runtime of the root sender at London_1 and so is slighthly less than the lowest data rate at the graph on figure 3 London_1 is the node with the highest outbound traffic of 4.15 Gbps in total. The average link bandwidth across the edges was 1.28 Gbps.

More detailed result of the experiment is shown on Figure 4. Data sets with rates were processed with Savitzky-Golay Filter to get rid from the outliers.

The plot shows that, starting from 20 seconds, data rates on all long links after London region were stabilized near 1.2 Gbps. This value can be used for stable multipoint streaming.

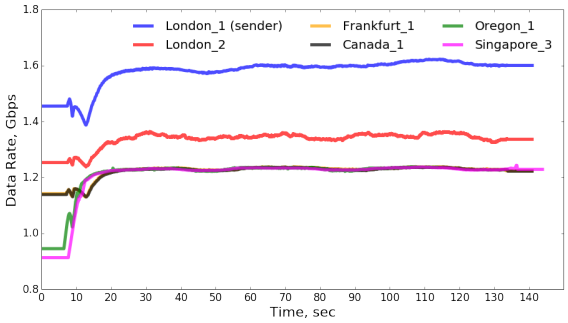

Figure 4: Data rates from the nodes of the tree.

6 CONCLUSIONS

Several conclusions can be drawn from the conducted experiment:

1) The alternative way for data transmission, e.g., via flat, non-hierarchical connection topology requires about 16 Gbps outbound link bandwidth for sender, with 100% network resource utilization. The experimental results show that the hierarchical scenario achieves the same performance with only 5 Gbps of link bandwidth at the maximum loaded sender.
2) RMDT with its BQL congestion control is able to serve in both flat (pure point-to-multipoint) as well as in hierarchical (tree-based) connection topologies, and achieves much higher bandwidth per link utilization in the latter case.
3) Despite the fact that ISP providers have fat pipes, e.g 10 Gbps, it is not always possible to fit into such limit due to variety of obstacles, such as server hardware or virtual configurations limits. However, with WAN acceleration it is possible to overcome most of these limits.
4) RMDT with BQL congestion control shows 15 times higher data rate, than a comparable TCP-based multipoint data transmission realization.

There are a lot of ways to further improve the ALM approach for data transmission. The future steps towards the improvement of the current approach might be the following:

1) To change MST algorithm, which currently does not consider maximum output number limitations of the sender. The alternative to MST could be degree-constrained minimum spanning tree (DCMST) algorithm [10].
2) Using additional metrics for tree nodes. MST algorithm does not consider the performance of

the node and decides to split or to make a chain of nodes only on edges metrics. Thus, a question remains open: what kind of local topology gives the best performance? There is room for investigation of that question in the future.

3) Another weak point of RTT-based MST is that it does not take into account the lower layer infrastructure. Considering this circumstance, it makes sense to build a transition tree, based on available L4 infrastructure.
4) The public network is always changing due to variety of factors, such as cross traffic from other customers, their activity in specific time and date, and so on. Tracking the history of such events and analyzing their effect to network conditions might be helpful for WAN acceleration applications.

ACKNOWLEDGMENTS

This work has been funded by Volkswagen Foundation for trilateral partnership between scholars and scientists from Ukraine, Russia and Germany within the CloudBDT project: “Algorithms and Methods for Big Data Transport in Cloud Environments”.

REFERENCES

[1] V. Kirova, E. Siemens, D. Kachan, O. Vasylenko, and K. Karpov, “Optimization of Probe Train Size for Available Bandwidth Estimation in High-speed Networks,” in MATEC Web of Conferences, vol. 208, p. 02001, 2018.

[2] A. V. Bakharev, E. Siemens, and V. P. Shuvalov, “Analysis of performance issues in point-to-multipoint data transport for big data,” in 2014 12th International Conference on Actual Problems of Electronics Instrument Engineering (APEIE), 2014, pp. 431–441.

[3] N. Mareev, D. Kachan, K. Karpov, D. Syzov, E. Siemens, and Y. Babich, “Efficiency of a PID-based Congestion Control for High-speed IP-networks,” in Titel: Proceedings of the 6th International Conference on Applied Innovations in IT, 2018.

[4] M. Hock, R. Bless, and M. Zitterbart, “Experimental evaluation of BBR congestion control,” in 2017 IEEE 25th International Conference on Network Protocols (ICNP), 2017, pp. 1-10.

[5] R. L. Graham and P. Hell, “On the history of the minimum spanning tree problem,” Annals of the History of Computing, vol. 7, no. 1, pp. 43-57, 1985.

[6] S. Tan, G. Waters, and J. Crawford, “A survey and performance evaluation of scalable tree-based application layer multicast protocols,” 2003.

[7] S. Banerjee and B. Bhattacharjee, “A comparative study of application layer multicast protocols,” Network, vol. 4, no. 3, 2002.

[8] Y. Chu, S. G. Rao, S. Seshan, and H. Zhang, “A case for end system multicast,” IEEE Journal on Selected Areas in Communications, vol. 20, no. 8, pp. 1456-1471, Oct, 2002.

[9] J.-M. Beaufils, “How Do Submarine Networks Web the World?,” Optical Fiber Technology, vol. 6, no. 1, pp. 15-32, Jan, 2000.

[10] S. C. Narula and C. A. Ho, “Degree-constrained minimum spanning tree,” Computers & Operations Research, vol. 7, no. 4, pp. 239-249, 1980.