A BENCHMARKING SYSTEM FOR MULTIPATH OVERLAY MULTIMEDIA STREAMING

Svetlana Boudko, Wolfgang Leister

Norwegian Computing Center, Norway

Carsten Griwodz, Pål Halvorsen

Simula Research Laboratory, Norway
Department of Informatics, Univ. Oslo, Norway

ABSTRACT

The rapid growth of the Internet multimedia services brings new challenges to how multimedia streams can be delivered to the users over bandwidth-constraint networks. Different strategies that exploit multipath streaming in order to provide better utilization of the Internet resources have been proposed by the research community. However, there exists no metric that allows us to evaluate how close these strategies are to the optimal resource utilization. This paper proposes a static benchmarking system that models the best possible distribution of streams along multiple paths in an overlay network that is shared by several senders and receivers. We have tested it with several different network topologies, and present the test results in this paper.

Index Terms— multimedia streaming, resource allocation, overlay networks, adaptation, multipath routing

1. INTRODUCTION

The rapid growth of the various multimedia streaming services provided on the Internet in combination with the significant increase of the Internet population brings new challenges to how multimedia content can be delivered to the end users.

Although bandwidth in the backbone is generally considered sufficient today, several challenges for high quality multimedia streams still persist. Problems that can seriously affect multimedia streaming can be listed as follows: access networks are still not provisioned enough; slow convergence of routing protocols after link or node failures; BGP and OSPF, although designed to support resource efficient streaming, are still not so used.

Multimedia streaming applications are bandwidth consuming and delay sensitive. As the routing in the Internet is determined by policies of the routers and cannot be controlled by the end hosts, there is no guarantee that the path selected to deliver multimedia streams is provided with sufficient available bandwidth. At the same time, an alternative path or several alternative paths can be found in the network, which utilization gives higher throughput and can significantly improve the quality of the delivered stream. Constructing several paths for data forwarding can then be implemented in application level overlay networks.

Recently several different strategies [1, 2] have been proposed that intend to exploit multipath delivery for multimedia streaming by distributing streams between different paths. However, one cannot evaluate how good these strategies are in terms of optimal resource utilization when several multimedia senders and receivers share the same delivery infrastructure and compete for the same network resources. Therefore, we propose a baseline system that provides us with the best possible distribution of the streams over available delivery paths given that the complete knowledge of the network including its topology and resource availability is obtainable. The baseline system can then be used to quantify the difference between the optimal solution and solutions provided by algorithms that operate in dynamically changing networking environments with partial knowledge of the network.

The structure of the paper is organized as follows. Section 2 gives an overview of related work recently done by the research community. The model for the baseline system is formulated in Section 3. We present the results of stream distribution among multiple paths for selected network topologies in Section 4. Our conclusion and discussion of our future work is given in Section 5.

2. MULTI-PATH STREAMING

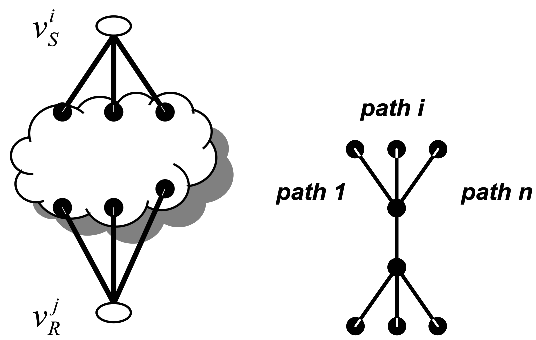
Multipath streaming is one of the approaches that have been proposed in several papers [3–5] in order to overcome packet loss and bandwidth limitations in the Internet during the delivery of multimedia streams.

There are mainly two ways of implementing multipath routing. It can be done either in the network layer exploiting for example traffic engineering techniques [6] or in the application layer using overlay networks. When implemented in the network layer several routing protocols and algorithms can be used including OSPF Optimized Multipath protocol [7], Multipath Distance Vector Algorithm [8], Quality of Service (QoS) routing mechanisms [9], etc. However, these algorithms have serious drawbacks. They require routers to maintain detailed information about all paths between the router and every possible destination and therefore scale poorly. Implementing multipathing at the network layer also implies a certain cooperation between ISPs including that all ISPs choose to use the same routing protocol. In addition, ISPs are unlikely to allow others to control the traffic going through their networks.

Implementing multipath streaming in the application layer by using overlay networks helps to overcome the problems in deployment related to ISPs. However, it is certainly less efficient in terms of latency. CollectCast [10] is an overlay service that provides forwarding of streams from multiple senders to one receiver. Operating at the application level this service is also able to detect and exploit underlying network topology and network performance characteristics. In SplitStream [11] multiple trees are built and used for streaming in order to balance the forwarding load. The content is split into several stripes, which are then multicast using separate trees.

Using application layer overlays for multipath streaming is also beneficial since not all data in the multimedia stream is equally important in terms of user-perceived quality. Therefore, depending on conditions observed along the overlay links an overlay node can perform certain processing of the forwarded streams [10] including applying error correction techniques, selective drop of packets, or forwarding more important packets along more reliable paths.

The effectiveness of using overlay networks for multipath streaming depends on their ability to (1) detect the current net-



(a) Total throughput between one receiver and one sender.

(b) Shared link.

Fig. 1. Link and path constraints.

work conditions such as latency and available bandwidth and (2) avoid using points of shared congestion. Available network resources can be detected using network tomography and inference techniques [12].

Research in defining correlation techniques for detecting shared congestion has been done by the authors [13]. However, in spite of the considerable amount of work that has been done in the area, we have so far found only examples that demonstrate improved performance over other ad-hoc mechanisms or over unicast performance. We have therefore decided to create the benchmark for optimal performance presented in the next section.

3. BENCHMARK

To build our benchmarking system, we model the overlay network that includes senders, receivers and overlay nodes as a fully meshed graph $D^o = (V^o, A^o)$, where V^o is the set of vertices that represents the nodes of the network, and A^o is the set of arcs that represent the overlay links between the nodes. Sets $V_s^o \in V^o$, $V_o^o \in V^o$ and $V_r^o \in V^o$ are disjoint subsets of vertices representing respectively sender nodes, overlay nodes and receiver nodes.

In the overlay graph, we define a set of all possible overlay paths between the receivers and the senders. Note that these paths should not contain loops. For each overlay path, we build a corresponding path in the underlying Internet connecting the vertices of the overlay path by the shortest paths in the underlying network. Thus, we define a set of all possible paths P in the underlay. Then, $p_{i,j}^k$ denotes the k-th path of the subset which contains the paths connecting the sender i with the receiver j. $K_{i,j}$ denotes the number of paths from the sender i with the receiver j.

The intersection of all paths P forms the underlay graph $D = (V, A)$, where V is the set of vertices representing the nodes of the underlying network and A is the set of arcs representing the underlay links. For each path p and each ark a in the underlay graph we define a function δ as follows:

$$\delta(a, p) = \begin{cases} 1, \text{if} & a \in p \\ 0, \text{if} & a \notin p \end{cases} \tag{1}$$

To model network resources we define two functions on the underlay arcs. One is the bandwidth function that expresses the available bandwidth of the arc a:

$$b : A \rightarrow R_0^+ \tag{2}$$

The second one is the latency function that defines the latency of the arc a:

$$l : A \rightarrow R_0^+ \tag{3}$$

The same functions are defined on the paths. The available bandwidth of the path p is defined as the lowest bandwidth among all arcs that belong to the path p:

$$b(p) = \min\{b(a)\}, a \in p \tag{4}$$

The latency of the path p is the sum of all latencies of the arcs that belong to the path p:

$$l(p) = \sum_{a \in p} l(a) \tag{5}$$

Streaming requirements of the requested streams are defined by the matrix R, where the matrix element $r_{i,j}$ represents the required bitrate at which the stream from the sender v_s^i is streamed to the receiver v_r^j. In addition, we define the matrix R^b which is similar to the above matrix r, but contains the bitrates for the basic layers. The basic layer represents the lowest acceptable perceived quality of the multimedia content.

One more matrix used to specify the receivers' requirements for stream delivery is matrix D. Its elements show the acceptable delay for stream arrival to the receivers. The values for these elements can differ significantly depending on the nature of the streaming content. Live streaming is more delay sensitive than on-demand streaming or video downloading.

The stream distribution problem is specified as the following. The streams requested by the receivers have to be distributed among the paths in a way that the overall throughput of the system is maximized while the delivery of the basic layer is guaranteed. This problem can be formulated as a linear programming problem, and to solve it we apply the Simplex method [14].

We introduce the variable $x_{i,j}^k$ that denotes a share of the multimedia stream sent from the content provider v_S^i to the receiver v_R^j through the path k. To find the best possible distribution of the sending streams among the available paths, we maximize the following objective function:

$$\max \sum_{p_{i,j}^k \in P} x_{i,j}^k \cdot r_{i,j} \tag{6}$$

The objective function is subject to the set of constraints given below.

As not all data in the multimedia stream is equally important for the perceived quality, we may drop less important data though the delivery of the basic layer should be guaranteed for all streams:

$$\forall\{i,j\} : \sum_{k=0,K_{i,j}} x_{i,j}^k \cdot r_{i,j} \geq r_{i,j}^b \tag{7}$$

The next constraint is illustrated in Fig. 1(a). It implies that the sum of sending rates along all paths from one sender to one receiver should not exceed the bitrate assigned to this stream.

$$\forall\{i,j\} : \sum_{k=0,K_{i,j}} x_{i,j}^k \leq 1 \tag{8}$$

As depicted in Fig. 1(b) the links in the underlay can be shared between several delivery paths. We need to specify that the total sending rate does not exceed the available bandwidth of the shared link.

$$\forall\{a\} : \sum_{p_{i,j}^k \in P} x_{i,j}^k \cdot r_{i,j} \cdot \delta(a, p_{i,j}^k) \leq b(a) \tag{9}$$

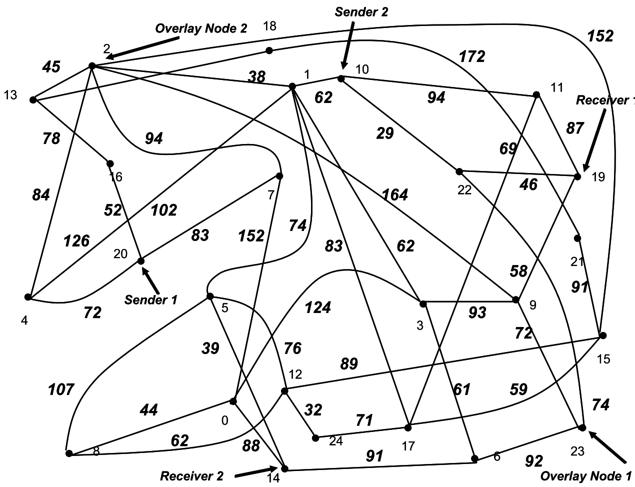


Fig. 2. Example for Waxman topology.

Finally, the latency of the paths participating in delivering a particular stream should not exceed the acceptable delay assigned to this stream.

$$l(p_{i,j}^k) \leq d_{i,j} \tag{10}$$

The objective function and the set of constraints described in this section define our benchmarking system. This system provides a reference for dynamic algorithms since it selects paths in a multi-source, multi-destination streaming scenario in such a way that the global resource usage is optimal. Although the system is not practically applicable because of the algorithm's execution time and the amount of information, which is needed to be collected in real time in order to execute the algorithm, it provides a means of comparing the performance of other algorithms with respect to the optimal choice of paths. We see it as an alternative to simply comparing with another ad-hoc strategy.

In the following section, we demonstrate the results of the benchmark for two example scenarios.

4. USAGE EXAMPLE

Our benchmark computes the optimal bandwidth assigment for a given static network condition and application situation. It is not a topic of this paper to compare any existing multipath streaming technique with this benchmark, but we consider it necessary to demonstrate that the optimal multipath streaming decision does not have a trivial solution in the general case. We have done this by applying the benchmark computation to several topologies and overlay node placements. Here, we show only two examples.

One of the examples is that of two senders, two receiver and two overlay nodes that were randomly placed in a Waxman topology, a typically considered a fairly realistic generation strategy for networks. The Waxman topology depicted in Fig. 2 was generated using the Brite topology generator [15]. In the second example, we placed the same number of nodes randomly in a chessboard topology of edge length 6. This topology, the vertex labels and the link bandwidth resources are depicted in Fig. 3.

We applied the benchmark to these two situations, where both senders sent one stream to each receiver. For comparison, we computed also the through achievable in a unicast approach that used a direct shortest path route between each pair of sender and receiver.

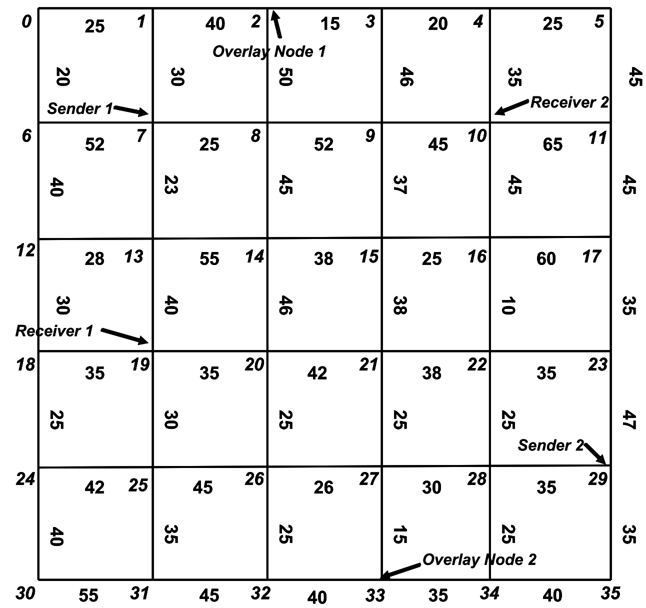


Fig. 3. Example for Chessboard topology.

Shortest path from	sending rate distribution for Waxman to v_r^1	to v_r^2	sending rate distribution for Chessboard to v_r^1	to v_r^2
v_s^1	42.5	42.5	23.0	25.0
v_s^2	87.0	39.0	25.0	35.0

Table 1. Total sending rate using shortest path routing.

The throughput achieved in the shortest-path unicast case is then reported in Table 1, the throughput achieved using the benchmark's optimization is shown in Table 2. The latter table shows also how much of the throughput is due to each of the individual paths between a sender and a receiver. These individual path use optionally one or more of the overlay nodes and contribute to the overall throughput achieved between this pair, while cooperating with the other sender-receiver pairs in the consumption of the underlay links' resources to achieve the optimal overall throughput.

We see cleary that the benchmark achieves a much higher per-pair throughput than the shortest path approach, but this is to be expected for non-trivial topologies. However more importantly, when the benchmark assigns the bandwidth shares of the individual overlay paths that are available to each sender-receiver pair, it has to do this in a non-trivial way to achieve the optimum throughput. We can even see that considerable resources are gained from using non-trivial paths. In case of the chessboard configuration, we can even see that a non-trivial path ($v_s^2 \rightarrow v_o^2 \rightarrow v_r^1$) contributes the largest share to the overall throughput for one pair.

We are therefore confident that meeting the limits of achievable bandwidth is a challenge for ad-hoc multipath streaming decision mechanisms. In creating our benchmark, we have gained a means of evaluating the performance of dynamic mechanisms for their potential for further improvement.

5. CONCLUSIONS AND FUTURE WORK

We have presented a benchmarking system for multipath media streaming of scalable media which we have applied to different

Overlay path		sending rate distribution for Waxman		sending rate distribution for Chessboard	
from	through	to v_r^1	to v_r^2	to v_r^1	to v_r^2
v_s^1		32.0	31.0	19.0	11.0
v_s^1	v_o^1	10.1	10.8	7.5	1.6
v_s^1	v_o^2	9.8	10.1	2.2	10.0
v_s^1	v_o^1, v_o^2	1.0	7.3	2.0	1.8
v_s^1	v_o^2, v_o^1	9.1	5.7	1.0	2.3
sum		62.0	64.9	31.7	26.7
v_s^2		87.0	39.0	12.0	34.0
v_s^2	v_o^1	11.0	8.0	6.0	1.0
v_s^2	v_o^2	0.5	12.0	26.0	4.0
v_s^2	v_o^1, v_o^2	6.0	3.0	0.7	1.1
v_s^2	v_o^2, v_o^1	5.6	6.0	1.1	0.7
sum		110.1	68.0	45.8	40.8

Table 2. Distribution of sending rates along overlay paths.

network topologies. For a given topology and set of overlay nodes it determines, for a given streaming demand, the optimal allocation of link bandwidth with the goal of allocating the maximum bandwidth to each of the streams. As bandwidth demands for streams may vary widely, alternative optimization goals could be achieved by modifying equation 1, in order to maximize the bandwidth assigned to a stream relative to its minimal bandwidth.

Since the calculation is rather computing-intensive, and since the knowledge of the entire system state is necessary for the calculation this method is not suited for an implementation on a network node due to scalability reasons. However, our benchmarking system will be used to compare other algorithms with the optimal case provided by this work.

We plan to implement multipath streaming solutions from the literature and use the results of the benchmark to evaluate how closely these overlay solutions can track the optimal resource utilization decision in situations where the network conditions are changing dynamically.

As a further step, we intend to develop distributed algorithms that operate under partial knowledge of the network topology and dynamically changing networking parameters such as available bandwidth and packet loss. These algorithms are then to be integrated into the overlay node together with other basic streaming functionalities. These can include stream caching, transcoding of multimedia content, error correction mechanisms, etc., and will require considerable extension of the benchmark as well.

Another important issue is the interaction between the overlay network and wireless access networks especially in connection with handovers between two access networks [16]. It is important to consider how signaling between the mobile device and the overlay net is handled. Signaling should be done so that both the terminal mobility and the session mobility are provided for the users.

6. ACKNOWLEDGMENTS

The work presented in this paper has been conducted as a part of the Adimus (Adaptive Internet Multimedia Streaming) project funded by the Nordunet-3 program.

7. REFERENCES

[1] Dan Jurca and Pascal Frossard, “Distributed media rate allocation in overlay networks,” in *2006 IEEE International Conference on Multimedia and Expo*, July, 2006, vol. 9-12, pp. 1401 – 1404.

[2] Dan Jurca and Pascal Frossard, “Video packet selection and scheduling for multipath streaming,” *IEEE Transactions on Multimedia*, vol. 9, no. 3, pp. 629–641, April 2007.

[3] L. Golubchik, J. Lui, T. Tung, A. Chow, W. Lee, G. Franceschinis, and C. Anglano, “Multi-path continuous media streaming: What are the benefits,” 2002.

[4] John G. Apostolopoulos and Mitchell D. Trott, “Path diversity for enhanced media streaming,” *Communications Magazine, IEEE*, vol. 42, pp. 80– 87, 2004.

[5] H. Chu and K. Nahrstedt, “Dynamic multi-path communication for video traffic,” in *Proceedings of the Thirtieth Hawaii International Conference on System Sciences*, 1997, vol. 1, pp. 695–704.

[6] Attila Takács, András Császár, Róbert Szabó, and Tamás Henk, “Generic multipath routing concept for dynamic traffic engineering,” in *Communications Letters, IEEE*, 2006, vol. 10, pp. 126 – 128.

[7] C. Villamizar, “Ospf optimized multipath (ospf-omp),” in *Proceedings of the forty-fourth Internet Engineering Task Force*, 1999, pp. 45–54.

[8] P. Narvaez, K.-Y. Siu, and H.-Y Tzeng, “Efficient algorithms for multi-path link-state routing,” .

[9] R. Guerin, A. Orda, and D. Williams, “Qos routing mechanisms and ospf extensions,” 1996.

[10] Mohamed Hefeeda, Ahsan Habib, Boyan Botev, Dongyan Xu, and Bharat Bhargava, “Promise: peer-to-peer media streaming using collectcast,” in *MULTIMEDIA ’03: Proceedings of the eleventh ACM international conference on Multimedia*, 2003, pp. 45–54.

[11] M. Castro, P. Druschel, A. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, “Splitstream: High-bandwidth multicast in cooperative environments,” 2003.

[12] Tian Bu, Nick Duffield, Francesco Lo Presti, and Don Towsley, “Network tomography on general topologies,” in *SIGMETRICS ’02: Proceedings of the 2002 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, 2002, pp. 21–30.

[13] D. Rubenstein, J.F. Kurose, and D.F. Towsley, “Detecting shared congestion of flows via end-to-end measurement,” *IEEE/ACM Transactions on Networking*, vol. 10, no. 3, pp. 381–395, June 2002.

[14] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery, *Numerical Recipes 3rd Edition The Art of Scientific Computing*, Cambridge University Press, 2007, ISBN 13:9780521880688.

[15] A. Medina, A.Lakhina, I. Matta, and J. Byers, “Brite: An approach to universal topology generation,” in *MASCOTS ’01: Proceedings of the International Workshop on Modeling,Analysis and Simulation of Computer and Telecomunications Systems*, 2001.

[16] Tiia Sutinen and Tapio Frantti, “Reference service architecture for multimedia services in heterogeneous multi-access networks,” in *Mobilware’08: Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications*, 2008.