Multiple streaming at the network edge

Alhaisoni, M.; Ghanbari, M.; Liotta, A.

Published in:
Proceedings First International Conference on Advances in P2P Systems, AP2PS'09, Sliema, Malta, October 11-16, 2009

DOI:
10.1109/AP2PS.2009.31

Published: 01/01/2009

Document Version
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the author's version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain.
• You may freely distribute the URL identifying the publication in the public portal.

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
Multiple Streaming at the Network Edge

Majed Alhaisoni, Mohammed Ghanbari
School of Computer Science and Electronic Engineering
University of Essex
Colchester, UK
malhai@essex.ac.uk
ghan@essex.ac.uk

Antonio Liotta
Electrical Engineering Department
& Mathematics and Computer Science Department
Technische Universiteit Eindhoven
The Netherlands
a.liotta@tue.nl

Abstract— Streaming video over the Internet, including cellular networks, has now become commonplace. Network operators typically use multicasting or variations of multiple unicasting to deliver streams to the user terminal in a controlled fashion. An emerging alternative is P2P streaming, which is theoretically more scalable but suffers from other issues arising from the dynamic nature of the system. User’s terminals become streaming nodes but these are not constantly connected. Another issue is that they are based on logical overlays, which are not optimized for the physical underlay infrastructure. An important proposition is that of finding effective ways to increase the resilience of the overlay whilst at the same time not conflicting with the network. In this article we look at the combination of two techniques, multi-streaming (redundancy) and locality (network efficiency) in the context of both live and video-on-demand streaming. We introduce a new technique and assess it via a comparative, simulation-based study. We find that redundancy affects network utilization only marginally if traffic is kept at the edges via localization techniques.

Keywords- P2P; Multimedia; Redundancy; Locality-awareness

I. INTRODUCTION

Overlays and P2P (peer-to-peer) systems, originally developed as an alternative to IP multicast and to provide file sharing services, have moved beyond that functionality. These technologies have deeply improved the distribution of information on the Internet by enabling resourceful cooperation among end consumers. With the growing bandwidth capacity provided by the Internet, they are also proving to be key technologies for the delivery of real-time video and of video-on-demand content.

By the cooperation among peers in helping each other in the network, P2P technology overcomes various limitations of the more conventional client-server paradigm to attain user and bandwidth scalabilities. In a P2P streaming application, multimedia contents are delivered to a large group of distributed users with low delay, high quality and high robustness [1]. P2P-based versions of IPTV, Video on Demand (VoD), and conferencing are thus becoming popular.

Many hosts can be supported by a P2P multimedia system, possibly in excess of hundreds or even millions, with miscellaneous heterogeneity in bandwidth, capability, storage, network, and mobility. Another aim is to maintain the stream even under dynamic user churn, frequent host failures, unpredictable user behaviors, network traffic and congestion. To accomplish these goals, it is imperative to address various challenges to achieve effective content delivery mechanisms, including routing and transport support.

In this article, we are primarily concerned with finding effective ways to increase the resilience and scalability of the overlay whilst at the same time minimizing the impact on the physical network (or underlay). We find that P2P frameworks mostly fail to address the latter issue and, in doing so, they tend to cause severe network operational and management issues. In turn, this limits P2P scalability when traffic streams traverse, and thus congest, large portions of the network. Another issue is that existing P2P streaming systems are intrinsically best-effort. This fact, combined with their network unfriendly behavior, often leads the operator to impair P2P traffic, with detrimental consequences for the resulting quality of service.

The key question we are addressing herein is whether and how it would be possible to increase the user quality of experience (QoE) in P2P streaming. A common technique is to increase redundancy, i.e. send multiple streams to the same user in order to reduce packet loss. The downside is that redundancy increases traffic, thus reducing network utilization and, hence, increasing congestion.

In order to retain the benefits of redundancy (QoE) and reduce its detrimental effects on the network, we study the combination of two techniques, multi-streaming and network locality. We find that by keeping traffic local among the peers and mainly at the edges of the network, the benefits of multi-streaming outweigh their shortcomings. We carry out a comparative evaluation using a popular P2P TV system, Joost, as benchmarking. Initial results indicate that QoE is significantly improved at little cost for the network.

II. RELATED WORK

As we are concerned with network QoS, P2P locality awareness, and stream redundancy in this paper, we give an
overview of different studies that have looked at these topics individually.

Ways to pursue efficiency between overlay and underlay have started to be investigated only recently. Authors in [2] propose a technique, where the peers on the overlay are chosen based on their mutual physical proximity, in order to keep traffic as localized as possible. A similar approach is described in [3], where they measure the latency distance between the nodes and appropriate Internet servers called landmarks. A rough estimation of awareness among nodes is obtained to cluster them altogether, as in [4] [5].

On the other hand, another study in [6] proposes different techniques where the video stream is divided into different flows that are transmitted separately to increase parallelism and, hence, reduce transmission latency. The authors use the PSQA technique that gives estimates of the quality perceived by the user. This study was concerned on how to impact and improve on quality (as measured by PSQA). They introduce three cases: sending a single stream between nodes; sending two duplicate streams via different paths; and sending two disjoint sub-streams whose union recreates the original one. In our work we look at the case of multiple redundant streams, looking at the effects that redundant streams have on both the network and the user QoE. Also we emphasize on techniques to choose inter-communicating peers based on their mutual proximity, to keep traffic local and minimize the impact on the network.

Overlay locality is studied also by [7], where the authors make use of network-layer information (e.g. low latency, low number of hops and high bandwidth). We use though a different distance metric, based on RTT (round trip time) estimations, to prioritize overlay transmissions. Additionally, we use a cluster management algorithm whereby intercommunicating peers are forced to periodically handover, in order to pursue computational as well as network efficiency (as explained in [13] and [14]).

Hefeeda et al [8] have proposed a mechanism for P2P media streaming using Collectcast. Their work was based on downloading from different peers. They compare topology-aware and end-to-end selection based approaches.

The latter approach is also the subject of [9], which employs a simpler version of our RTT approach based on continuous pinging of peers. Similarly, we adopt clustering to limit the signaling overheads associated with this process and prevent bottlenecks.

Other studies such as [10], propose relevant methods to serve multiple clients based on utility functions or clustering. A dynamic overlay capable of operating over large physical networks is presented in [11]. In particular, they show how to maximize the throughput in divisible load applications.

Looking at previous studies, we can say that our main contributions are:
1) To study a new combination of existing techniques (cross-layer optimization, localization, forced handovers, and multi-streaming)
2) To take the perspective of the network operator, in trying to harmonize overlay and underlay networks
3) To look for trade-offs between redundancy levels (to increase QoE) and network efficiency.

III. PROPOSED APPROACH

A. Target Architecture

In our study we increase the number of redundant (multi) streams from 1 to 5 as shown in Figure 1. Sources {S1…S5} are chosen based on locality and are also continuously (periodically) forced to handover, choosing new sources from a pot of available sources. These are prioritized based on mutual inter-peer distances to ensure that traffic is kept as local as possible. On the other hand, forced handovers ensure that the important feature of computational load-balancing is maintained – we have discussed this particular aspect in previous publications [13] [14].

![Figure 1 - Proposed architecture](image)

Instead, herein we are mainly interested in understanding whether location-aware P2P techniques can actually reduce the detrimental effects of P2P redundant streaming. Under architectures other than P2P (unicast, multiple unicast and multicast), redundant streams increase QoE but have the side effect of increasing network congestion. An interesting proposition is that of finding the minimum redundancy level, which leads to the maximum QoE improvement. By contrast if we adopt a P2P approach that succeeds in keeping traffic away from the core network, we have a good chance that redundancy does not always result into network congestion. Our aim is to verify this hypothesis and better understand its implications.

B. Redundancy study

In order to study the effect of redundancy in relation to both QoS and QoE we first measure relevant parameters (as detailed in section IV) for a Joost-like system [17], which in practice chooses sources randomly, and use this as a benchmarking function. Redundancy is increased from 0 (1 source per destination) to 4 (5 sources per destination).

We then compare this with our proposed approach, in which sources are forced to handover continuously (as in Joost) - to ensure computational load balancing – but are not chosen randomly.

We prioritize the choice of sources based on their mutual distance from the destination. In essence we adopt a previously published hierarchical RTT monitoring approach
[15] to maintain a list of sources \( \{S_i\} \), ranked based on their distance from the recipient (\( R \)). Periodically, a new set of sources \( \{S_i^{new}\} \) is chosen from this pot and handover is forced from \( \{S_i\} \) sources to \( \{S_i^{new}\} \) sources. Our hypothesis is that this forced handover strategy does not impact network congestion if traffic is kept away from the core network. We wish to establish, however, until which point we can increase redundancy without triggering network congestion.

IV. ASSESSMENT METHOD

We study the effects of redundancy on QoS and QoE, for each of the two cases under scrutiny: 1) Randomized scenario (new sources are chosen randomly); and 2) localized scenario (new sources are chosen based on minimal mutual distances from the recipient. We add background traffic into the simulated network in order to simulate lightly and heavily congested networks, respectively. Simulation parameters and design and describe below.

A. Evaluation metrics

Packets loss ratio: usually this is defined as the ratio between the dropped and transmitted data packets. This gives an account of efficiency and of the ability of the network to discover routes. However, in this study, a new way of calculating the packets loss ratio is presented in order to study the particular issues relating to redundancy. In our case a packet is actually transmitted by several sources. So a packet is considered lost if is never received through any of the streams. We use the following formalism:

\[
X_{ij} = \begin{cases} 
1 & \text{if } P_i \text{ sent by } d_j \text{ is lost} \\
0 & \text{if } P_i \text{ is received}
\end{cases}
\]

The decision as to whether a packet is lost or not will be according to the following Cartesian product:

\[
PL_i = \prod_{j=1}^{d} X_{ij}
\]

Therefore, if \( P \) is the total number of packets forming a given stream, the packet loss ratio will be expressed as in equation 1

\[
PL = \frac{\sum_{i=1}^{P} PL_i}{P}
\]

Average end-to-end delay: The average time span between transmission and arrival of data packets. This includes all possible delays introduced by intermediate nodes for processing and querying of data. End-to-end delay has a detrimental effect on real-time IPTV. This can be countered only to some extent by buffering techniques.

**Peak Signal to Noise Ratio**: PSNR is an objective way to measure the quality of the transmitted video, which eventually affects the user QoE.

![Figure 2 - Packets loss ratio – lightly loaded network](image)

![Figure 3 - Packets loss ratio – heavily loaded network](image)

B. Simulation setup

The proposed approach was implemented and tested on the ns-2 network simulator [16]. Moreover, various parameters were set on the used topology. First of all, link bandwidth is distributed to all the links evenly (2 mbps) with the same delay; so, all the participants’ peers have the same characteristics. IP as the network protocol and UDP as the transport protocol have been chosen. For simulation of video traffic, “Paris” sequence of CIF resolution with 4:2:0 formats, was H.264/AVC coded and its packets were sent from one and multiple peers to the receiver.

Secondly, in order to load the network, it was imperative to set CBR background traffic to vary network load and enable us to study the localized multi-stream approach under different conditions. The CBR traffic has been setup from different sources to different destinations, with a 512 byte packet size. This background traffic operates during the whole duration of the simulations.
The localized and randomized approaches have been simulated independently and repeated 10 times. The presenting results corresponds the average results of these simulations.

When we do increase redundancy, we immediately observe that packet loss decreases thanks to the fact that multiple copies of the same packets are now sent to the recipient. In our case, packet loss goes down to zero when at least 3 senders are used, indicating that this would be the ideal choice for this particular configuration.

Figures 2 and 3 show our findings about packet loss, for the cases of lightly and heavily networks, respectively. Noticeably from Figure 2 is that both scenarios do not lead to any packet loss up to a redundancy level equal to 4. This is because, although we are increasing the number of streams and, thus, traffic, the network only reaches a congestion state when 4 sources are in use. The benefits of the locality feature of our approach are apparent, since in our case the network can sustain up to 5 streams. By contrast, the Joost-like approach, by selecting sources randomly, distributes traffic across the network and incurs bottlenecks earlier on.

The network-friendly behavior of our approach is even more apparent from Figure 3. In this case, the network is brought close to congestion levels by the background traffic (not by the P2P streams under scrutiny). Because of that, there is packet loss even when only one sender is in action.

Looking now at end-to-end delay (Figures 4 and 5), we can draw similar conclusions and we can follow the same rationale as for packet loss. Localization does have a significant, positive effect on communication latency. More specifically, when we looked at the variation of E2E delay, i.e. jitter; we noticed that our approach is even more beneficial.

The localized and randomized approaches have been simulated independently and repeated 10 times. The presenting results corresponds the average results of these simulations. When we do increase redundancy, we immediately observe that packet loss decreases thanks to the fact that multiple copies of the same packets are now sent to the recipient. In our case, packet loss goes down to zero when at least 3 senders are used, indicating that this would be the ideal choice for this particular configuration.

Figures 2 and 3 show our findings about packet loss, for the cases of lightly and heavily networks, respectively. Noticeably from Figure 2 is that both scenarios do not lead to any packet loss up to a redundancy level equal to 4. This is because, although we are increasing the number of streams and, thus, traffic, the network only reaches a congestion state when 4 sources are in use. The benefits of the locality feature of our approach are apparent, since in our case the network can sustain up to 5 streams. By contrast, the Joost-like approach, by selecting sources randomly, distributes traffic across the network and incurs bottlenecks earlier on.

The network-friendly behavior of our approach is even more apparent from Figure 3. In this case, the network is brought close to congestion levels by the background traffic (not by the P2P streams under scrutiny). Because of that, there is packet loss even when only one sender is in action.

By contrast, the Joost-like approach cannot bring packet loss down to zero; since streams are not kept local and, thus, those streams actually contribute to packet loss by increasing the level of congestion.

Looking now at end-to-end delay (Figures 4 and 5), we can draw similar conclusions and we can follow the same rationale as for packet loss. Localization does have a significant, positive effect on communication latency. More specifically, when we looked at the variation of E2E delay, i.e. jitter; we noticed that our approach is even more beneficial.

Having looked at the network QoS parameters, we now examine PSNR, a parameter that gives an account of the stream quality as perceived by the end user. We assume that the stream is coded and decoded using H.264/AVC encoder JM15. Figures 6 and 7 exhibit a behavior compatible with
our QoS findings (Figures 2 to 5). When the network is lightly loaded, there is hardly any packet loss up to a redundancy of 4, at which point the randomized scenario generates congestions and, thus, a drop in PSNR. Our approach does not generate congestion (packet loss) and maintains a constant level of QoE even when we inject 5 redundant streams. Figure 7 is consistent with this rationale too. Noticeably, our approach improves PSNR steadily up to a redundancy of 3, when the quality reaches the maximum theoretically achievable (packet loss is zero at that point).

VI. CONCLUDING REMARKS

A large variety of popular applications, including VoD, live TV and video conferencing, make use of P2P streaming frameworks. These have emerged from the fundamental principles of insulation and abstraction between the network and the application layers. With this regard, several studies published recently (e.g. [19]), including also some by the authors of this article (e.g. [14] [20] [21]), have identified that when the P2P overlay is designed in isolation from the underlying physical network, the P2P stream has detrimental effects on the network itself. To aim for scalability and user QoE, P2P solutions adopt redundancy, caching, statistical handovers and other similar techniques, which generate substantial network management and control problems to the network operator.

This problem motivates our initial work aimed at studying ways to maintain the QoE and scalability of the overlay, whilst reducing its detrimental effects onto the underlying system. This article represents our initial attempt to pursue network-friendly P2P streaming. Our initial hypothesis that the combination of network locality and multi-streaming can lead to significant improvements is reinforced by the initial findings presented herein.

The difficulty in realizing this approach in practical systems is that it entails breaking the concept of network insulation from the application. In our current work we plan to further study the potential of other cross-layer optimization techniques, which opens the way towards stimulating research. Once the overlay is made aware of the underlying system, or vice versa, the potential of other techniques can be unleashed. For instance, we are studying the use of machine learning for the purpose of correlating network and application conditions and building a network-friendly overlay network.

REFERENCES


