Hindawi Publishing Corporation
International Journal of Digital Multimedia Broadcasting
Volume 2010, Article ID 843574, 12 pages
doi:10.1155/2010/843574

Research Article

Localized Multistreams for P2P Streaming

Majed Alhaisoni,[1] Mohammed Ghanbari,[1] and Antonio Liotta[2]

[1] *School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, UK*
[2] *Electrical Engineering Department and Mathematics and Computer Science Department, Technische Universiteit Eindhoven, P.O. Box 513, PT 11.29 5600 MB Eindhoven, The Netherlands*

Correspondence should be addressed to Majed Alhaisoni, malhai@essex.ac.uk

Received 7 June 2009; Revised 26 November 2009; Accepted 3 February 2010

Academic Editor: Stan Moyer

Streaming video over the Internet, including cellular networks, has now become a commonplace. Network operators typically use multicasting or variants of multiple unicasting to deliver streams to the user terminal in a controlled fashion. P2P streaming is an emerging alternative, which is theoretically more scalable but suffers from other issues arising from the dynamic nature of the system. Users' terminals become streaming nodes but they 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 to find 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, redundant streaming and locality awareness, 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.

1. Introduction

Overlays and P2P (peer-to-peer) systems, that have originally been developed as alternatives to IP multicast and to provide file sharing services, now 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 of increasing 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, that is, send multiple streams to the same user in order to reduce packet loss. The downside is that redundancy increases traffic load, thus increasing congestion and hence reducing network utilization.

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

2. Related Work

As we are dealing with network QoS, QoE, 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 the 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 an estimate of the quality perceived by the user. This study was concerned on how to influence 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 substreams 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 load and the user QoE. Also we emphasize on techniques to choose intercommunicating peers based on their mutual proximity, to keep traffic local and minimize the impact on the network load.

Overlay locality is also studied 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 distribute computational load as well as network efficiency (as explained in [8, 9]).

Hefeeda et al. [10] 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 [11], 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 [12] 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 [13, 14]. In particular, they show how to maximize the throughput in divisible load applications.

Moreover,Locher et al. [15] proposed a distributed hash table which is suitable for high dynamic environment. Their work was designed to maintain fast lookup in terms of low delay and number of routing hops. In their work, number of hops was the main metric which used to determine locality-awareness. According to their work, neighboring nodes are grouped together to form a clique. Nodes share the same ID in a clique; moreover, the data will be replicated on all the nodes on the clique to avoid data loss.

Additionally, a clique has an upper and lower bound in terms of the number of nodes, such that cliques are forced to merge or split. Another aspect of their work is to assume that all the nodes are distributed uniformly in a two-dimensional Euclidean space. However, this may not work in a large network such as the internet. In addition, the link structure is updated periodically in order to establish a structured network. On the other hand, their proposal is based on pining nodes to join the closet clique which will drastically introduce extra signaling overhead.

Another study similar to [15] was conducted by Asaduzzaman et al. [16]; their proposal was built on top of [15], with some modifications by introducing stable nodes (supernode) and replicating the data among the stable nodes only. However, their proposal elects one or more stable nodes of highest available bandwidth in each cluster and assigns special relaying role to them.

Their work is based on a combination of tree and mesh architectures where the nodes on the clique form a mesh and the stable nodes are connected in a tree structure.

For each channel, a tree based is formed between the stable nodes including only one stable node in each clique. However, stable nodes are elected based on their live session. So, in this case a clique may have more than a stable node. The downside to this approach is that the relaying nodes (super nodes) are forming a tree; so reconstructing them in case of failures and peers churn will be costly and can introduce some latency.

By contrast to the abovementioned two works, our proposal aims not only to retain the benefits of redundancy (QoE) but also to reduce its detrimental side effects on the network. We study the combination of two techniques, multistreaming and network locality, while in [15, 16] they are mainly concerned with network locality. We prioritize the choice of sources based on their mutual distance from the destinations. In essence we adopt a previously published hierarchical RTT monitoring approach [17] to maintain a list of sources $\{S_{\ i}\}$, ranking their order based on their distances from the recipient (R). Periodically, a new set of sources $\{\mathrm{S}_{i\mathrm{new}}\}$ is chosen from this pot and handover is forced from $\{S_i\}$ sources to $\{S_{i\mathrm{new}}\}$ sources.

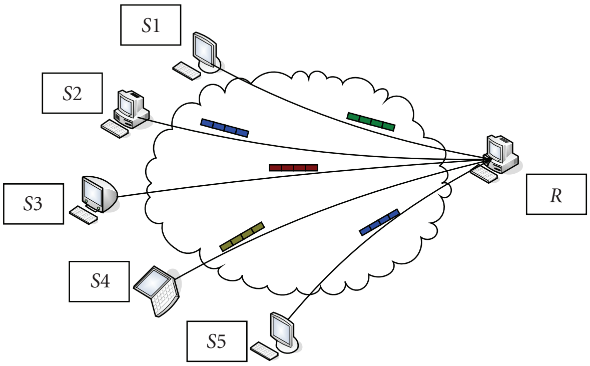


Figure 1: Proposed Architecture.

The hypothesis here is that this forced handover strategy does not impact network congestion if traffic is kept away from the core network. We aim to establish, however, until which point we can increase redundancy without triggering network congestion. Moreover, our work not only introduces locality-awareness and multiple-streams but also loads balances computing and network resources. Additionally, we introduce a new way of calculating the packet loss ratio, end-to-end delay, network utilization with upper and lower bonds and receiver utilization with the stream redundancy.

Furthermore, QoS and QoE are tested out by transmitting a video to examine the scalability and resilience of this proposition. In an earlier publication [9], we have examined the locality-awareness and computational efficiency and compared them to some popular P2P streaming applications. On the other hand, from the locality-awareness viewpoint, our proposal is similar to [15, 16] but with a different aim and approach.

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 multistreaming),

(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.

3. Proposed Approach

3.1. Target Architecture. In our study the number of redundant (multi)streams varies from 1 to 5 as shown in Figure 1. Sources $\{S1 \cdots S5\}$ 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 interpeer distances to ensure 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 issue in previous publications [8, 9].

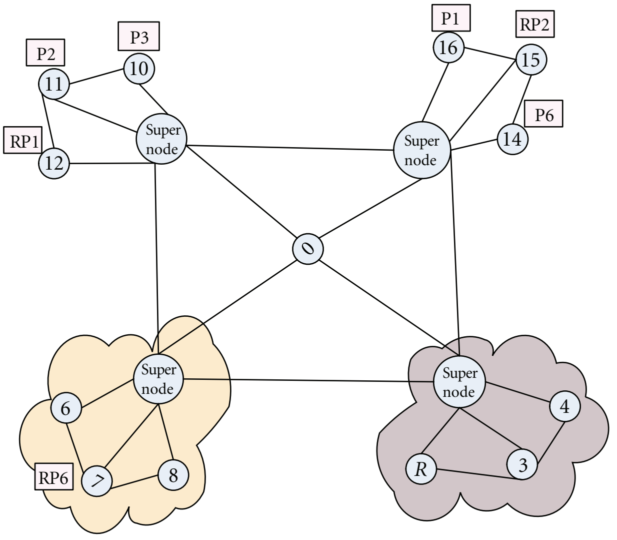
Instead, herein we are mainly interested in understanding whether location-aware P2P techniques can actually reduce the detrimental side effects of P2P redundant streaming. Under architectures other than P2P (unicast, multiple unicasts, and multicast), redundant streams reduce packet loss but have the side effect of increasing network congestion. Interestingly in P2P, redundant streams act as backup of each other and although more redundant streams increase congestion, they actually reduce packet loss. Thus an interesting question is what is the optimum redundancy level that can lead to maximum QoE improvement? By contrast if we adopt a P2P approach that succeeds in keeping traffic away from the core network, we have a better chance that redundancy does not directly result into network congestion. Our aim is to verify this hypothesis and better exploit its implications. However, before dealing with the redundancy issue, another important factor in this proposition is the locality-awareness; so next section is given a wider view of how this is introduced in the proposed study.

3.2. Locality-Awareness. Network efficiency (locality) is the ability to keep traffic as local as possible, which can be achieved by connecting to those peers which are nearby and changing the sources among the participants. Therefore, in the proposed method, a decision is made among the participant peers based on the measured RTT values by the monitoring system. Peers are prioritized on the order of lower RTT values, and the connections are setup based on these values. Consequently, this will not only maintain the network locality among the intercommunicating nodes but it will also improve the QoS and, hence, the user's quality of experience (QoE).

However, offering network locality only without changing the sources among the peers would be drastically impairing load balancing or, in other words, the load distribution between the network and the computing sources. Therefore, different techniques are embedded to the proposed method. The main aim of these techniques is to distribute the load among the participants and at the same time having the network locality not impaired. This can be shown in the next section.

3.3. Computing and Network Resources Load. In order to maintain the load balancing among the contributing peers, different handover techniques have been embedded into the proposed approach. Two conditions trigger the handover among the interconnected peers.

Switching Over. Since the network may experience various constraints such as congestion, bottleneck, and link failures, the RTT values will be severely affected and may not be reliable. Additionally, these stochastic conditions will drastically affect the network locality and degrade the quality of service (QoS) parameters such as throughput, packet loss, and end-to-end delay. There is also another important requirement arising directly from the adoption of P2P: peers are not reliable entities and cannot be assumed to be always connected. Nodes may leave and join at unpredictable times. So, we must adopt a mechanism which allows the receiving peers (in client mode) to maintain a continuing reception of



P: Packet
RP: Redundant packet
R: Receiver
CBR: Source
Sink: Destination

Figure 2: Simulation Topology.

video, although the streaming peers (in server mode) are not constantly available.

One solution to this requirement is that any intending client should regularly update the neighbor's list and reorder them based on the lower RTT values. In our implementation, we keep a ranking list of the peers based on their RTT values. Each peer streams from $\{S1 \cdots S5\}$ other peers. At the switching over time, a new set of the top peers are chosen (those with lower RTT values to the peer under consideration).

Enforced Handover. Another favorable property in the proposed method is its computational efficiency. This can be achieved when the load is periodically distributed among the peers. Under normal network conditions, peers with lower RTT are selected; but when link latency changes, switch over is applied and new peers having lower RTT values are selected.

Some peers may not experience any constraints such as congestion, bottleneck, and link failures. The RTT values will not be affected and may not be changed; so those peers may become the best in every periodical check. Therefore, selecting them regularly would impair computational load balancing among the peers. To avoid this condition, enforced handover is applied.

Furthermore, to avoid pure randomness on the enforced handover process, network locality is applied into clusters of peers, named superpeers, similar to the one adopted in KaZaA [18]. Thus, peers are grouped and they are managed by a special peer, or a super node (Figure 2 depicts a sample of the used topology). Our experiments have confirmed that peers on the same cluster share nearly the same RTT values.

3.4. Redundancy Principle. In order to study the effect of redundancy in relation to both QoS and QoE, we first measure relevant parameters (as detailed in Section 4) for a Joost-like system [19], which in practice chooses sources randomly and is used here as a benchmark. 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 [17] to maintain a list of sources $\{\mathbf{S_i}\}$, ranked based on their round trip delay distances from the recipient ($\mathbf{R}$). Periodically, a new set of sources $\{\mathbf{S_{inew}}\}$ are chosen from this pot and handover is forced from $\{\mathbf{S_i}\}$ sources to $\{\mathbf{S_{inew}}\}$ sources. Our hypothesis is that this forced handover strategy does not have ill effects on network congestion if traffic is kept away from the core network based on locality. We aim to determine, however, until which point we can increase redundancy without triggering network congestion.

4. 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 (locality) from the recipient). We add background traffic into the simulated network in order to simulate network performance under congestion. Simulation and design parameters are described below.

4.1. Simulation Setup. The proposed approach was implemented and tested on the ns-2 network simulator (http://isi.edu/nsnam/ns/). A sample of the used topology is shown in Figure 2. Although the used topology is fixed in the simulation, the two scenarios are treated in a different way. So, for the randomized scenario, senders and receivers are chosen randomly for every run. On the localized (proposed) scenario, the senders are chosen based on locality and the receivers are selected randomly for every run. This gives the advantage of testing the localized scenario under different conditions over the used topology.

Moreover, various parameters were set on the used topology. First of all, each link has a bandwidth of 2 Mbps with equal length (delay). However, the actual delay will be according to the nodes distance of each other; 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, the "Paris" video clip of CIF resolution with 4:2:0 format was H.264/AVC coded and the video packets were sent from one and multiple peers to the receiver.

Secondly, in order to overload the network, it was imperative to set the CBR background traffic to vary the network load and enable us to study the localized multistream approach under different loading conditions. The CBR traffic was setup from different sources to different destinations, with a 512 byte packet size. This background traffic operates during the whole duration of the simulations. This was set to 1 Mbps for the moderately congested network and 1.7 Mbps for the heavily congested network; this is added to the stream video on the running simulation.

The localized and randomized approaches were simulated independently and repeated 10 times. The presented results correspond to the average values of these simulations.

4.2. Evaluation Metrics

Packet Loss Ratio. Packet loss ratio is usually defined as the ratio of the dropped over the transmitted data packets. It gives an account of efficiency and of the ability of the network to discover routes. However, in P2P communication, a new way of calculating the packet loss ratio needs to be defined to study the particular issues relating to redundancy. In our case a packet is actually transmitted by several sources and is considered lost only if it is never received through any of the streams. This is formalized as follows:

$$P_i: \quad \text{generic packet } (i) \text{ sent by all source nodes,}$$

$$d_j: \quad \text{sending or source nodes,}$$

$$X_{ij} = \begin{cases} 1, & \text{if } P_i \text{ sent by } d_j \text{ is lost,} \\ 0, & P_i \text{ is received.} \end{cases} \tag{1}$$

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

$$\mathrm{PL}_i = \prod_{j=1}^{d} x_{ij}. \tag{2}$$

Therefore, if P is the total number of packets required to reconstruct a given stream, the packet loss ratio will be

$$\mathrm{PL} = \frac{\sum_{i=1}^{P} \mathrm{PL}_i}{P}. \tag{3}$$

Receiver Usage. Receiver usage is defined as the ratio of received useful packets over the number of packets required (P) to reconstruct the video as

$$R_U = \frac{P - \mathrm{PL}}{P}, \tag{4}$$

where PL is the packet loss ratio, thus, this parameter increases with reduction on the packet loss ratio, and it can be an efficient tool of P2P performance analysis.

4.2.1. Network Utilization. In point-to-point communication (*i.e., Client-Server*), it is common that network utilization is defined as the ratio of received packets over the sent packets, but in P2P paradigm, when some of the sent or received packets are not useful, this definition is not helpful. Also, since lost packets from any source may not have any ill effect on the received quality as long as these packets can be received from other peers, thus, in the multistream, one can define an upper and lower bound to the network utilization, where the actual utilization stands somewhere in between.

Upper Bound. Upper bound is defined as the ratio of received useful packets over the total number of received packets from all the sending peers. That is,

$$U_b = \frac{P - \mathrm{PL}}{\sum_{i=0}^{n} R_i}, \tag{5}$$

where U_b represents the upper bound utilization, P is the required number of packets to reconstruct the video, and n is the number of sending peers. R_i is the total number of received packets from the ith peers and PL is the lost packets

Lower Bound. Lower bound is defined as the ratio of received useful packets over the total number of sent packets from all the peers. That is:

$$L_b = \frac{P - \mathrm{PL}}{\sum_{i=0}^{n} S_i}, \tag{6}$$

where L_b represents the lower bound utilization, P is the number of required packets to reconstruct the video, n is the number of sending peers, and S_i is total sent packets by the ith peer and PL is the lost packets.

The difference between the upper and lower bounds is in fact due to some stray packets of the senders that are lost in the network. Although these stray packets may not be useful to the receiver, as the duplicate packets from the other peers will replace them, nevertheless their presence in the network can congest the network. Thus the difference between the lower and upper utilization is an indication of the side effect that multistream may have on the network. The lower the

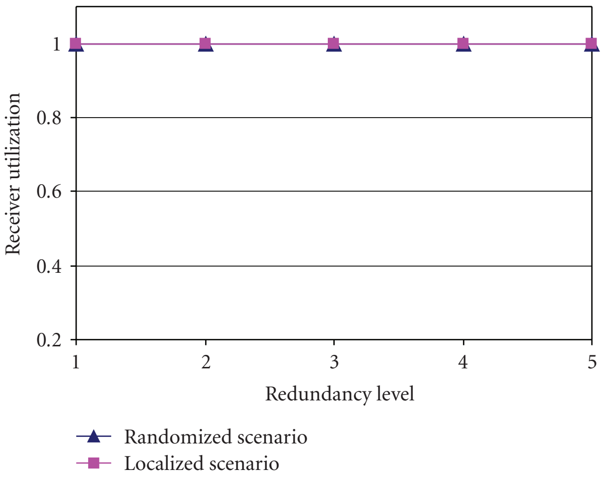


Figure 3: Receiver Usage—Moderately Loaded.

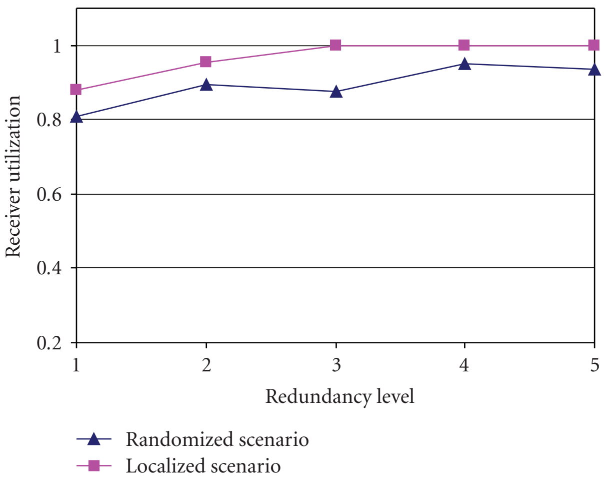


Figure 4: Receiver Usage—Heavily Loaded.

difference is, the better is the performance of the P2P stream. This difference, Nu is thus

$$Nu = \frac{P - \mathrm{PL}}{\sum_{i=0}^{n} R_i} - \frac{P - \mathrm{PL}}{\sum_{i=0}^{n} S_i}. \tag{7}$$

Average End-to-End Delay. Average end-to-end delay is average time span between transmission and arrival of data packets. In multistreams P2P, delay of each received packet is the minimum delay among all the packets of the same type sent by all senders. This includes all possible delays introduced by the intermediate nodes for processing and querying of data. End-to-end delay has a detrimental effect on real-time IPTV.

Peak Signal to Noise Ratio. PSNR is an objective quality measure of the received video, taken as the user QoE. It is defined as the logarithm of the peak signal power over the difference between the original and the received captured video:

$$\mathrm{PSNR} = 10 \log \frac{225^2}{E^2}. \tag{8}$$

5. Simulation Results

In order to assess the proposed scheme, different network parameters should be identified, tested, and evaluated. Therefore, the following parameters have been considered.

5.1. Receiver Utilization. Figures 3 and 4 show the receiver utilization, as defined in (4). For moderate network load our approach is almost similar to the benchmarking scheme which is based upon a random scenario. At higher network load shown in Figure 4, the relative superiority of localized over random connection is evident. The figure shows that at all redundancy levels, the localized connection has a better utilization. Moreover, for more than 3 levels of redundancy (receiving packets from more than 3 peers), the localized approach reaches 100% utilization. Beyond this value, increasing the number of sending peers does not increase receiver utilization. This figure also implies that 3 sending peers are sufficient enough to bring receiver utilization defined in (4) to 100% saturation level.

On the other hand, the randomized connection can never achieve the 100% utilization. Thus in this case there will always be packet losses, irrespective of how many sending peers are chosen, although increasing the number of sending peers improves receiver utilization and hence reduces the packet loss rate.

5.2. Network Utilization. For the network utilization, upper and lower bounds are defined and measured for both scenarios under moderately and heavily congested network loads.

5.2.1. Upper Bound. For the moderately loaded network which is not included here, the two scenarios almost behave similarly since the packet loss is almost negligible (see Figure 8). On the other hand, in a heavily loaded network, the upper network utilization shown in Figure 5, the localized method shows lower utilization than the randomized method. As mentioned before this parameter on its own may not be a good quality indicator, but its deviation from the lower bound is a better indicator.

5.2.2. Lower Bound. Like the upper bound, the lower bound for the moderately loaded network does not show any noticeable difference between the localized and randomized methods. However, for heavily load network shown in Figure 6, the localized method shows a better utilization than the randomized method. The lower bound can show actual network utilization better than the upper bound, since the lower bound indicates how much of the received packets are useful.

Perhaps the best indication of network utilization is the departure of its higher and lower bounds, as shown in Figure 7. This is because the larger difference between them can indicate that some of the sent packets are lost in the network, congesting it at the cost of receiving packets

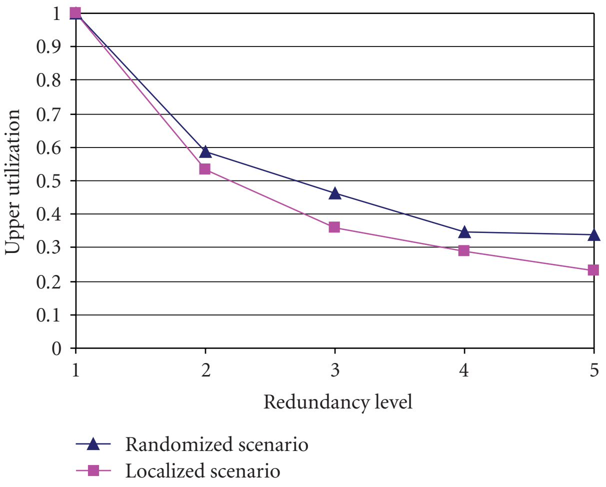


Figure 5: Upper Utilization—Heavily Loaded network.

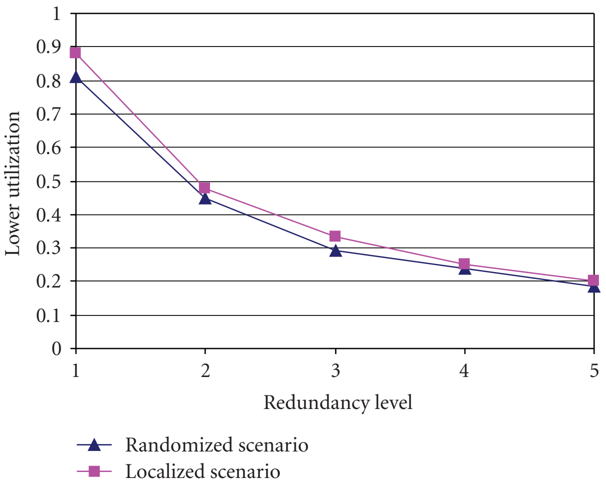


Figure 6: Lower Utilization—Heavily Loaded Network.

from peers to improve video quality. Hence, the smaller the difference between upper and lower bounds of network utilization, the better is the performance, which is in favor of localized method, as shown in Figure 7.

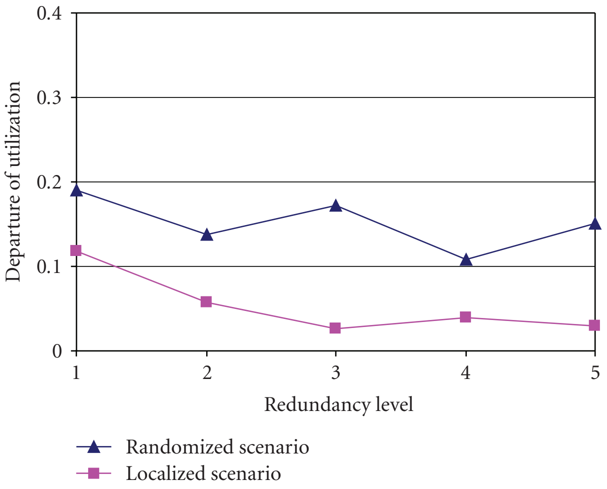


Figure 7: Departure of Utilization—Heavily Loaded.

5.3. Quality of Service. Quality of Service was considered to determine whether the proposed approach is also actually affecting the quality at the application (or user) level. In P2P networking, Quality of service is inherited to different metrics on the network. These metrics are intrinsic to each other. For this reason, to quantify and test the localized multistream approach, different effective parameters have been presented and used here which reflect the performance of this proposition.

5.3.1. Packets Loss. Figures 8 and 9 show our findings of packet loss, for the cases of moderately and heavily loaded networks, respectively. At moderate network load shown in Figure 8 both methods do not lead to any packet loss up to a redundancy level equal to 4. This means that for randomized redundant streams, up to 4 sending peers, enough redundant packets can be received to compensate for any losses. However beyond 4, the added traffic creates congestion that leads to more losses, such that backed up redundant packets may also be lost. With the localized method, the figure shows that even up to 5 redundant streams does cause any packet loss. This is due to the fact that no matter how the network is congested, there is always enough number of redundant packets to be used at the receiver.

A network friendly behavior of the locality aware approach is even more apparent at higher network load, as shown in Figure 9. In this case, the network is brought close to congestion by the background traffic (not by the streams under scrutiny). The network is severely congested; then even one or two senders in action can lead to packet loss. By increasing the redundancy level (e.g., more senders), the packet loss is reduced in both methods. However, with the localized method, there is almost no packet loss after receiving from 3 senders. This is due to the fact that multiple copies of the same packets are now sent to the recipient.

By disparity, the randomized connection of Joost-like approach cannot bring packet loss down to zero. In this case, even increasing the number of senders (more redundancy), the senders themselves create additional congestion such that, beyond 4 senders, congestion increases as shown in Figure 9.

5.3.2. Average End-to-End Delay. Delay is another important quality of experience parameter. Figures 10 and 11 show the average end-to-end network delay, for the moderately and heavily congested scenarios, respectively. It is important to note that at heavy network load, the end-to-end delay under localized connection has the least value at the redundancy level of 3-4 senders. Considering that packet loss rate is almost eradicated with just about 3 redundant senders (Figure 9), it appears that the optimal redundancy level is comprised between 3 and 4.

5.4. Quality of Experience. Finally, the objective and subjective qualities of the decoded video under both loading

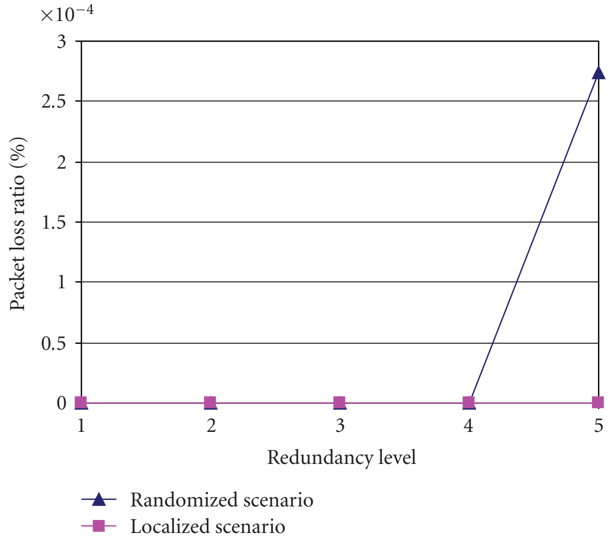


Figure 8: Packets loss ratio—Moderately loaded.

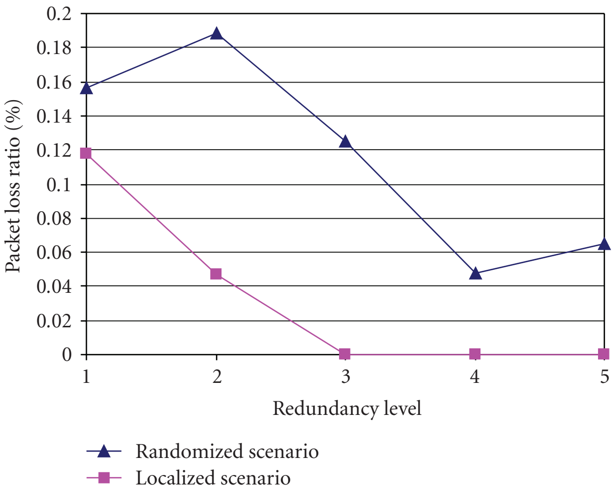


Figure 9: Packets Loss ratio—Heavily loaded.

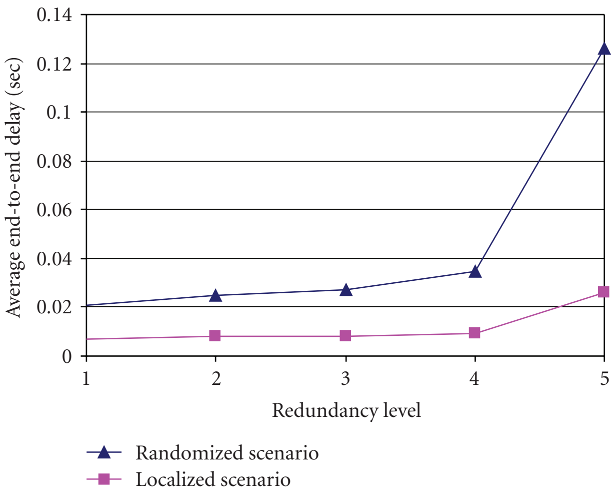


Figure 10: Avg. E2E Delay—Moderately loaded.

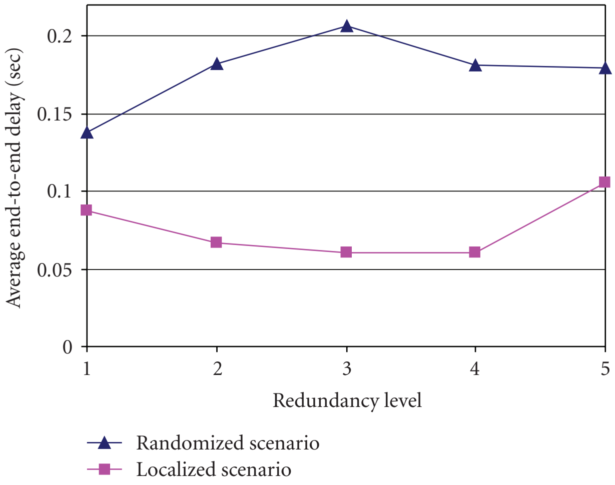


Figure 11: Avg. E2E Delay—Heavily loaded.

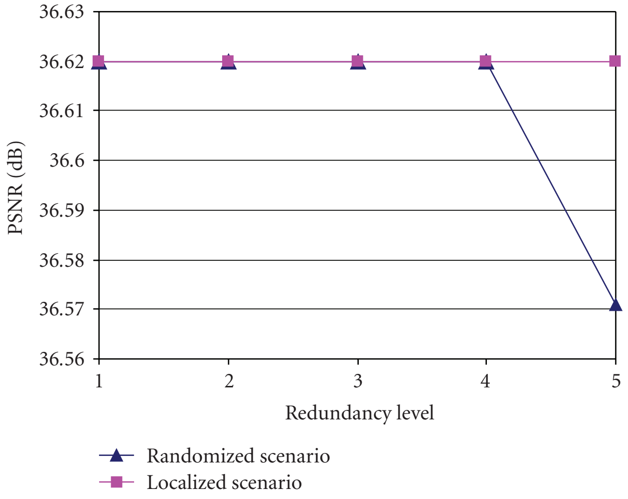


Figure 12: PSNR—Moderately loaded.

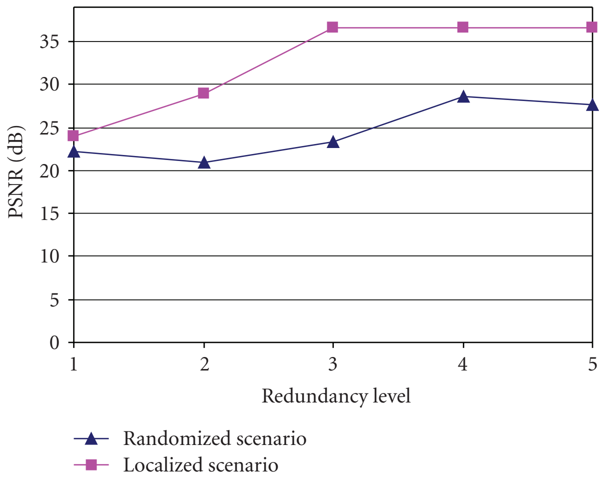
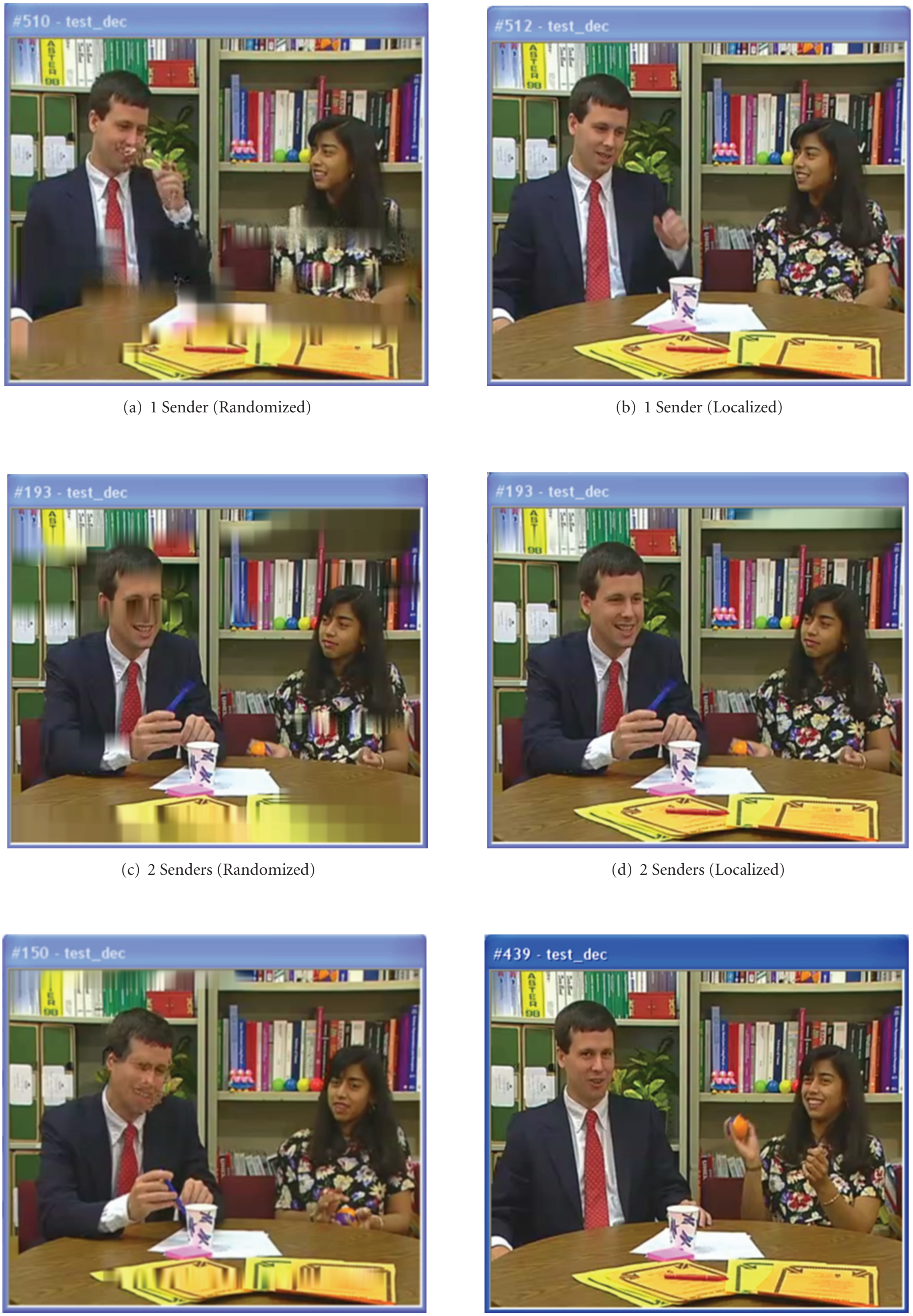


Figure 13: PSNR—Heavily loaded.



(a) 1 Sender (Randomized)

(b) 1 Sender (Localized)

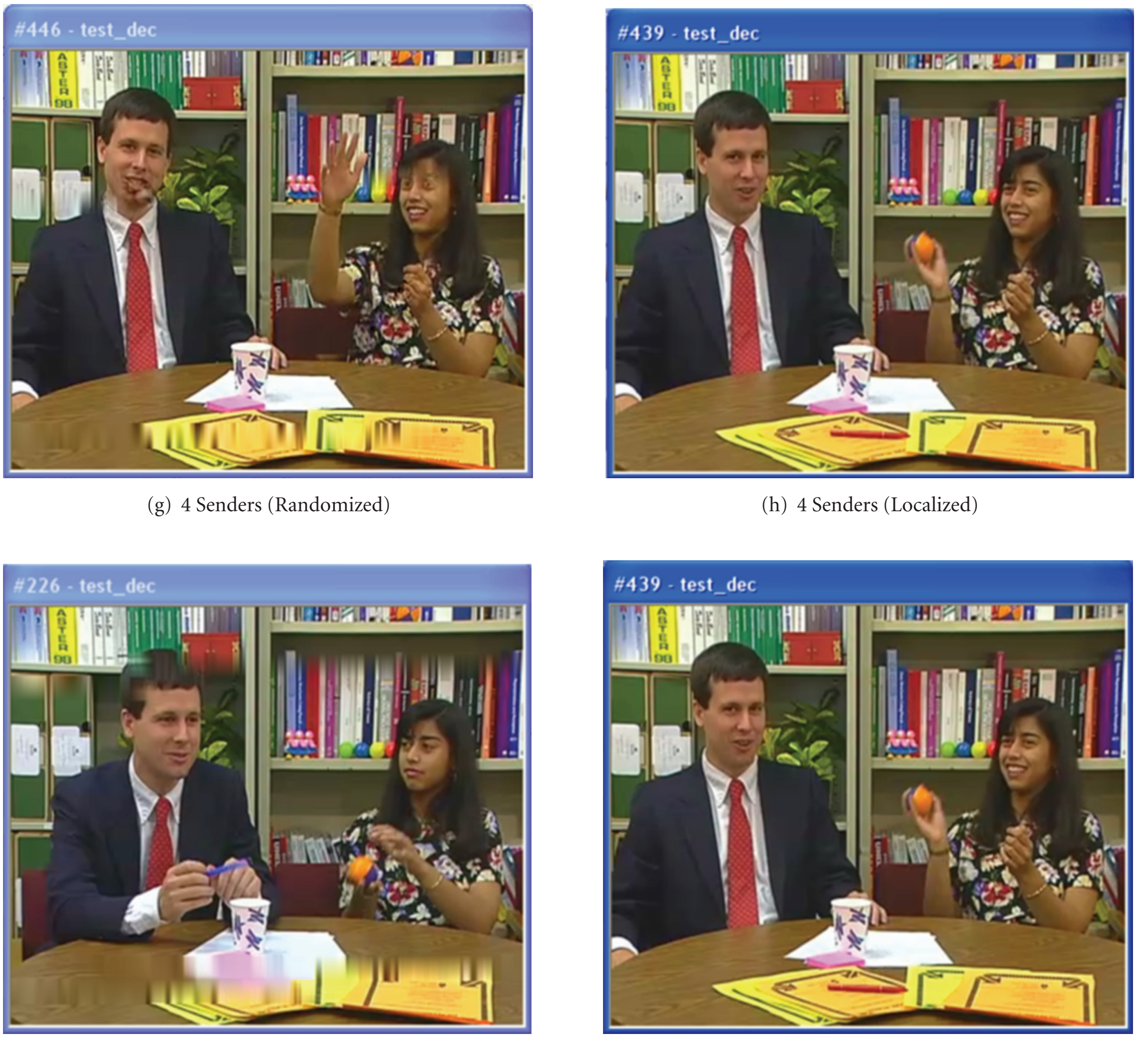
(c) 2 Senders (Randomized)

(d) 2 Senders (Localized)

(e) 3 Senders (Randomized)

(f) 3 Senders (Localized)

Figure 14: Continued.



(g) 4 Senders (Randomized)

(h) 4 Senders (Localized)

(i) 5 Senders (Randomized)

(j) 5 Senders (Localized)

Figure 14: Subjective quality of the localized method against the randomized method for each redundancy level (S1...S5).

conditions are compared. The streams were coded and decoded with an H.264/AVC encoder/decoder of type JM15. Figures 12 and 13 show the objective video quality as measured by PSNR for moderately and heavily congested network perceptively. As expected they exhibit a behavior similar to the packet losses findings of Figures 8 and 9. When the network is moderately loaded, there is hardly any packet lossup to a redundancy of 4, at which point the randomized scenario generates congestions and, thus, a drop in PSNR. The localized approach does not show any packet loss and maintains a constant level of QoE even with 5 injected redundant streams. Figure 13 is also consistent with this rationale. Noticeably, the localized approach improves PSNR steadily up to a redundancy level of 3, where the quality reaches its maximum theoretically achievable value (packet loss is zero at that point).

5.4.1. Subjectivity Quality. For the assessment of the subjective quality perceived by the end-users, in this section snapshots of the two methods are shown. For each case of the five redundant streams, a picture of the proposed localized multistream is compared against the randomized approach (Joost-like). Figures 14(a), 14(b), 14(c), 14(d), 14(e), 14(f), 14(g), 14(h), 14(i), and 14(j) show the subjective quality of the localized method against the randomized methods. Quality of pictures on the table, with the randomized method, specially cup, pen, and papers are distributed in all the redundancy levels, but in the localized method, picture quality is as good as can be compared to the original source.

6. Discussion

In this paper, a localized multi-source is proposed to offer a redundancy of the same content over the network with high quality and low end-to-end delay. This proposal has been tested and run under different scenarios and network conditions. In order to quantify its robustness, it has been run under a point-to-point connection where there is only one sender and one receiver. Furthermore, multiconnections were introduced where the receiver can connect from 2 and up to 5 senders (peers) simultaneously. Redundant streams are used in combination with locality-awareness to assess our initial hypothesis.

Varieties of connections have been run in two extreme congestion levels, moderately and heavily congested. Consequently, different effective parameters on the network have been measured to show and validate how robust and practical is the proposed localized scheme with the offered redundancy by the chosen sending peers.

However, in order to adjudicate on the goodness of the localized multistream, a benchmark of a popular VoD application [20] is compared against this proposal to show how the localized multistream behaves in contrast to the randomized scheme.

For that reason, one of the valuable and insightful network parameter to be considered is the network utilization which gives the impact of this proposition over the network. Our results (Figure 7) have shown that locality-aware combined with stream redundancy has positive impact on the network utilization. Moreover, another vital network parameter is packets loss; to gauge this factor, a new way of measuring packet loss has been defined mainly to quantify the performance of the proposed localized multistream as shown earlier in (3). The presented results show that the localized scheme is performing better across all the quality of service and QoE parameters. Consequently, by looking at the packets loss ratio, it is apparent from Figures 8 and 9 that the localized approach is better and mainly in case of the heavily congestion network. This was mainly due to the offered awareness of sending peers complemented by stream redundancy.

Moreover, end-to-end delay is almost consistent on both congestion levels and particularly from 2 to 4 senders as shown in Figures 10 and 11; this was taken as the minimum delay among all the received packets. On the other hand, QoE is maintained appropriately on the localized multistream. So, Figures 12 and 13 are an extensive evidence of the perceived video quality to the end-consumers. However, the most divergence can be derived between the two compared schemes on Figure 13 where the network is heavily congested, which is the case of the internet traffic nowadays.

Finally, another interesting point that has resulted from the localized multistream is the finding of the required optimum number of peers to serve a client. According to the presented results by the localized multistream, it is so clear from all the figures that 3 to 4 peers are good enough to provide high quality to the end-users within the current configuration. In contrast to that, with the randomized approach, it is difficult to give an indication for that as the interconnections among peers are chaotic and randomly.

7. Conclusion

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., [21]), including also some by the authors of this article (e.g., [9, 22, 23]), have identified that when the P2P overlay is designed in isolation from the underlay 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 underlay. This article represents our initial attempt to pursue network friendly P2P streaming. Our initial hypothesis that the combination of network locality and multistreaming can lead to significant improvements on QoE is reinforced by the initial findings presented herein.

References

[1] Y. Liu, Y. Guo, and C. Liang, “A survey on peer-to-peer video streaming systems,” *Peer-to-Peer Networking and Applications*, vol. 1, no. 1, pp. 18–28, 2008.

[2] Y. Liu, X. Liu, L. Xiao, L. M. Ni, and X. Zhang, “Location-aware topology matching in P2P systems,” in *Proceedings of the 32rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM ’04)*, vol. 4, pp. 2220–2230, Hong Kong, March 2004.

[3] Z. Xu, C. Tang, and Z. Zhang, “Building topology-aware overlays using global soft-state,” in *Proceedings of the 23rd International Conference on Distributed Computing Systems (ICDCS ’03)*, p. 500, May 2003.

[4] S. Banerjee, B. Bhattacharjee, and C. Kommareddy, “Scalable application layer multicast,” *Computer Communication Review*, vol. 32, no. 4, pp. 205–217, 2002.

[5] B. Zhao, A. Joseph, and J. Kubiatowicz, “Locality aware mechanisms for large-scale networks,” in *Proceedings of Workshop on Future Directions in Distributed Computing (FuDiCo ’02)*, Bertinoro, Italy, June 2002.

[6] P. Rodríguez-Bocca, *Quality-centric design of peer-to-peer systems for live-video broadcasting*, Ph.D. thesis, 2008.

[7] T. P. Q. Nguyen and A. Zakhor, “Distributed video streaming over Internet,” in *Multimedia Computing and Networking*, vol. 4673 of *Proceedings of SPIE*, pp. 186–195, San Jose, Calif, USA, January 2002.

[8] M. Alhaisoni, A. Liotta, and M. Ghanbari, “An assessment of self-managed P2P streaming,” in *Proceedings of the 5th International Conference on Autonomic and Autonomous Systems (ICAS ’09)*, pp. 34–39, Valencia, Spain, April 2009.

[9] M. Alhaisoni, A. Liotta, and M. Ghanbari, “Resource-awareness and trade-off optimization in P2P video streaming,” *Advances in Differential Equations*, vol. 4, no. 1, pp. 59–77, 2010.

[10] M. M. Hefeeda, A. Habib, B. Botev, D. Xu, and B. Bhargava, “PROMISE: peer-to-peer media streaming using CollectCast,” in *Proceedings of the 11th ACM International Conference on Multimedia (MM’03)*, pp. 45–54, Berkely, Calif, USA, November 2003.

[11] T. Nguyen and A. Zakhor, “Distributed video streaming with forward error correction,” in *Proceedings of the 12th*

International Packet Video Workshop, pp. 1–12, Pittsburgh, Pa, USA, April 2002.

[12] M. Mushtaq and T. Ahmed, "Adaptive packet video streaming over P2P networks using active measurements," in *Proceeding of the 11th IEEE Symposium on Computers and Communications (ISCC '06)*, pp. 423–428, IEEE Computer Society, Los Alamitos, Calif, USA, June 2006.

[13] M. M. Hefeeda and B. K. Bhargava, "On-demand media streaming over the internet," in *Proceedings of the 9th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS'03)*, p. 279, 2003.

[14] K. Ponnavaikko and D. Janakiram, "Overlay network management for scheduling tasks on the grid," in *Proceedings of the 4th International Conference on Distributed Computing and Internet Technology (ICDCIT '07)*, pp. 166–171, Springer, Bangalore, India, December 2007.

[15] T. Locher, S. Schmid, and R. Wattenhofer, "eQuus: a provably robust and locality-aware peer-to-peer system," in *Proceedings of the 6th IEEE International Conference on Peer-to-Peer Computing*, pp. 3–11, Cambridge, UK, September 2006.

[16] S. Asaduzzaman, Y. Qiao, and G. Bochmann, "CliqueStream: an efficient and fault-resilient live streaming network on a clustered peer-to-peer overlay," in *Proceedings of the 8th IEEE International Conference on Peer-to-Peer Computing*, pp. 269–278, Aachen, Germany, September 2008.

[17] C. Ragusa, A. Liotta, and G. Pavlou, "An adaptive clustering approach for the management of dynamic systems," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 12, pp. 2223–2235, 2005.

[18] http://www.kazaa.com/.

[19] http://isi.edu/nsnam/ns/.

[20] http://www.joost.com/.

[21] H. J. Kolbe, O. Kettig, and E. Golic, "Monitoring the impact of P2P users on a broadband operator's network," in *Proceedings of IFIP/IEEE International Symposium on Integrated Network Management (IM '09)*, pp. 351–358, IEEE Press, New York, NY, USA, June 2009.

[22] N. N. Qadri and A. Liotta, "Effective video streaming using mesh P2P with MDC over MANETS," *Journal of Mobile Multimedia*, vol. 5, no. 4, pp. 301–316, 2009.

[23] A. Liotta and L. Lin, "The operator's response to P2P service demand," *IEEE Communications Magazine*, vol. 45, no. 7, pp. 76–83, 2007.

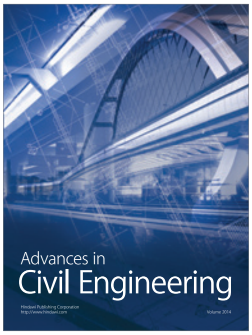
Journal of
Engineering

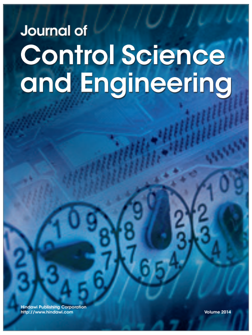
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

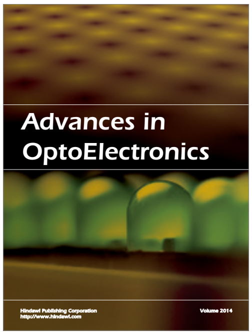
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

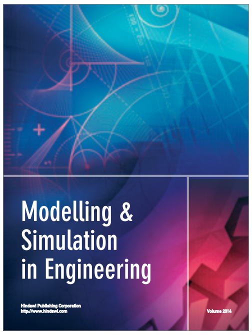
Journal of
Robotics

Journal of
Electrical and Computer Engineering

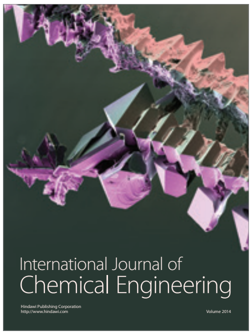

Advances in
OptoElectronics

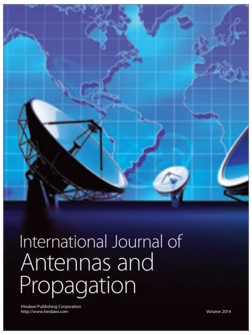
VLSI Design

International Journal of
Navigation and Observation

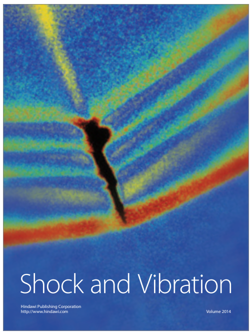

Modelling & Simulation in Engineering

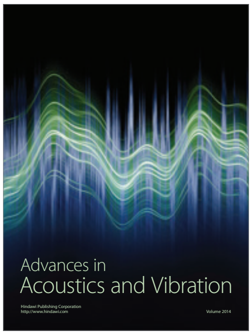
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration