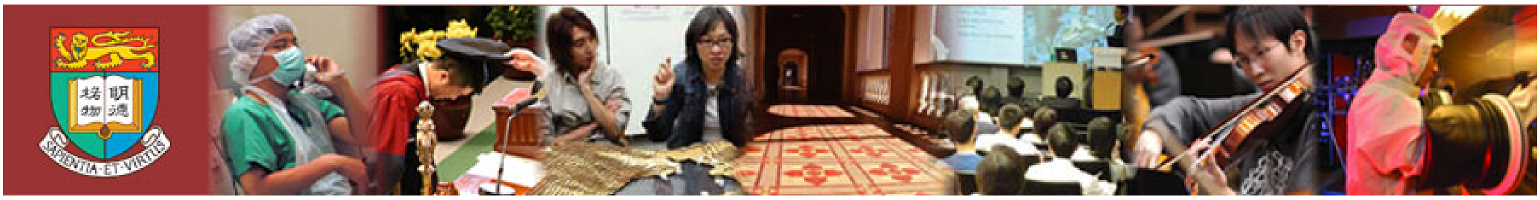

Title	A QoE based performance study of mobile peer-to-peer live video streaming
Author(s)	Fung, KC; Kwok, YK
Citation	The 13th International Conference on Parallel and Distributed Computing, Applications, and Technologies (PDCAT 2012), Beijing, China, 14-16 December 2012, In Conference Proceedings, 2012, p. 707-712
Issued Date	2012
URL	http://hdl.handle.net/10722/189861
Rights	

A QoE Based Performance Study of Mobile Peer-to-Peer Live Video Streaming

Kwok-Chun Fung and Yu-Kwong Kwok
Department of Electrical and Electronic Engineering
The University of Hong Kong

***Abstract*—Peer-to-peer (P2P) Mobile Ad Hoc Networks (MANETs) are widely envisioned to be a practical platform to mobile live video streaming applications (e.g., mobile IPTV). However, the performance of such a streaming solution is still largely unknown. As such, in this paper, we aim to quantify the streaming performance using a Quality of Experience (QoE) based approach. Our simulation results indicate that video streaming performance is highly sensitive to the video chunk size. Specifically, if the chunk size is small, performance, in terms of both QoE and QoS, is guaranteed but at the expense of a higher overhead. On the other hand, if chunk size is increased, performance can degrade quite rapidly. Thus, it needs some careful fine tuning of chunk size to obtain satisfactory QoE performance.**

***Index Terms*—Mobile ad hoc networks, live streaming, overlay networks, peer-to-peer (P2P) computing**

I. Introduction

PEER-to-peer (P2P) networks, as a cost-effective solution for data and multimedia sharing, have been developing rapidly and widely adopted in the Internet. Most notably, real-reployed P2P live video streaming networks (e.g., YouTube [1], PPTV [2], UUSee [3]) is rapidly replacing traditional client-server broadcasting systems.

In the recent couple of years, we have also witnessed the remarkable proliferation of smart phones and other hand-held mobile computing devices (e.g., tablets) with high speed wireless networking capabilities (e.g., equipped with 4G LTE interfaces). Thus, it is widely envisioned that mobile P2P networks will disruptively occupy the Internet. In such a wireless environment, the physical connections among devices are also “peer-to-peer” in the sense that the devices potentially formulate a mobile ad hoc network (MANET) when they come to close proximity. Consequently, it is hard to doubt the eventual proliferation of MANETs running P2P applications. In particular, mobile video content sharing (e.g., live broadcast of IPTV programs) is widely considered to be a killer application with tremendous business potential.

However, to gain support from users of mobile devices, quality must be guaranteed. Unfortunately, the application level performance of a MANET, supporting P2P video streaming, is still largely unknown. As such, in our study, we investigate this performance issue using a simulated MANET platform running real-life video streaming applications (e.g., YouTube). Most importantly, we employ a Quality-of-Experience (QoE) approach in evaluating video streaming quality. Specifically our contributions are:

1) We adopt subjective and objective measurements to quantify the user perceived performance. Our MANET system architecture is based on hybrid content distributed network (CDN). The key feature is that it can scale up the capacity of the Web tier so that the system can handle multiple ad hoc. We believe that such a design choice closely resembles what we will experience in the near future.
2) We provide a remedy to a critical problem commonly found in contemporary live streaming systems. Specifically, we incorporate a lightweight Python script to get rid of the interruption problem of the users' interface which is resulted from the inappropriately defined video packet sizes in live streaming. As detailed in Section II below, this is also an important system design issue that must be addressed in a practical live video streaming system.
3) We perform an extensive performance study on the simulated MANET, supporting practical video streaming (e.g., clips from YouTube). Most importantly, we obtain interesting results which indicate that a shorter or the shortest path in the MANET system may not be the optimum. Indeed, we get outstanding video streaming results by using a value-added service of a passive link estimator, which judiciously evaluates the residual energy and end-to-end delay before establishing the connection and packet forwarding.

There are a number of interesting studies on a similar vein reported in the literature. Dalal [5] applied reduced feature set to predict user perceived stream quality by using two-nearest neighbor algorithms for evaluating the quality rating in specific conditions including video-on-demand services. The results show that a combination of retransmitted and/or lost application layer packets is the most accurate indicator for predicting stream quality. Hosfeld *et al.* [6] demonstrated the QoE PDH (Provisioning-Delivery-Hysteresis) for voice-over IP, live video streaming, and Web browsing. They illustrated the importance to control the packet loss caused by packet jamming. Hosfeld *et al.* [8] also addressed the challenge of assessing and modeling QoE for online video services. They described a QoE model for YouTube video services and delineated the key factors that mould quality perception. They then proposed a subjective QoE assessment methodology for online video based on crowd sourcing. Cheng *et al.* [9]

978-0-7695-4879-1/12 $26.00 © 2012 IEEE
DOI 10.1109/PDCAT.2012.139

presented an empirical investigation about the social structure of YouTube, addressing friend relations and their correlation with tag applied to uploaded videos. Their study shows that social interaction on YouTube appears to be structured in a conventional social network manner but with greater semantic coherence around contents.

Agarwal *et al.* [10] examined the performance of live P2P video multicast sessions in large-scale networks. Their performance analysis study indicated that an efficient video dissemination requires high bandwidth, high peer churn and low peer persistence in the P2P multicast system. Specifically, they measured the QoS of popular video contents. They then correlated the observed quality with peer actions and underlay network, and provided strategies for performance optimization.

Kim *et al.* [17] focused on the critical issues of real-time streaming services such as high connection setup and media delivery latency. These issues inevitably lead to significant service disruptions due to biased peer selection without location awareness. Thus, they proposed a group based CDN-P2P hybrid architecture which provides location and content aware peer selection. In this design, a super-peer performs location aware peer selection by CAN. It also manages peer admission control with content awareness, and focuses on a group of peers with the same channel as the sub-overlay network. Through a detailed performance evaluation, Kim *et al.*'s proposed architecture outperformed others in terms of connection setup delay and media delivery time.

Fesci-Sayit *et al.* [18] proposed a bandwidth-aware system based on multiple multicast trees. The system is built on top of a hierarchical cluster based P2P overlay architecture model for scalable video coding. They assumed the knowledge of end-to-end delay and peer bandwidth in the trees. They then attempted to maximize the quality of received video. Indeed, overall performance of the system is significantly better if peers with higher available bandwidth are placed closer to the root in the tree and peers with lower bandwidth are near the leaves.

We believe that our study nicely complement the above-mentioned pioneering work in QoE based performance study of streaming systems in that none of these previous works is based on a practical MANET model. The rest of this paper is organized as follows. In Section II, we describe in detail our MANET system model so as to provide a clear understanding of what we envision to be a practical mobile video streaming scenario. We then discuss the QoE based evaluation methodology. Specifically, we describe the various performance characterizations in our mathematical model for computing the QoE parameters that we use in our study. Section III reports the detailed simulation results and our interpretations. Finally, Section IV concludes the paper.

II. System Model

A. MANET System Architecture

One of the most critical components in our MANET system architecture is the Landmark-based routing model [12] which provides a free-scale and highly dynamic routing functionality. The model forms a *base network* in which each landmark is operated by both geographical and IP-based location services. The traffic matrices are estimated for selecting an optimal path from request nodes to their destinations under infrastructureless and unpredictable network topology. Specifically, the unique identifier of each landmark is used to map the corresponding location to its nearest distance accurately. The available link is then estimated by the assistant table in vertex-edge graph, where vertices and edges represent mobile nodes and distances among them, respectively (illustrated in Fig. 1). Each link is indexed by 7-tuple of $mobile\_ID_u$, $Landmark\_ID_u$, $Segment\_ID_u$, $mobile\_ID_v$, $Landmark\_ID_v$, $Segment\_ID_v$ and scheduling, *s*. The value of *s* will be increased incrementally. The worst case is that even though *s* is maximized, it would still be possible to suffer from dead-link eventually.

Algorithm 1 –Assistant table

```
//initialization
for (Mu,Lu,Su,Mv,Lv,Sv)do
 if (destination = 0) then
   Assist_Table(Mu,Lu,Su,Mv,Lv,Sv,S) =yes;
  else
   Assist_Table(Mu,Lu,Su,Mv,Lv,Sv,S)=no;
 endif
//get available links
endfor
for s=1 to max. schedule do
  forall (Mu,Lu,Su,Mv,Lv,Sv,S)
    Assist_Table(Mu,Lu,Su,Mv,Lv,Sv,S-1)=yes
    do
    Assist_Table(Mu,Lu,Su,Mv,Lv,Sv,S)=yes
      forall
        Assist_Table(NMu, NLu, NSu, NMv, NLv, NSv) do
      if
        Assist_Table(NMu, NLu, NSu, NMv, NLv, NSv,S-1) =no
      then
Assist_Table(Mu,Lu,Su,Mv,Lv,Sv,S) =no
       break
      endif
    end for
  end for
end for
```

Fig. 1 Algorithm for Available Link Assignment.

In order to get rid of the dead-end problem, it is important to obtain a high quality link and optimize the upper layers' throughput. A quick response IP-based passive link estimator, which requires much less memory, is incorporated to each landmark. Based on the architecture of physical link technologies such as WiFi, the sequence number of each incoming packet will be captured by the link estimator. The use of WMEWMA (Window Mean Exponential Weighted Moving Average) will consequently reduce short-term fluctuations of signal and highlight longer-term trends of signal. The ratio of the received packets to all sending packets can finally be obtained within a pre-defined time interval. The landmark updates and announces its residual energy and end-to-end delay history to its neighbors. Consequently, NF (Nice Forwarding) is computed as in (1):

$$NF = HopCount_{bound} \times \frac{1}{WMEWMA_{LinkQuality}} \times (\mu L_{energy} \oplus \mu \frac{1}{L_{Delay}}) \quad (1)$$

where $HopCourt_{bound}$ is bound nodes distance, $WMEWMA_{LinkQuality}$ represents quality link estimated by

WM-EWMA. L_{energy} and $1/L_{delay}$ are residual energy and link delay respectively. Here, μ is a constant-coefficient multiplier. The normalization values of LinkQuality, energy and end-to-end delay are to maximize the NF. The virtual backbone is built on the landmark election and landmark connection algorithms as shown in Figs. 2 and 3, respectively.

```
Landmark Election
begin
 if metric =max. among node then //metric = throughput
   set as landmark
   inform other nodes
 else
   let j be the edge / neighbor with the max metric (form landmark)
end
```

Fig. 2 Algorithm for Landmark Election.

```
Backbone connection
start
  for every node I in initial hop edge do
  if landmark = I then
    Landmark = true;
    for every node j in edge do
      if landmark=j then
        Landmark =true
        else
          gn = true //general edge node
    else gn = true
  end
start
    for every node i in ...do
      if landmark=i then
         for every node j in ... do
           if j ≠ i && landmark=j && landmark = false then
               landmark←i
           if landmark ≠ j && gn =false then
               landmark←i
               landmark←j
end
```

Fig. 3 Algorithm for Backbone Connection.

B. *QoE Based Evaluation Methodology*

Quality of Experience (QoE) is closely related to Quality of Service (QoS). Indeed, QoE subsumes various QoS parameters and captures user's qualitative evaluation.

How to Quantify User Experience?

Subjective Measurement: We choose the following important QoS parameters (see Table I) for the subjective measurement: Smoothness, Resolution, Audio-to-video synchronization, Churn Rate, and Pause Rate as the Key Performance indicators, *KP*, that are varied from different scenarios and traffic conditions. The parameter values are estimated and recorded in Table I. We derive subjective measurement of QoE to track the degree of user experience, called Key Quantity (*KQ*). For example, Mean Opinion Score (MOS) is often used to indicate the quality of web-TV streaming service. The results are generated from a set of standard and subjective rating tests. Score 5 represents Excellent of imperceptible, 4 represents good with perceptible but not annoying, and 3 represents fair but slightly annoying. Finally, the two lowest MOS are 2 and 1, representing poor and bad with annoying and very annoying respectively. The range

TABLE I: QUALITY RATING

PARAMETERS	Key Quantity (KQ) %
Smoothness, S	38%
Resolution, R	33%
Audio-to-video Synchronization, AV Syn	25%
Churn Rate, CR	3%
Pause Rate, PR	1%

is from 1 (worst) to 5 (best). User should take a trade off when selecting the MOS. For example, YouTube video maintains its high quality in resolution (1280x720). Nonetheless, the dilemma is that it consumes the most bandwidth streaming, and may eventually result in degradation of video quality. Bandwidth can be measured quantitatively, while video quality requires human interpretations.

Objective Measurement: We use KP to measure the QoS for YouTube live streaming. Furthermore, we use KQ to measure the QoE of those YouTube live streaming users through the estimation of MOS values. Then we get a set of KP values for QoS rating. We set up a mapping (see Table II) to show the relationship between performance and the corresponding QoS parameters.

TABLE II: LIVE STREAMING MEASUREMENT

MOS Score of Key Quality (KQ)	% of Key Performance (KP)
Audio MOS Score	Delay, Jitter, Packet Loss
Video MOS Score	Blocking, Jerkiness, Blurry
Audio + Video MOS Score	Delay, Jitter, Packet Loss

The relationships between QoE and QoS have been investigated and formulated in [14], [15], [16], [19] which yielded significant and interesting QoS and QoE results especially in real time video. Specifically, in our study, the method in [6] is adopted to derive Equations (2) to (5) for conducting our experiments.

We denote by $\bar{Q}oS_{qsb}$ the quality standard bounded QoS function. We assign a set of KP values to $\bar{Q}oS_{qsb}$ function and obtain normalized values. Then, we have,

$$\bar{Q}oS_{qsb} = \sum \text{ QoS parameters} \times \text{weight} \tag{2}$$

In (2), the value of $\bar{Q}oS_{qsb}$ is normalized and estimated with summation of values multiplying the measured $\bar{Q}oS_{qsb}$ parameter with assigned weight. We use YouTube as the reference Hybrid CDN P2P for the $\bar{Q}oS_{qsb}$ influences evaluation. We further define three levels of user experience for qualitative measurement. They are Packet Loss level $\bar{Q}oS_{PacketLoss}$, Packet Delivery level $\bar{Q}oS_{PacketDelivery}$, and Resource Contribution level $\bar{Q}oS_{RC}$. We use these parameters in exponential and logarithmic models.

If $\bar{Q}oS_{PacketLoss}$ is high, it represents that it is still far from the expected range. However, if $\bar{Q}oS_{PacketDelivery}$ and $\bar{Q}oS_{RC}$ are high, it implies more improvement in *QoE* and bandwidth contribution. Thus, by using the QoS_{qsb} measurement, we evaluate the QoE through $\overline{Q}oS_{PacketLoss}$, $\overline{Q}oS_{PacketDelivery}$, $\overline{Q}oS_{RC}$.

$$QoE(\overline{Q}oS_{RC}) = A + B\,|\log(\overline{Q}oS_{RC})^{\Omega/R}| \tag{3}$$

$$QoE(\overline{Q}oS_{PacketLoss}) = A - B\,|\log(\overline{Q}oS_{PacketLoss})^{\Omega/R}| \tag{4}$$

$$QoE(\overline{Q}oS_{PacketDelivery}) = A + B\exp(\overline{Q}oS_{PacketDelivery})^{\Omega/R} \tag{5}$$

where $A \geq 0, B \geq 0$ are coefficients for values bounding. The parameter **Ω** is the size of video packet in Mbit for streaming, which critically affects the streaming performance.

Specifically, if chunk size is large, processing overhead is amortized and is effectively lower. Yet it might also result in lower video performance due to longer delay. On the other hand, if chunk size is small, video performance is better but the processing overhead can be very high. In our experiments, we set R and **Ω** to be 1 by default.

C. YouTube Architecture for P2P Live Streaming

Similarly, authors of [4], [7], [18] were aware of the necessity of Hybrid CDN-P2P over live streaming. CDN will directly affect the performance in P2P streaming system because it contains copies of data and it is distributed to different edge locations with assigned Uniform Resource Locator (URL) in the network. However, some factors such as mobility that should be carefully considered in peer selection (e.g., super-peer selection). For example, when a request is coming, the CDN evaluates it and routes to its edge location precisely. It minimizes the bandwidth and latency to access the data from peers. If the speed of peer movements is too high, the peer cannot locate and re-direct the route to other neighbors.

Consequently, the mobility of the peer should be relatively lower than that of the other peers. It enables the peer to access a copy of data to the nearest neighboring-peer. Conversely, in order to avoid the bottleneck problem of the nearest neighboring peer to the server and increase the streaming efficiency, all peers access to the same central server. We scale up the web application architecture to prevent the web tier system from over-provisioning and the associated substantial expense.

Promising Distributed Lighttp Web Services

We adopt the Lighttp web server model for YouTube live streaming because Lightttp is a high performance and efficient web server in terms of security, flexibility and fast-speed. Lighttp sustains memory in a minimal manner, and gets effective management of central processing unit load with advanced feature set, unlike other web servers that serves as a solution for web servers suffered from load problems.

Mini-Cluster Overlay Tree

The major reason for designing mini-cluster overlay tree is to minimize the cost of searching and delivery when chunk rate increases. It directly benefits each parallel stream in an optimal condition and minimizes the problem of outliers. More video can be cached and served by distributed machines at the same time so as to multiply the users.

Lightweight Python Application Code

The benefits of python coding is to provide a more flexible base for web development because it can either be embedded in the web server like PHP or run in a separate process. Besides, python coding minimizes the redundant traffic in audio session that results in audio-to-video lagging problem. We adopt a simple tuner algorithm to illustrate this problem. Python-based idle callbacks algorithm is used to avoid some task duplications (e.g., loading tasks). When performing a long computation intensive processing of audio and video content from CDNs, the loading tasks are generated repeatedly, thereby significantly interrupting the users' interface. Thus, it may severely degrade the overall performance. It processes accumulated work load to CDNs and peers. Therefore, this simple algorithm takes the advantage of idle callback mechanism to facilitate the system keep working in a good condition. There is no blocking of user interface during the tasks.

Idle Callback Algorithm for Chunking

```
def idle_callback_mechanism(self)
   complete = fine tuner ()
   count = 0
   while incomplete && count < Ω :
   //where Ω is a size of video packet, Mbit
      complete = fine tuner ()
   if incomplete:
      return true
      cleanup()
      return false
   initialize()
Gdk.threads_add_idle(self.idle_callback_mechanism)
```

During the implementation, a tradeoff should be taken because if Ω is too large, it will trigger and interrupt the output interface for cleaning up action. However, if Ω is too small, it will not filter duplicate messages and lead to high overhead after a period of computation.

III. Simulations

Simulation Environment

Our simulation study is based on the platform of Omnet++ [21] network simulator. Specifically, the framework of OverSim [22], which is integrated with patched INET [23] framework, is adopted in our simulation scenarios. Moreover, at the application level, a P2P live streaming system is used. The physical distribution of mobile nodes is generated randomly. The behaviors of users at application level are recorded by the trace manger which can parse the scenarios and trace files containing different events. As the output results are saved as lined-oriented text files, we can conveniently use various programming languages (e.g., Python) for statistical data analysis and data graphs generation. Several major simulation scenarios are described as follows.

Scenario One

A 24-hour live streaming camera is chosen from JNN (Japan News Network) [20]. The resolution is 360 pixels wide. Users now start streaming audio and video in a P2P manner. Each node observes the network conditions periodically and takes appropriate adjustment. If the network conditions are deteriorating, the quality of the video output over P2P network will be degraded, some users will leave the network. If it is unable to receive the stream, user could only receive the nice image and audio again.

Scenario Two

Users may suddenly leave the system from P2P overlay without any announcement. Peers served by these suddenly disconnected peers have to receive the video streams from others. Then, the remaining users have to perform peer selection again. When the disconnected user re-connects to the P2P overlay, trackers will treat the user as a new peer.

Scenario Three

When a mobile user joins a P2P live video streaming overlay, trackers will start a handoff functionality to check the threshold of physical and network layer parameter and perform resource utilization enhancing schemes such as packet scheduling and quantization adaptation. These are important in tackling the problem of diverse resource constraints and user preferences while optimizing the overall utility of video. Then mobility control is performed to initiate a handoff process. The mobile user terminal periodically updates the measurement statistics of both QoE and physical network layer, such as packet dropping and renewal of its client profile.

Scenario Four

Trackers evaluate the minimum requirements for a new peer to join the overlay. The audio and video will be streamed and synchronized between group members so that the video conferencing is achieved and they can communicate with one another in real time. Group members can also either reconnect or being invited to join by a new peer. If they meet the minimum requirements evaluated at their own group, they will join the group by consuming video content through P2P network. Otherwise, they only obtain a nice image and audio.

Scenario Five

Suppose user A and user B share the same network. User A consumes most of the bandwidth. User B joins a live stream and obtains low QoE of live stream visualization. The access network reacts to the new conditions by providing the required bandwidth for P2P streaming. The streaming system accesses at the expense of the background traffic allocated for downloading. The user therefore can watch the streaming content at a higher QoE level. This action implies that the user connects to the web portal and prioritizes the live stream over the background traffic, so as to provide the required bandwidth to the P2P flows. The result is that background traffic bandwidth is shrunk until live streaming meets the QoE requirements (see Fig. 5).

In Fig. 6 (left), if Ω = 3Mbit, the packet loss of QoE is plunged into 0.46 for maximum goodput (i.e. 100% goodput) because of the occurrence of long delay, the user interface will be interrupted seriously. As this is obviously not a desirable outcome of streaming, the parameter Ω will be carefully reduced to 2 Mbit. The packet loss of QoE (=1.68) is improved for maximum goodput (see Fig. 6 (centre)). It still cannot reach the best result until Ω = 1 Mbit (see Fig. 6 (right)), where we have packet loss of QoE = 3.24.

Fig. 7 (centre) indicates that throughput contribution of QoE achieves 2.72 only if Ω = 0.8 Mbit for maximum goodput. The worst case is caused by the overhead as shown in Fig. 7 (right) because Ω = 0.4 Mbit which is too small to be discarded by the cleanup action. As a result, it may be accumulated in low bandwidth conditions. The user interface will be interrupted. The throughput contribution of QoE = 1.96 in Fig. 7 (right) illustrates the problem. Therefore, in Fig. 7 (left), we carefully fine tune the size of video packet to 1 Mbit. Finally, we have the throughput contribution of QoE = 3.28.

Moreover, Fig. 8 (centre) and Fig. 8 (right) show that when Ω = 0.8 Mbit and Ω = 0.4 Mbit, packet delivery of QoE are 2.80 and 2.00, respectively. However, if the Ω =1Mbit, the packet delivery of QoE can reach 5.00 (see Fig.8 (left)). Consequently, when the optimal video packet size is equal to 1 Mbit, the overall satisfactory QoE performance is obtained.

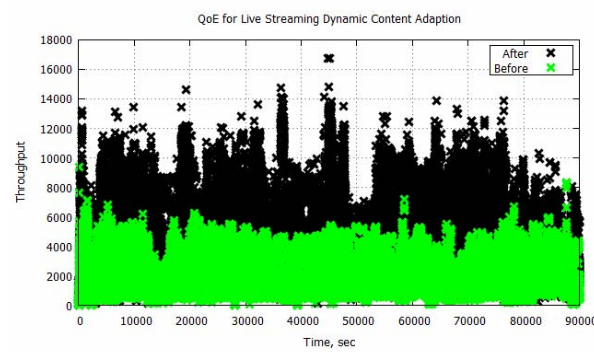

Fig. 4 QoE for live streaming dynamic content adaption

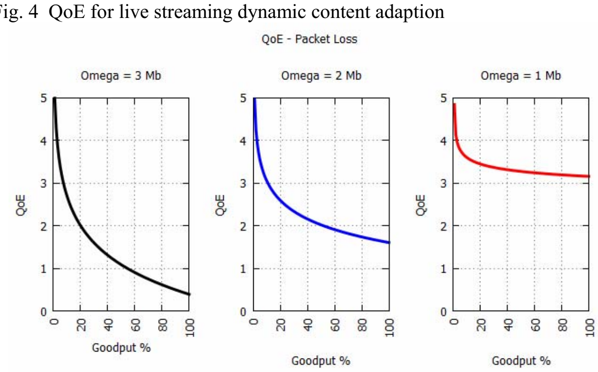

Fig. 5 Simulation Results: Packet Loss

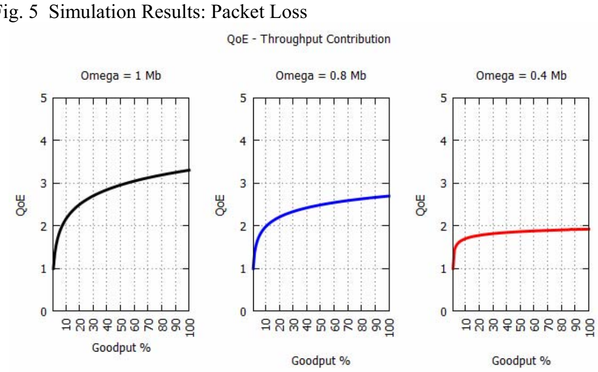

Fig. 6 Simulation Results: Throughput Contribution

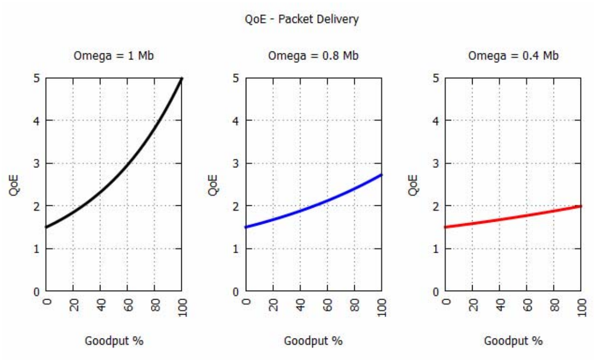

Fig. 7 Simulation Results: Packet Delivery

IV. Conclusions

In this paper, we investigated in detail the QoE performance of a MANET based P2P video streaming system. Our simulated P2P MANET streaming system is based on a contemporary video streaming architecture (e.g., YouTube). We believe that our results will be insightful to improve the performance of mobile P2P streaming in the future.

V. Acknowledgments

The authors would like to thank our research group member, Mr. Xin Jin, for his useful comments and advice on the presentation in this paper. Thanks are also due to the anonymous reviewers for their constructive comments.

References

[1] YouTube Live, 2012, http://www.youtube.com/live
[2] PPTV, 2012, http://pptv.com
[3] UUSee, 2012, http://www.uusee.com
[4] H. Yin, X. Liu, T. Zhan, V. Sear, F. Qiu, C. Lin, H. Zhang, and B. Li, "Design and Deployment of a Hybrid CDN-P2P System for Live Video Streaming: Experiences with LiveSky," *Proc. 17th ACM Int. Conf. on Multimedia*, pp. 25–34, Oct 2009.
[5] A. C. Dalal, "User-Perceived Quality Assessment of Streaming Media Using Reduced Feature Sets," *ACM J. TOIT*, vol. 11, no. 2, Dec. 2011.
[6] T. Hosfeld, M. Fiedler, and T. Zinner, "The QoE Provisioning-Delivery-Hysteresis and its Importance for Service Provisioning in the Future Internet," *7th EURO-NGI*, pp. 1–8, June 2011.
[7] S. Banerjee, C. Kommareddy, K. Kar, B. Bhattacharjee, and S. Khuller, "Construction of an Efficient Overlay Multicast Infrastructure for Real-time Applications," *Proc. IEEE 22nd Annu. Joint Conf. on Computer and Communications*, vol. 2, pp. 1521–1531, April 2003.
[8] T. Hosfeld, M. Seufert, and R. Schatz, "Quantification of YouTube QoE via Crowdsourcing," *Proc. IEEE Int. Symposium on Multimedia*, pp. 494–499, Jan. 2012.
[9] X. Cheng, C. Dale, and J. Liu, "Statistics and Social Network of YouTube Videos," *Proc. 16th Int. Workshop on QoS,* pp. 229–238, June 2008.
[10] S. Agarwal, J. P. Singh, A. Mavlankar, P. Baccichet, and B. Girod, "Performance and QoS Analysis of a Live P2P Video Multicast Session on the Internet," *Proc. 16th Int. Conf. on QoS*, pp. 11–19, June 2008.
[11] B. Cheng, L. Stein, H. Jin, X. Liao, and Z. Zhang, "GridCast Improving Peer Sharing for P2P VoD," *Proc. Trans. Multimedia Computing, Communications, and Applications*, vol.4, Art. 26, Oct. 2008.
[12] Landmark Routing Protocol, 2002, http://tools.ietf.org/html/draft-ietf- manet-lanmar-05
[13] Y. Guo, C. Liang, and Y. Liu, "Adaptive Queue-Based Chunk Scheduling for P2P Live Streaming", *Proc.7th IFIP-TC6,* pp. 433–444, May 2008.
[14] H. French, Jie Lin, Tung Phan, and A. C. Dalal, "Real Time Video QoE Analysis of RTMP Streams," *Proc. 30th IEEE IPCCC*, pp. 1–2, Nov. 2011.
[15] T. Zinner, O. Hosfeld, O. Abboud, T. Hossfeld, "Impact of Frame Rate and Resolution on Objective QoE Metrics," *QoMEX*, pp. 29–34, 2010.
[16] P. Reichl, S. Egger, R. Schatz, and A. D. Alconzo, "The Logarithmic Nature of QoE and the Role of the Weber-Fechner Law in QoE Assessment," *Proc. IEEE ICC*, pp. 1–5, May 2010.
[17] T. N. Kim, S. Jeon, and Y. Kim, "A CDN-P2P Hybrid Architecture with Content/Location Awareness for live Streaming Service Networks," *Proc. IEEE 15th Int. Symposium on Consumer Electronics*, pp. 438–441, Aug. 2011.
[18] M. Fesci-Sayit, E. T. Tunali, and A. Murat Tekalp, "Bandwidth-Aware Multiple Multicast Tree Formation for P2P Scalable Video Streaming Using Hierarchical Clusters," *Proc. 16th IEEE ICIP*, pp. 945–948, Feb. 2010.
[19] M. Fiedler and T. Hosfeld, "Quality of Experience-Related Differential Equations and Provisioning-Delivery Hysteresis," *21st ST ITC Specialist Seminar on Multimedia Applications – Traffic, Performance and QoE*, Miyazaki, Japan, March 2010.
[20] YouTube Live News from JNN, http://radioactive.eu.com/index.php?option=com_k2&view=item&id=4:live-feed&Itemid=193
[21] Omnet++, http://www.omnetpp.org/
[22] I. Baumgart, B. Heep, S. Krause, "OverSim: A Flexible Overlay Network Simulation Framework", *Proc. 10th IEEE GI in conjunction with IEEE INFOCOM 2007*, pp. 79 - 84, May 2007.
[23] Patched INET framework, http://www.oversim.org/chrome/site/INET-OverSim-20101019.tgz