Scalable Peer-to-Peer Video Streaming in WiMAX Networks

Muhammad Aman†, Biplab Sikdar† and Shyam Parekh‡
†Department of ECSE, Rensselaer Polytechnic Institute, Troy, NY 12180
‡ Bell Labs, Alcatel-Lucent, Murray Hill, NJ 07974

Abstract— **The increasing popularity and success of web-based peer-to-peer (P2P) systems for streaming video applications make them likely candidates for injecting large volumes of traffic in the emerging WiMAX networks. This paper develops a lightweight mechanism for P2P streaming in WiMAX networks that significantly reduces the load on the network and improves the scalability of the streaming system. The proposed system uses the broadcast mechanism provided in the IEEE 802.16 mechanism for providing the scalability, without breaking the P2P semantics. The scalability of the proposed system is analytically evaluated and also quantified using simulations. Our results show that the degree of improvement in the performance of the proposed system is lower bounded by the average number of peers served by an Access Service Network Gateway in the WiMAX networks.**

I. Introduction

An increasingly popular application in the Internet today is peer-to-peer (P2P) video streaming [1], [2], [3]. With the advent of many commercial grade systems that are freely available, P2P streaming has attained a large number of users. For example, the Chinese New Year celebrations were broadcast by a commercial P2P streaming systems to over 200,000 users in the Internet with bits rates in the range 400-800 Kbps [4]. When one considers realistic future scenarios with potentially millions of users watching streams of 500 Kbps or more, the Internet backbone and access networks such as those offered by WiMAX can be easily overwhelmed with traffic. In the context of WiMAX as the access network for the users of these services, in addition to the scalability provided by the P2P streaming systems, additional WiMAX specific features may be required to ensure satisfactory user experience. From the perspective of WiMAX operators, it is thus of interest to evaluate the scalability of video streaming in their networks and facilitate the integration of features in the P2P systems that exploit the flexibility and unique features of the PHY and MAC layers of WiMAX networks. Existing work on P2P streaming systems primarily focuses on wired networks [1], [2], [3] and these solutions are not well suited for WiMAX networks and neither are they capable of exploiting the features and functionalities of these networks.

This paper addresses the above issues in two ways: (1) it proposes a mechanism for P2P streaming specific to WiMAX networks that tries to maximize the service capacity of the video streaming system by exploiting the multicast services provided in IEEE 802.16 and (2) it evaluates the scalability of P2P streaming services in WiMAX networks. The proposed streaming mechanism requires only minor changes to existing P2P protocols while maintaining P2P semantics and does not require any changes to the IEEE 802.16 standards. Using both analysis and simulations, we show that the proposed scheme can improve the scalability of the P2P streaming system by a magnitude at least as large as the average number of peers served by an Access Service Network Gateway (ASN-GW).

The rest of the paper is organized as follows. In Section II we present an overview the architecture of popular P2P streaming systems. Section III presents the proposed mechanism for P2P streaming in WiMAX networks while Section IV presents the scaling model. Finally, Section V evaluates the performance of the proposed system using simulation and Section VI presents the concluding remarks.

II. Overview of P2P Streaming Systems

Existing schemes for the delivery of streaming multimedia or IPTV content include IP multicasting [5], application level infrastructure overlays [6], peer-to-peer multicast trees [7] and mesh-pull peer-to-peer streaming [1], [8]. Among these, the mesh-pull based streaming systems such as PPLive, PPStream and CoolStreaming are among the most popular. While these systems are proprietary, they follow the basic architecture of generic mesh-pull P2P streaming systems shown in Figure 1.

A typical mesh-pull P2P streaming system consists of a tracker server and individual peers that cooperatively distribute the media content among themselves. The media content is initially available at the channel streaming server which breaks the content into chunks that form the unit of information exchange among the peers. Each peer consists of a streaming engine and a media player. The streaming engine at each peer downloads media chunks from other peers and the channel streaming servers and exchanges these chunks with other peers. The chunks received at a peer are reassembled to form the original content and played back by the media player.

When a peer desires to join the network and download media, it first contacts the tracker server to obtain a list of the media available in the P2P network and obtains an initial list of peers that are currently watching the same media or channel as the peer wishes to download. The peer then contacts the peers in the list to obtain additional lists of peers watching the same media. As the peer starts to download and view the media, it maintains a buffer with few minutes worth of chunks. These chunks may include those that have already been viewed as well as those that are waiting to be played.

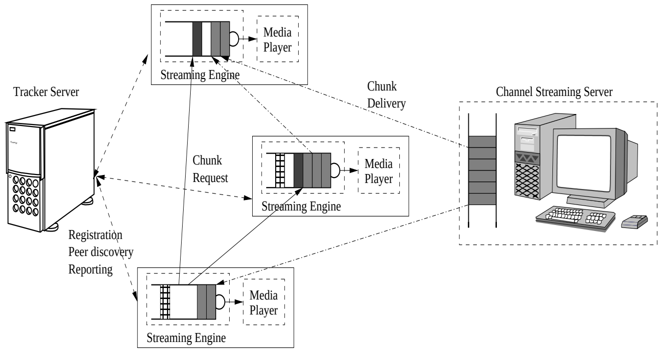

Fig. 1. Architecture of mesh-pull P2P streaming systems.

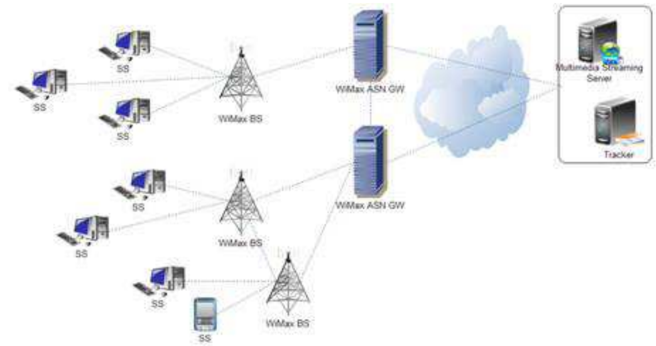

Fig. 2. Architecture of P2P streaming in WiMAX networks.

Each peer maintains a *buffer map* that indicates which chunks that it currently has buffered for sharing with others. Peers exchange buffer maps with other peers in their peer list and can request one or more chunks from them. Chunks may also be downloaded from the original channel streaming server in parallel with the downloads from other peers. While TCP is typically used for transferring the chunks, some mesh-pull P2P systems use UDP as the transport protocol in order to ensure timely delivery.

III. Proposed Architecture

Consider a scenario where a number of SSs in a WiMAX network are interested in joining a P2P streaming session. When there are multiple SSs associated with a single BS that wish to view the same channel or content, the BS has to send the same media chunks to all the peers, separately. Thus an intuitive method for reducing the load on the BS and allow the P2P network to scale would be to develop a mechanism that can exploit the inherent broadcast nature of the transmissions from the BS to avoid redundant transmissions not only in the last hop (i.e. from the BS to the SSs), but also inside the core network or the Internet. However, these need to be done without breaking the P2P semantics and without any major changes to the underlying P2P protocol. Additionally, the desired solution should be lightweight, scalable and should be implemented without any changes to the WiMAX entities. This section highlights the challenges and the proposed solution for a scalable P2P streaming system for WiMAX networks.

A. Overview

Consider Figure 2 which shows the basic architecture of a P2P streaming system in WiMAX networks. If any of the SSs wants to receive the P2P stream, it first contacts the tracker in the same way as any other P2P node. However, the goal of our proposed protocol is that if now another SS wants to receive the same stream, it should be able do so without any additional bandwidth requirement. To solve this problem, we propose that the ASN-GW be the proxy for a multicast session associated with its SSs that are involved in the P2P streaming of a particular channel. Thus, after one SS starts receiving the stream, adding another SS to the same stream would require only adding the next SS to the multicast session. However, to do so, the SS should be able to identify and join the target multicast session. A BS in WiMAX networks cannot distinguish an Layer 3 (L3) multicast group. Consequently when an AS-GW receives a multicast packet, it has to make a copy of the packet for each of the SSs in the multicast group, and delivers the copies of the same packets to the BS. To eliminate this redundancy, we use a technique similar to that in [9] where the BS needs to send the multicast IP address along with the Multicast Polling Assignment Request (MCA-REQ) messages, so that the SS can identify the L3 multicast group of the MCA-REQ message and join the correct group.

In the proposed scheme, in addition to the usual attributes about the peers, the tracker includes two additional fields for each peer. The first field consists of one bit to indicate whether the peer is in a WiMAX network or not. For peers that are WiMAX SSs, the second field then stores the multicast IP address associated with the SS in its ASN-GW. To populate these entries, peers provide the necessary information either during registration, or later on acquiring a multicast IP address. When the tracker responds to new peers with the initial list of peers, it preferentially lists existing peers that are WiMAX SSs. This increases the chances that the new WiMAX peer will be able to find other peers that are SSs in its own ASN-GW, if any. After receiving the addresses of the peers and the multicast IP addresses of WiMAX peers, the new peer now listens to the periodic MCA-REQ messages sent by its BS to check if any of the ongoing multicast sessions matches the ones listed by the tracker. If there is a match, the peer joins this multicast session to receive the multimedia stream. We now describe the proposed mechanism in detail by considering various scenarios of operation.

B. Case 1: No Existing WiMAX Peers in Same ASN-GW

We first consider the case when there are no existing WiMAX peers in the ASN-GW of the new peer (SS) that wishes to join the streaming session. In this scenario, as shown in Figure 3, a new peer starts by contacting the tracker for a list of available peers, with a preference for peers in WiMAX networks. The tracker replies with a list of available peers with preference to peers in WiMAX networks and their multicast IP addresses. On receiving this information, the peer then compares the listed multicast IP addresses, if any, with the multicast IP addresses in the periodic MCA-REQ messages sent by its BS. In this case, there will not be any matches. To

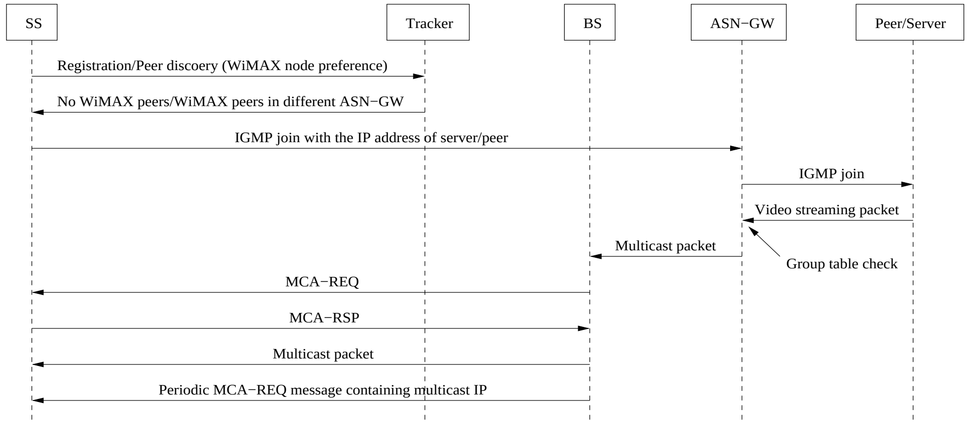

Fig. 3. Protocol operation in the absence of peers in the same ASN-GW.

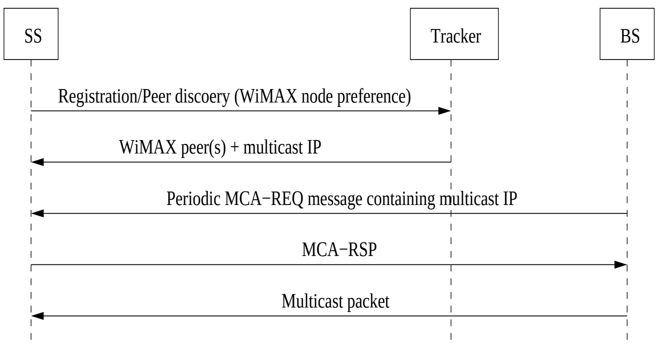

Fig. 4. Protocol operation in the presence of peers in the same BS and ASN-GW.

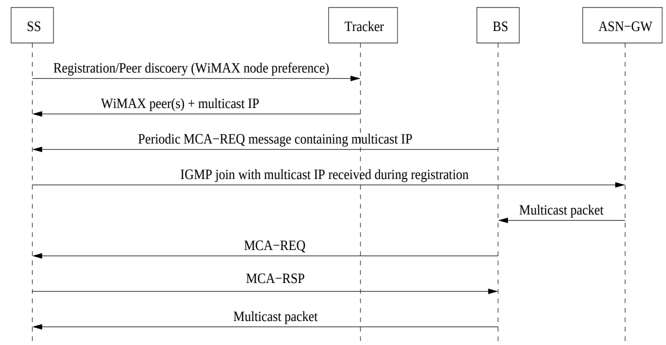

Fig. 5. Protocol operation in the presence of peers in the same ASN-GW but different BS.

check if any of the multicast sessions in the received list exist in another BS served by its ASN-GW (see case 3 below), the peer then sends IGMP join requests to its ASN-GW for these multicast IP addresses. The ASN-GW checks its group table and in the current case does not find any match, and drops these requests. Thus the peer concludes that there there is no active multicast session for the requested P2P media stream within its ASN-GW. Thus the new peer initiates a new P2P session with the server or one of the peers outside its ASN-GW by sending it an IGMP join message. Similar messages may be sent to multiple peers. Moreover, the new peer will request the server and the other peers to send the media stream to its destination multicast address instead of a unicast address. Once the server or the peers join the multicast session, all streaming packets to the new peer are sent over the multicast session. At the last hop, the streaming packets are sent to the peer SSs using MCA-REQ and MCA-RSP exchanges. Finally, the new peer updates the tracker with the multicast IP address of the sessions that it has initiated.

C. Case 2: Existing WiMAX Peer in Same BS and ASN-GW

The second scenario deals with the case when a peer exists in the same BS and ASN-GW as the SS which wants to join the P2P stream. As in the previous case, when the new peer contacts the tracker for P2P nodes, it will notify the tracker that it is a WiMAX node, as shown in Figure 4. In turn the tracker will provide the new peer with a list of peers in WiMAX networks and other information such as the associated multicast IP addresses. The peer then listens to the MCA-REQ messaged broadcast by its BS and compares the multicast IP addresses in the list with those in the MCA-REQ messages. In this case there is a match and the peer then joins the multicast session. Subsequently, it can listen on to the streaming multimedia packets that are broadcast by the BS.

D. Case 3: Existing WiMAX Peer in Same ASN-GW but Another BS

The third scenario deals with the case when no peer exists in the same BS as the new peer but there exists a peer in the region of a BS being served by the same ASN-GW as the BS of the joining SS. In this scenario, as shown in Figure 5, the new peer will perform peer discovery as in the previous two cases. As usual, the tracker replies with a list of existing peers, with a preference to peers in WiMAX networks. However, when the SS compares the multicast IP addresses obtained from the tracker with the multicast IP addresses received in the periodic MCA-REQ messages from its BS, the comparison fails. Thus the peer SS sends IGMP join messages to the ASN-GW with the same multicast IP addresses as it received from the tracker. The ASN-GW checks it's group table and finds out that a multicast session with one of the multicast IP addresses is already in progress and thus adds the new BS into the multicast group. The join messages for IP addresses not in its group table are dropped. Next time when the BS receives the multicast packet from the ASN-GW, it will send the MCA-REQ message with multicast IP address of the P2P stream. Thus, the new peer accepts the MCA-REQ message by sending an MCA-RSP message and starts receiving the multimedia stream.

E. Effect on the P2P Streaming Protocol

The proposed protocol can scale a P2P multimedia stream to a large number of WiMAX users with very minor changes to the existing P2P protocols. Firstly, each peer adds a bit in the registration message it sends to the tracker to indicate that it is in a WiMAX network. Secondly, the tracker now needs to save the multicast IP as well as the unicast IP address of the peers in the case of nodes in WiMAX networks. The third change that is required is that if a non-WiMAX P2P node receives an IGMP join message, it should start sending the chunks to a multicast IP address rather than to an unicast IP address. The upload portion of the P2P protocols remains the same, i.e. if a non-WiMAX node requests a chunk from a WiMAX node, the procedure followed will be similar to existing P2P protocols.

IV. Scaling Behavior of the Proposed Protocol

In a streaming session, the usability of any chunk arriving at a peer depends on the time that has elapsed since the chunk was first generated by the streaming server (source). Thus the number of peers that may be supported in a P2P streaming session depends on the delay bound as well as the overlay P2P topology. In this section we evaluate a bound on the number of peers that may be supported by the proposed P2P system and compare it with that of a native P2P system.

We start with a characterization of an optimal P2P streaming session. The optimal session is defined as one that achieves the largest number of peers receiving data at a given rate r within a maximum tolerated delay, without considering fairness among peers. Now, the delay incurred by a chunk is a sum of the queueing delays and the propagation delays. Queueing delays are incurred when chunks are enqueued at a peer waiting to be played or to be relayed to other peers. It is obvious that there is no queueing delay in an optimal session. This is because if all peers receive data at rate r, then buffering is not required and a peer may immediately play a block it receives. Consequently, we need to consider only propagation delays.

Let d_i be the propagation delay from the source to peer i, measured in terms of the number of P2P overlay hops. A peer is considered served if the upload capacity of the source and all the existing served peers have the capacity to serve it at rate r, while ensuring that the downloads of none of the existing peers is disrupted. We consider a network with N peers and without loss of generality we assume that the N peers are served in the order $1, 2, \cdots, N$. Then the propagation delay for peer i is $k+1$ (i.e. $d_i = k + 1$) if k is the smallest integer for which

$$c_s + \sum_{j=1}^{i-1} u_j \mathcal{I}(d_j \leq k) \geq ir \tag{1}$$

where c_s and u_j are the upload bandwidths of the source and peer j respectively, and $\mathcal{I}(x)$ is an indicator function that takes the value of 1 if x is true and 0 otherwise. The optimal streaming session tries to maximize the number of peers served while ensuring that the maximum delay at any peer is not more than a threshold Δ. Intuitively, placing peers with large upload capacities closer to the source will increase the number of peers that can successfully join the P2P streaming session. This result can be formalized as:

Claim 1: Consider a P2P streaming session with N peers where the server's bandwidth is c_s and the other peers have a bandwidth drawn from a distribution $F(u)$. By placing the peers with higher upload capacities closer to the source, the number of peers supported with delays lower than a threshold Δ is no less than that of an optimal session.

Proof: Assume that there exists an optimal session Π in which there exists at least one set of peers (a_r, a_v) such that the upload capacity $u_r < u_v$ but $d_r > d_v$. The number of peers supported by the optimal P2P session is then

$$\begin{aligned}\Omega(\Pi) &= \sum_{i=1}^{N} \mathcal{I}(d_i < \Delta) \\ &= \sum_{i=1}^{r-1} \mathcal{I}(d_i < \Delta) + \mathcal{I}(d_r < \Delta) + \sum_{i=r+1}^{v-1} \mathcal{I}(d_i < \Delta) \\ &\quad + \mathcal{I}(d_v < \Delta) + \sum_{i=v+1}^{N} \mathcal{I}(d_i < \Delta)\end{aligned} \tag{2}$$

Now consider an alternate topology Π' which is identical to Π in all respects except that the positions of a_r and a_v are swapped. Let the distance for peer i in this topology be d'_i. The number of peers supported by Π' is then

$$\begin{aligned}\Omega(\Pi') &= \sum_{i=1}^{r-1} \mathcal{I}(d'_i < \Delta) + \mathcal{I}(d'_r < \Delta) + \sum_{i=r+1}^{v-1} \mathcal{I}(d'_i < \Delta) \\ &\quad + \mathcal{I}(d'_v < \Delta) + \sum_{i=v+1}^{N} \mathcal{I}(d'_i < \Delta)\end{aligned} \tag{3}$$

Since a peer with higher upload capacity has been placed in position r, the new distance $d'_r = d_r$. Also, the peers in positions 1 to r–1, r+1 to v–1, and v+1 to N are not changed. Thus we have $d'_i = d_i$ for $1 \leq i \leq r - 1$. The difference in the number of peers supported in the two sessions is then

$$\begin{aligned}\Omega(\Pi) - \Omega(\Pi') &= \sum_{i=r+1}^{v-1} \mathcal{I}(d_i < \Delta) + \mathcal{I}(d_v < \Delta) + \sum_{i=v+1}^{N} \mathcal{I}(d_i < \Delta) \\ &\quad - \sum_{i=r+1}^{v-1} \mathcal{I}(d'_i < \Delta) - \mathcal{I}(d'_v < \Delta) - \sum_{i=v+1}^{N} \mathcal{I}(d'_i < \Delta)\end{aligned} \tag{4}$$

Now $d_i = k + 1$ if k is the smallest value for which $c_s + \sum_{j=1}^{i-1} u_j \mathcal{I}(d_j \leq k) \geq ir$, i.e.

$$d_i = 1 + \arg\min_k \left\{ \sum_{j=1}^{i-1} u_j \mathcal{I}(d_j \leq k) + c_s \geq ir \right\} \tag{5}$$

Since $u'_r > u_r$ and $u'_j > u_j$, $\forall j$ with $j \neq v$, we have $\sum_{i=1}^{j} u'_i > \sum_{i=1}^{j} u_i$ and $d'_i \leq d_i$ for $r < j < v$. This in turn implies

$$\mathcal{I}(d'_j < \Delta) \geq \mathcal{I}(d_j < \Delta) \tag{6}$$

for $r < j < s$. Now since the nodes in positions r and v of Π are swapped in Π', we have $u'_r + u'_v = u_r + u_v$. This implies $\sum_{i=1}^{j} u'_i = \sum_{i=1}^{j} u_i$ for $j > v$. As before, we again have

$$\mathcal{I}(d'_j < \Delta) \geq \mathcal{I}(d_j < \Delta) \tag{7}$$

for $j > s$. From Eqns. (6) and (7) we then have

$$\Omega(\Pi') - \Omega(\Pi) \geq 0 \tag{8}$$

This shows that the session under Π' is no worse than the optimal session. ■

From the result above, we can also conclude that in an optimal session, the server never uploads media to peers that are more than one hop away. Next we consider a means to obtain a lower bound on the number of supported peers by any P2P streaming system. For this, we consider the case where all peers have the same, constant upload capacity. We show that such a system serves as a lower bound on the number of peers that can be supported by any P2P streaming system.

Claim 2: Consider an arbitrary upload capacity distribution $F_a(u)$ that has the same mean as a constant upload capacity distribution $F_c(u) = \delta(\bar{u})$. As the network size increases, the number of peers that can be supported with delays less than Δ when the peers have upload capacities drawn from $F_a(u)$ is no less than the corresponding number of peers supported when the upload capacities are constant.

Proof: From Claim 1, to maximize the number of supported peers, we place the peers with the higher upload capacities closer to the source. Also, the peer upload capacities, in decreasing order, in the arbitrary distribution are denoted by $a_1, a_2, \cdots, a_N$ and in the constant capacity case by $b_1, b_2, \cdots, b_N$ with $b_i = b_j$, $\forall i, j$. Since the means of the two distributions is the same, we have

$$\lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} a_i = \frac{1}{N} \sum_{i=1}^{N} b_i \qquad \text{w.p. } 1 \tag{9}$$

where w.p. stands for "with probability". The probability that there exists at least one peer (i.e. the peer with the highest upload capacity) with upload capacity greater than $\bar{u}$ is

$$P[a_1 > \bar{u}] \;=\; 1 - \prod_{i=1}^{N} P[a_i \leq \bar{u}] \;=\; 1 - [P[a \leq \bar{u}]]^N$$

As $N \to \infty$, we have

$$P[a_1 > \bar{u}] \;=\; \lim_{N\to\infty} 1 - [P[a \leq \bar{u}]]^N \;=\; 1 \tag{10}$$

Also since from Eqn. (9) we have $\sum_{i=1}^{N} a_i = \sum_{i=1}^{N} b_i$, Eqn. (10) implies that $\sum_{i=1}^{j} a_i > \sum_{i=1}^{j} b_i$, for $1 \leq j < N$, and thus $d_{a_j} \leq d_{b_j}$. This in turn implies

$$\mathcal{I}(d_{a_i} < \Delta) \geq \mathcal{I}(d_{b_i} < \Delta) \tag{11}$$

for $1 \leq j < N$ and hence

$$\sum_{i=1}^{N} \mathcal{I}(d_{a_i} < \Delta) \geq \sum_{i=1}^{N} \mathcal{I}(d_{b_i} < \Delta) \tag{12}$$

which concludes the proof. ■

Next we proceed to obtain the number of peers that may be supported for a given delay bound in the native as well as the proposed P2P streaming system using the methodology in [10]. Here we assume that all peers are in WiMAX networks. Also, the number of peers under each ASN-GW is denoted by M. Note that from Claims 1 and 2, to compare the worst case performance of the two systems, it suffices to compare their performance under the case of constant upload capacity distribution. We consider the native P2P system first and assume that c_s/r and $\bar{u}/r$ are integers. Based on the ratio of $\bar{u}$ and r, we have two cases and the following results summarize the performance in each case.

Claim 3: The number of peers supported in the native P2P streaming system for a delay constraint Δ when $\bar{u} \geq r$ is

$$N_{max} = \frac{c_s}{r} \frac{(\frac{\bar{u}}{r})^{\Delta} - 1}{\frac{\bar{u}}{r} - 1} \tag{13}$$

Proof: The streaming server serves c_s/r peers and each of these in turn can serve $\bar{u}/r$ peers. With a delay constraint of Δ, we can have $\Delta - 1$ such hops. The total number of peers supported is then

$$N_{max} \;=\; \sum_{i=0}^{\Delta-1} \frac{c_s}{r} \left(\frac{\bar{u}}{r}\right)^i \;=\; \frac{c_s}{r} \frac{(\frac{\bar{u}}{r})^{\Delta} - 1}{\frac{\bar{u}}{r} - 1} \tag{14}$$

which concludes the proof. ■

When $\bar{u} < r$, the addition of a peer introduces less upload bandwidth in the network than that used by the peer. Thus if Δ is large, there will not be sufficient bandwidth to support any more peers after a number of hops, h. In particular, h is given by the smallest i such that $\frac{c_s}{r}(\frac{\bar{u}}{r})^i - 1 < r$, i.e.

$$i > \log_{(\frac{\bar{u}}{r})} \left(\frac{r^2}{c_s}\right) + 1 \tag{15}$$

since $\log_\alpha(x) > \log_\alpha(y)$ if $\alpha < 1$ and $x < y$. The number of peers supported in this case is given by the following result.

Claim 4: The number of peers supported in the native P2P streaming system for a delay constraint Δ when $\bar{u} < r$ is

$$N_{max} \begin{cases} < \frac{(\frac{\bar{u}^3}{r^2}) - \frac{c_s}{r}}{\frac{\bar{u}}{r} - 1} & \text{for } h = \log_{(\frac{\bar{u}}{r})} \left(\frac{r^2}{c_s}\right) + 1 < \Delta \\ = \frac{c_s}{r} \frac{(\frac{\bar{u}}{r})^{\Delta} - 1}{\frac{\bar{u}}{r} - 1} & \text{otherwise} \end{cases} \tag{16}$$

Proof: In the first case, the bandwidth is exhausted before the delay bound is met. The number of supported peers is then

$$N_{max} \;=\; \sum_{i=0}^{\left\lceil \log_{(\frac{\bar{u}}{r})} \left(\frac{r^2}{c_s}\right) \right\rceil} \frac{c_s}{r} \left(\frac{\bar{u}}{r}\right)^i \;<\; \frac{(\frac{\bar{u}^3}{r^2}) - \frac{c_s}{r}}{\frac{\bar{u}}{r} - 1} \tag{17}$$

In the case where the delay bound is reached before the available bandwidth is exhausted, we can use Eqn. (14) to derive the number of supported peers. This completes the proof. ■

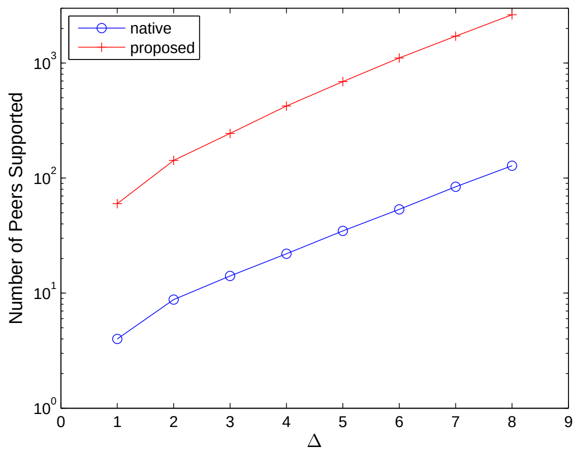

Fig. 6. Scalability of the proposed system when $\bar{u} > r$.

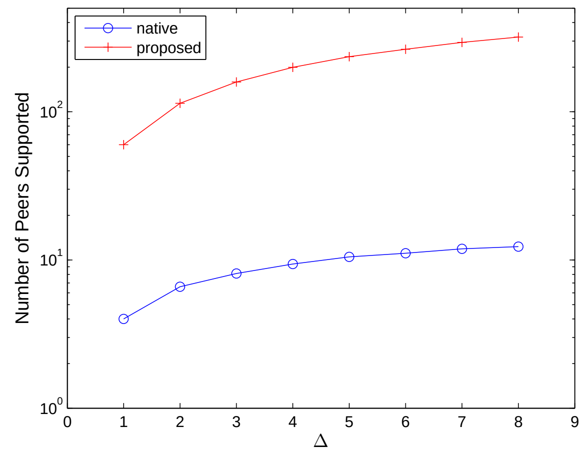

Fig. 7. Scalability of the proposed system when $\bar{u} < r$.

For the proposed P2P streaming system, we can use the same results as above by interpreting N_{max} in the equations above to represent ASN-GWs rather than peers. Then since each ASN-GW can support M peers while using the same download capacity required by one peer in the native system, the proposed system is at least M times as scalable. Further improvements may occur when we consider the fact that each of the M peers in the ASN-GW may now contribute upload bandwidth.

V. Simulation Results

In this section we use simulations to demonstrate and quantify the improvement in scalability of the proposed P2P streaming system for WiMAX networks. The simulations were conducted using a custom built simulator. In the scenario considered, the network had a total of 100 ASN-GWs with three BSs in each ASN-GW. Also, each BS had five SSs that try to participate in the P2P streaming session. The required download rate at each peer is $r = 340$ Kbps and the upload capacity at each peer depends on its distance from its BS and the corresponding highest modulation and coding scheme that can be supported. The possible modulation and coding schemes were BPSK(1/2), QPSK(1/2), QPSK(3/4), 16 QAM(1/2), 16 QAM(3/4), 64 QAM(2/3) and 64 QAM(3/4). For each simulation setting, the results presented are the average over 10 runs of the simulator.

In the first scenario, the nodes are randomly placed such that they are more likely to be closer to the BS and the average upload capacity of the peers was 501.5 Kbps, i.e. $\bar{u} > r$. For this case, Figure 6 compares the number of peers supported by the proposed P2P streaming system with that of the original, native system. We see that the improvement in the proposed system provides is of the order of the number of peers in an ASN-GW, as shown in the previous section. The degree of improvement increases as the delay bound increases. Figure 7 shows the corresponding results for the case where the average upload capacity of the peers was 323 Kbps, i.e. $\bar{u} < r$. In this case, the number of peers supported saturates much faster in the native P2P streaming system. Also, the improvement in the proposed system increases as the delay bound increases.

VI. Conclusion

In this paper we proposed a mechanism for improving the scalability of P2P streaming protocols in WiMAX networks. The proposed protocol is based on exploiting the multicast feature of WiMAX networks to avoid redundant, multiple transmissions of the same packet from the BS to multiple peer SSs. A protocol for facilitating the multicast transmissions is proposed and the scalability of the proposed system is evaluated. Our results show that the proposed system can significantly improve the scalability of P2P streaming systems when the peers are part of a WiMAX network.

References

[1] X. Zhang, J. Liu, B. Li and T.-S. Yum, "DONet/CoolStreaming: A Data-driven Overlay Network for Live Media Streaming," *Proc. IEEE INFOCOM,* March 2005.

[2] M. Zhang, Q. Zhang, L. Sun and S. Yang, "Understanding the power of pull-based streaming protocol: Can we do better?," *IEEE Jour. on Sel. Areas in Comm.*, vol. 25, no. 9, pp. 1678-1694, December 2007.

[3] F. Pianese, D. Perino, J. Keller and E. Biersack, "Pulse: an adaptive, incentive-based, unstructured p2p live streaming system," *IEEE Transaction on Multimedia,* vol. 9, no. 8, pp. 1645-1660, December 2007.

[4] X. Hei, C. Liang, J. Liang, Y. Liu and K. Ross, "A Measurement Study of a Large-Scale P2P IPTV System," *IEEE Transactions on Multimedia,* vol. 9, no. 8, pp. 1672-1687, December 2007.

[5] S. Deering and D. Cheriton, "Multicast routing in datagram internetworks and extended LANs," *ACM Transactions on Computer Systems,* vol. 8, no. 2, pp. 85-110, May 1990.

[6] L. Kontothanassis et. al., "A Transport Layer for Live Streaming in a Content Delivery Network," *Proceedings of the IEEE,* vol. 92, no. 9, pp. 1408-1419, 2004.

[7] Y. Chu, S. Rao and H. Zhang, "A Case For End System Multicast," *Proc. ACM SIGMETRICS,* pp. 1-12, Santa Clara, CA, June 2000.

[8] X. Hei, Y. Liu and K. Ross, "IPTV over P2P Streaming Networks: the Mesh-pull Approach," *IEEE Communications Magazine,* vol. 46, no. 2, pp. 86-92, Feb. 2008.

[9] K. Lee, S. Rhee and H. Youn, "A MAC Layer Multicasting Approach for WiMAX Access Networks," *Proc. IEEE PerCom,* March 2008.

[10] T. Small, B. Liang and B. Li "Fundamental Scaling Laws of Peer-to-Peer Live Multimedia Streaming," *Proc. ACM Multimedia,* October 2006.