Characterizing Locality-aware P2P Streaming

Jian Zhao, Chuan Wu
Department of Computer Science
The University of Hong Kong
Hong Kong, S. A. R.
Email: {jzhao, cwu}@cs.hku.hk

Abstract— **Peer-to-peer (P2P) live streaming systems have been increasingly popular and successful in today's Internet, which provide large collections of video channels to millions of users at low server costs. The large volumes of P2P streaming traffic are exceeding those incurred by BitTorrent-like file sharing applications, threatening huge traffic relay cost to the Internet service providers (ISPs). There have recently emerged proposals advocating locality-aware P2P streaming protocol design, which aim to constrain streaming traffic within ISP boundaries and to alleviate traffic relay cost to the Internet Service Providers (ISPs). Nevertheless, there is a lack of in-depth understanding on the impact of such a locality-aware design on P2P streaming performance. Taking an analytical approach, we model the relation between streaming performance and traffic locality in P2P live streaming systems, in order to acquire useful insights for designing high-performance locality-aware real-world systems. We use end-to-end streaming delays as the performance metric for live streaming, and the number of copies of the live streams imported into an ISP to evaluate the volume of inter-ISP traffic. Considering multiple ISPs at different bandwidth levels, we characterize the generic relationship between the volume of inter-ISP traffic and the streaming performance; we then analyze the traffic volume when the best streaming performance is achieved and the streaming performance when minimum inter-ISP traffic is incurred. Our models and analyses provide intriguing insights on the design of effective locality-aware peer selection protocols and server deployment strategies across multiple ISPs. We also evaluate our models and theoretical results with large-scale simulations under realistic settings.**

Index Terms— **P2P Live Streaming, Traffic Localization, Performance Modeling, End-to-End Delay**

I. INTRODUCTION

Peer-to-peer (P2P) live streaming applications have thrived in today's Internet, (*e.g.*, PPLive [1], SopCast [2], Zattoo [3]), bringing thousands of live channels to millions of users at low server cost. As compared to P2P file sharing, the mutual exchanges of streams among peers may incur persistent and more intense inter-ISP traffic at all times, due to the delivery of live content in real time. Therefore, P2P streaming traffic has increasingly become a major source incurring traffic relay cost to the ISPs [4], risking ISPs' packet filtering and rate throttling in the near future [5].

To prevent the fate of traffic filtering, a number of locality-aware P2P streaming designs have been proposed, which connect peers to nearby neighbors in the same AS or ISP, in order to reduce inter-ISP traffic [6], [7], [8]. P4P [6] advocates collaboration between P2P applications and ISPs, where ISPs provide necessary network information for a P2P streaming application to make localized peer selection decisions. Picconi *et al.* [7] propose a two-tier mesh overlay structure, where a highly clustered primary overlay is constructed for local stream propagation and the secondary inter-cluster links are used when necessary for global stream propagation, to minimize inter-ISP traffic. The recent work of Magharei *et al.* [8] bears some similarity, where an inter-ISP scheduling algorithm over an ISP-level overlay ensures that each ISP receives all substreams of a video, and an intra-ISP scheduling scheme further delivers all substreams to all internal peers. Though locality-aware protocol designs are present, an in-depth understanding on the relationship between traffic localization and P2P streaming performance is still lacking: Will streaming performance be affected when inter-ISP traffic is cut? If so, what is the quantitative relationship between inter-ISP traffic volume and P2P streaming performance? The answers to these questions are critical for P2P streaming protocol design to achieve optimal operations on traffic localization and streaming QoS provisioning, which we seek to address with extensive theoretical analysis in this paper.

A number of work have indeed been done on theoretical analysis of P2P live streaming applications [9], [10], [11], [12], [13], [14], [15], [16]. Kumar *et al.* [9] and Liu [10] have studied the maximum sustainable streaming rate and the minimum delay bound of mesh-pull based P2P streaming protocols. Liu *et al.* [11] and Chen *et al.* [13] investigate the performance bounds for minimum server load, maximum streaming rate, and minimum tree depth under different peer selection constraints for tree-push based streaming protocols. Different chunk selection strategies are analyzed by Zhou *et al.* [14] on their impact on startup latency and streaming continuity in mesh-pull based streaming systems. Bonald *et al.* [16] compare several tree-push based streaming protocols in terms of the optimality of achieved streaming rates and delays. However, none of the above work has considered locality-aware streaming protocols.

In this paper, we analytically explore the relationship between streaming performance and inter-ISP traffic in mesh-based P2P live streaming systems, in order to derive useful insights for designing high-performance locality-aware P2P streaming systems. We use end-to-end streaming delays as the performance metric for live streaming, and quantify the amount of inter-ISP traffic with the number of copies of the live streams imported into each ISP. Considering multiple ISPs at different bandwidth levels, we characterize the generic relationship between the volume of inter-ISP traffic and the streaming performance, as well as analyze the traffic volume when the best streaming performance is achieved and the streaming performance when minimum inter-ISP traffic is incurred. Our models and analyses provide useful insights on the design of effective locality-aware peer selection protocols and server deployment strategies across multiple ISPs, which achieve desired goals on inter-ISP traffic minimization or streaming performance optimization. We also evaluate the effectiveness of our models with large-scale empirical studies under realistic settings.

The remainder of the paper is organized as follows. We present our P2P streaming system model in Sec. II and characterize the relationship between traffic localization and streaming performance in Sec. III. We perform empirical study in Sec. IV, further discuss related work in Sec. V, and conclude the paper in Sec. VI.

II. SYSTEM MODEL

We consider a locality-aware mesh-based P2P live streaming system, where each peer retrieves the live stream of a video channel by exchanging available chunks of the stream with its neighbors.[1] There is one tracker server, which has full information of online peers and assigns neighbors to each peer. A live stream consists of consecutive chunks of the live video, and each chunk corresponds to one unit time of playback. The streaming rate of the live stream is R, equaling its playback bitrate. There are a total number of N peers distributed in M ISPs in the system, with N_i peers in ISP i. Let u_{pi} denote the average upload bandwidth of peers in ISP i, $i = 1, \ldots, M$, given as multiples of the streaming rate R, *i.e.*, $u_{pi}R$ is the average peer upload capacity in bps. The total bandwidth of the streaming server (referred to as *server capacity* hereinafter) deployed in the system is u_s, given as multiples of the streaming rate R as well. Let u_{si} be the server capacity deployed in ISP i, with $u_{si} \geq 0, i = 1, \ldots, M$, and $\sum_{i=1}^{M} u_{si} = u_s$.

We make the following assumptions on the streaming system: The upload bandwidth at peers constitutes the bandwidth bottleneck, while the download bandwidth of each peer is not. Peers in ISP i $(i = 1, \ldots, M)$ have an upload bandwidth level around the average u_{pi},[2] and the average peer upload bandwidth among the ISPs satisfy $u_{p1} \geq u_{p2} \geq \ldots \geq u_{pM}$. A peer in ISP i has C_i active neighbors, and the larger peer upload capacity is in an ISP, the more neighbors each peer in the ISP can have. Specifically, we assume the average link delay on intra-ISP links in all the ISPs is the same $t_h = \frac{C_i - 1}{u_{pi}}$ $(i = 1, \ldots, M)$. Let $\tau_{j,i}$ denote the delay on inter-ISP links from ISP j to ISP i, with $\tau_{j,i} \geq t_h$. Servers deployed in ISP i directly serve at most u_{si} peers at any given time, *i.e.*, a peer that directly streams from a server can download at the rate of R. All servers distributed in different ISPs have the entire live stream, and they pump out the same chunk at the same time to peers directly connected to them.

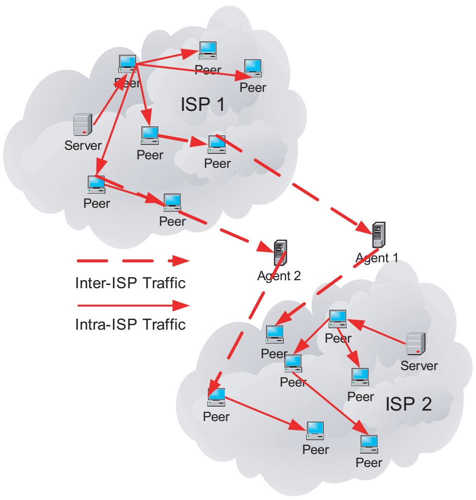

Figure 1. Agent model to characterize inter-ISP traffic.

Let $K_{j,i}$ denote the number of copies of live streams that peers in ISP i retrieve from ISP j. For better characterization of the inter-ISP traffic, we consider there are $K_{j,i}$ *virtual agents*, each downloading a copy of the live stream from ISP j and forwarding the chunks in the stream to peers in ISP i. We note that the agents are imaginary, modeled to characterize the number of live streams disseminated across ISP boundaries, and the chunks of a live stream are indeed downloaded by (possibly) different peers in ISP i from ISP j. An illustration of the agent model is given in Fig. 1. With this agent model, we introduce the simplification that an integer number of streams are downloaded from one ISP to another, while the derived insights can be readily applied to the general case with fractional streams.

The amount of inter-ISP traffic from ISP j to ISP i $(i \neq j)$ is $K_{j,i}R$. In a real-world streaming system, whether a packet incurs intra- or inter-ISP traffic can be judged based on the source IP address carried in the packet. There are existing tools to tell the ISP belonging of IP addresses, *e.g.*, ASFinder in the CoralReef suite [17].

In live streaming systems, peers need to play chunks of the live stream at similar paces. When a new chunk is pumped out from the server, it should be distributed to all peers as soon as possible. This deadline driven dissemination is different from the rarest first strategy in file sharing systems. Given the time sensitivity of live streaming, we evaluate the streaming performance in the system as the end-to-end chunk dissemination delay, *i.e.*, the time slipped from when the server pumps out a chunk to the time when all peers have received this chunk.

We summarize important notations in Table I for ease

---

[1] We focus on one video channel in our analysis and our analytical results can be readily extended to a system with multiple channels, since each peer typically only participates in one live streaming channel.

[2] We note that this is a reasonable simplification of real-world ISPs, as an ISP typically provides the same type of access network.

TABLE I.
IMPORTANT NOTATIONS

M	number of ISPs
N	total number of peers in the system
N_i	number of peers in ISP i
R	streaming bitrate
C_i	number of active neighbors at a peer in ISP i.
u_s	overall actual server capacity
u_{si}	actual server capacity deployed in ISP i
u_{pi}	average peer upload bandwidth in ISP i
$\mathscr{U}_s$	total effective server capacity
$\mathscr{U}_{si}$	effective server capacity in ISP i
t_h	delay on intra-ISP links
$\tau_{j,i}$	delay on inter-ISP links from ISP j to ISP i
$h_{j,i}^{(l)}$	hop count at which agent l pulls a chunk from ISP j to ISP i
$\tilde{u}_{j,i}^{l}$	effective server capacity brought by agent l into ISP i from ISP j
$\bar{u}_{i,j}^{l'}$	effective server capacity branched off from ISP i to ISP j by agent l'
$K_{j,i}$	number of agents downloading streams from ISP j to ISP i
$\mathscr{H}_{j,i}$	the set of hops at which agents pull chunks from ISP j to ISP i
H_i	min. number of dissemination hops for all peers in ISP i to receive a chunk
D_i	minimum end-to-end delay for all peers in ISP i to receive a chunk

of reference.

III. CHARACTERIZING THE IMPACT: TRAFFIC LOCALIZATION VS. STREAMING PERFORMANCE

We start by giving a lemma that provides theoretical bounds for minimum-delay chunk dissemination inside one ISP, to be used in our analysis.

Lemma 1. *Consider a P2P live streaming system with N peers over an ISP, where the server capacity is u_s, upload bandwidth of each peer is u_p, and the maximum number of neighbors each peer has is C. The minimum number of dissemination hops needed for all peers to receive each chunk is*

$$H = 1 + \lceil log_C \frac{N}{u_s} \rceil.$$

The minimum end-to-end delay for all peers to receive the chunk is:

$$D = 1 + \frac{C-1}{u_p} \cdot \lceil log_C \frac{N}{u_s} \rceil. \quad (1)$$

Proof: For mesh-based P2P streaming system, Liu [10] has shown that minimum end-to-end delay is achieved when each chunk is disseminated using a snow-ball approach, which can be applied to our case as follows: In the first hop, the server sends the chunk to u_s peers; in the following hop h ($h > 1$), each of the peers who have received the chunk during the previous $h-1$ hops further distributes the chunk to $C-1$ neighbors (excluding the one she has received the chunk from), who do not own the chunk yet (while the server continues distributing new chunks to u_s peers during this process).

According to this approach, the total number of peers that the chunk reaches after the 1st hop is u_s, that after 2 hops is $u_s + u_s(C-1) = u_sC$, that after 3 hops is $u_sC + u_sC(C-1) = u_sC^2$. The total number after h hops can be derived as u_sC^{h-1}. Considering the total number of peers is N, we derive the total number of peers that have received the chunk after h hops ($h \geq 1$), counting from the time the server pumps out the chunk, is

$$min\{N, u_sC^{h-1}\}.$$

The minimum number of dissemination hops for all peers to receive the chunk, H, can be derived by letting $u_sC^{H-1} \geq N$. Therefore,

$$H = 1 + \lceil log_C \frac{N}{u_s} \rceil.$$

Based on our assumptions that each chunk corresponds to 1 unit playback time and that any peer directly downloading from the server gets a streaming rate of R, we know that the first hop of the chunk dissemination from the server takes 1 unit time. Each other hop of the chunk dissemination from a peer to another takes $\frac{1}{u_p/C-1} = \frac{C-1}{u_p}$ time units. We can then calculate the minimum delay for all peers to receive the chunk as

$$D = 1 + \frac{C-1}{u_p} \cdot \lceil log_C \frac{N}{u_s} \rceil.$$

□

We now analyze the relationship between the volume of inter-ISP traffic and the minimum end-to-end delay in a multi-ISP P2P streaming system. We first use a two-ISP case to illustrate the idea of our model, and then present our model for the general multi-ISP case.

A. Two-ISP Case: An Illustration of Modeling Methodology

For ease of illustration, we assume server capacity u_s is only deployed in ISP 1, and thus there is only inter-ISP traffic from ISP 1 to ISP 2. We also simplify notations: Let K denote the number of virtual agents who download live streams from ISP 1 to ISP 2, with corresponding inter-ISP traffic KR. The delay of links between the two ISPs is τ. Suppose agent l ($1 \leq l \leq K$) downloads a chunk (any chunk in the live stream) at hop $h^{(l)}$ (counting from the time the server pumps out the chunk) from a peer in ISP 1, who has received the chunk previously. and we assume $1 \leq h^{(1)} \leq h^{(2)} \leq ... \leq h^{(K)}$ without loss of generality. Let $\mathscr{H} = \{h^{(1)}, \ldots, h^{(K)}\}$. We next calculate the minimum end-to-end chunk dissemination delay in each ISP. The maximum between the two is the minimum chunk dissemination delay in the entire system.

ISP 1. Let H_1 be the minimum number of dissemination hops for all peers in ISP 1 to receive a chunk. To achieve minimum end-to-end dissemination delay in ISP 1, the snow-ball algorithm discussed in Lemma 1 can be applied, which distributes a chunk to $u_sC_1^{H_1-1}$ peers within H_1 hops in the ISP, if no copies of the chunk branch off as pulled by agents into ISP 2. However, if agent l does pull a copy of the chunk from ISP 1 at hop $h^{(l)}$, there will be a reduction of $C_1^{H_1-h^{(l)}}$ peers in

ISP 1 that could receive the chunk within the H_1 hops of dissemination, derived as follows: If one peer in ISP 1 received the chunk rather than agent l at hop $h^{(l)}$, it could have further distributed the chunk to $C_1 - 1$ peers in hop $h^{(l)} + 1$, and then all C_1 peers could have further sent the chunk to $C_1(C_1 - 1)$ peers at hop $h^{(l)} + 2$, and so on. The total number of peers that could have received the chunk from hop $h^{(l)}$ to hop H_1 is therefore $1 + C_1 - 1 + C_1(C_1 - 1) + \ldots + C_1^{H_1 - h^{(l)} - 1}(C_1 - 1) = C_1^{H_1 - h^{(l)}}$.

In the case that K agents each retrieve one copy of the chunk at hop $h^{(1)}, \ldots, h^{(k)}$, respectively, the number of peers in ISP 1 that can receive the chunk in H_1 hops becomes $u_s C_1^{H_1-1} - \sum_{l=1}^{K} C_1^{H_1 - h^{(l)}}$. Therefore, the minimum number of hops for all peers in ISP 1 to receive the chunk in this case, H_1, can be derived by letting $u_s C_1^{H_1-1} - \sum_{h \in \mathscr{H}} C_1^{H_1-h} \geq N_1$. We derive

$$H_1 = 1 + \lceil log_{C_1} \frac{N_1}{u_s - \sum_{h \in \mathscr{H}} C_1^{1-h}} \rceil.$$

Again, the first hop of chunk dissemination from the server takes 1 time unit, and each other hop in ISP 1 takes $t_h = \frac{C_1 - 1}{u_{p1}}$ time units. The minimum end-to-end delay for all peers to receive the chunk in ISP 1 is

$$D_1 = 1 + t_h \cdot \lceil log_{C_1} \frac{N_1}{u_s - \sum_{h \in \mathscr{H}} C_1^{1-h}} \rceil. \qquad (2)$$

Comparing (2) with (1), we see that the effective server capacity used to serve peers in ISP 1 is indeed

$$\mathscr{U}_{s1} = u_s - \sum_{h \in \mathscr{H}} C_1^{1-h}.$$

ISP 2. When virtual agent l retrieves a chunk from ISP 1 at hop $h^{(l)}$, indeed a peer in ISP 2 obtains the chunk from ISP 1 at time $1 + t_h \cdot (h^{(l)} - 2) + \tau$ after the server pumps out the chunk (1 time unit for the server pumping out the chunk, $t_h \cdot (h^{(l)} - 2)$ time units for the chunk dissemination in ISP 1, τ time units for the chunk traversing the inter-ISP link), where τ is the delay of inter-ISP links. We next model the upload bandwidth that peers in ISP 1 use to serve peers in ISP 2 as effective server capacity deployed in ISP 2, in order to analyze the end-to-end chunk dissemination delay in ISP 2 using a similar methodology as applied previously:

Let $\tilde{u}^l$ denote the effective server capacity introduced by agent l. We can assume there is an imaginary server deployed in ISP 2, with the capacity of $\sum_{l=1}^{K} \tilde{u}^l$. The chunk disseminated from the server in ISP 1 to a peer in ISP 2 using a time $1 + t_h \cdot (h^{(l)} - 2) + \tau$, can be equivalently considered as distributed from an imaginary server in ISP 2 to the peer after traveling $1 + (h^{(l)} - 2) + \frac{\tau}{t_h}$ hops. This number of hops is calculated as follows: The first hop from the imaginary server takes 1 time unit, each other hop from one peer to another peer in ISP 2 takes t_h, $t_h = \frac{C_2 - 1}{u_{p2}} = \frac{C_1 - 1}{u_{p1}}$, time units, and the time $1 + t_h \cdot (h^{(l)} - 2) + \tau$ is thus equivalently to $1 + (h^{(l)} - 2) + \frac{\tau}{t_h}$ hops in ISP 2. With imaginary server capacity $\tilde{u}^l$, after $1 + (h^{(l)} - 2) + \frac{\tau}{t_h}$ hops, $\tilde{u}^l \cdot C_2^{1 + (h^{(l)} - 2) + \frac{\tau}{t_h} - 1} = 1$ peer receives the chunk. We can thus derive the effective server capacity introduced by agent l as $\tilde{u}^l = C_2^{(2 - h^{(l)} - \frac{\tau}{t_h})}$. Therefore, the total effective server capacity in ISP 2, introduced by K agents, is

$$\mathscr{U}_{s2} = \sum_{l=1}^{K} \tilde{u}^l = \sum_{h \in \mathscr{H}} C_2^{(2 - h - \frac{\tau}{t_h})}.$$

Suppose the imaginary server in ISP 2 pumps up the chunk at the same time as the server in ISP 1 does. The end-to-end dissemination delay in ISP 2, the duration from the time the server in ISP 1 pumps out the chunk to the time all peers in ISP 2 have received the chunk, can be calculated using a similar methodology as used before: let H_2 be the maximum number of dissemination hops in ISP 2 from its imaginary server to all peers. $u_{s2} \cdot C_2^{H_2 - 1}$ peers in ISP 2 can have the chunk after H_2 hops. Letting $u_{s2} \cdot C_2^{H_2 - 1} \geq N_2$, we derive

$$H_2 = 1 + \lceil log_{C_2} \frac{N_2}{\sum_{h \in \mathscr{H}} C_2^{(2 - h - \frac{\tau}{t_h})}} \rceil.$$

Recall that the first hop of the chunk dissemination from the server takes 1 unit time and each other hop in ISP 2 takes $t_h = \frac{C - 1}{u_{p2}}$ time units. The minimum end-to-end delay for all peers in ISP 2 to receive the chunk is

$$D_2 = 1 + t_h \cdot \lceil log_{C_2} \frac{N_2}{\sum_{h \in \mathscr{H}} C_2^{(2 - h - \frac{\tau}{t_h})}} \rceil. \qquad (3)$$

Therefore, the minimum chunk dissemination delay in the entire system can be derived as follows, while there are K agents that incur inter-ISP traffic KR:

$$D = \max\{D_1, D_2\}.$$

B. Multiple-ISP Case: Characterization and Analysis

We next model inter-ISP traffic and streaming performance with multiple ISPs in the system. Specifically, given the numbers of agents across ISPs (which correspond to volumes of inter-ISP traffic), we analyze the minimum end-to-end dissemination delay in each ISP, and then calculate the maximum among all as the minimum chunk dissemination delay in the entire system.

ISP *i*. Let $\mathscr{U}_{si}$ denote the effective server capacity in ISP i ($i = 1, \ldots, M$), which is the sum of actual deployed capacity u_{si} and server capacity brought by agents pulling chunks into the ISP, minus the capacity branching off by agents pulling chunks out of the ISP. Let H_i be the number of hops needed for all peers in ISP i to receive any chunk in the live stream, counting from the time the server pumps out the chunk. Based on the snow ball approach in Lemma 1, we know the maximum number of peers that can receive the chunk in H_i hops is $\mathscr{U}_{si} \cdot C_i^{H_i - 1}$. Letting $\mathscr{U}_{si} \cdot C_i^{H_i - 1} \geq N_i$ where N_i is the number of peers in ISP i, we derive

$$H_i = 1 + \lceil log_{C_i} N_i / \mathscr{U}_{si} \rceil.$$

Since the delay of each hop in ISP i is t_h except the first hop from servers which takes 1 time unit, the minimum end-to-end dissemination delay in ISP i (counting from the time the server pumps out the chunk) is

$$D_i = 1 + t_h \cdot \lceil log_{C_i} \frac{N_i}{\mathscr{U}_{si}} \rceil. \tag{4}$$

Let $h^l_{j,i}$ denote the hop at which agent l pulls a chunk from ISP j to ISP i. Equivalently, it means that a peer in ISP i obtains the chunk at time $1+t_h \cdot (h^l_{j,i}-2)+\tau_{j,i}$ after the server pumps out the chunk (recall that we assume all servers in all ISPs pump out the same chunk at the same time). Similar to our analysis on ISP 2 in the two-ISP case, the chunk disseminated from the server in ISP j to a peer in ISP i using $1+t_h \cdot (h^l_{j,i}-2)+\tau_{j,i}$ time slots, can be equivalently considered as distributed from an imaginary server in ISP i to the peer after traveling $1+(h^l_{j,i}-2)+\frac{\tau_{j,i}}{t_h}$ hops. Let $\tilde{u}^l_{j,i}$ be the effective server capacity brought by agent l into ISP i, which can be calculated as follows: With server capacity $\tilde{u}^l_{j,i}$, after $1+(h^l_{j,i}-2)+\frac{\tau_{j,i}}{t_h}$ hops, $\tilde{u}^l_{j,i} \cdot C_i^{1+(h^l_{j,i}-2)+\frac{\tau_{j,i}}{t_h}-1} = 1$ peer in ISP i receives the chunk. Therefore,

$$\tilde{u}^l_{j,i} = C_i^{(2-h^l_{j,i}-\frac{\tau_{j,i}}{t_h})}.$$

On the other hand, suppose agent l' pulls a chunk from ISP i to ISP j at hop $h^{l'}_{i,j}$. This results in one fewer peer to receive the chunk in ISP i at hop $h^{l'}_{i,j}$. Let $\bar{u}^{l'}_{i,j}$ be the reduced server capacity from ISP i due to agent l' pulling out the chunk. Using $\bar{u}^{l'}_{i,j} \cdot C_i^{h^{l'}_{i,j}-1} = 1$, we derive

$$\bar{u}^{l'}_{i,j} = C_i^{1-h^{l'}_{i,j}},$$

which is a generalization of the analytical result for ISP 1 in the two-ISP case.

Let $\mathscr{H}_{j,i}$ denote the set of hops at which $K_{j,i}$ agents pull chunks into ISP i from ISP j ($j \neq i$). Let $\mathscr{H}_{i,j}$ denote the set of hops at which $K_{i,j}$ agents pull chunks out of ISP i to ISP j ($j \neq i$). The overall effective server capacity in ISP i is

$$\begin{aligned} \mathscr{U}_{si} &= u_{si} + \sum_{j=1,j\neq i}^{M} [\sum_{h\in\mathscr{H}_{j,i}} C_i^{(2-h-\frac{\tau_{j,i}}{t_h})} \\ &- \sum_{h\in\mathscr{H}_{i,j}} C_i^{(1-h)}], \quad 1 \le i \le M. \end{aligned} \tag{5}$$

With the above analysis, the minimum end-to-end dissemination delay in the entire system is given in the following theorem.

Theorem 1. *Consider a P2P live streaming system with N peers over M ISPs, where N_i peers are distributed in ISP i with average upload capacity u_{pi} and C_i neighbors. The delay on intra-ISP links is $t_h = \frac{C_i-1}{u_{pi}}$ ($1 \le i \le M$), and the latency on inter-ISP links between ISP j and ISP i is $\tau_{j,i}$. The amount of server capacity deployed in ISP i is u_{si}. $K_{j,i}$ copies of the stream are distributed into ISP i from ISP j ($i \neq j$). The minimum end-to-end delay for all peers in the system to receive a chunk is*

$$D = \max\{D_1, D_2, ..., D_M\}, \tag{6}$$

$$D_i = 1 + t_h \cdot \lceil log_{C_i} \frac{N_i}{\mathscr{U}_{si}} \rceil, \qquad 1 \le i \le M, \tag{7}$$

$$\begin{aligned} \mathscr{U}_{si} &= u_{si} + \sum_{j=1,j\neq i}^{M} [\sum_{h\in\mathscr{H}_{j,i}} C_i^{(2-h-\frac{\tau_{j,i}}{t_h})} \\ &- \sum_{h\in\mathscr{H}_{i,j}} C_i^{(1-h)}], \quad 1 \le i \le M, \end{aligned} \tag{8}$$

where $\mathscr{U}_{si}$ is the effective server capacity to serve peers in ISP i.The amount of inter-ISP traffic incurred is $\sum_{i=1}^{M}\sum_{j=1,j\neq i}^{M} K_{j,i}R$, and the percentage over the total amount of P2P streaming traffic is

$$F = \frac{\sum_{i=1}^{M}\sum_{j=1,j\neq i}^{M} K_{j,i}R}{N \cdot R}.$$

Theorem 1 gives the generic relationship between the volume of inter-ISP traffic and the end-to-end chunk dissemination delay in multi-ISP systems. We discuss the implications of Theorem 1 with two corollaries.

Corollary 1. *Given server capacity deployment $u_{s1}, u_{s2}, \ldots, u_{sM}$, the amount of inter-ISP traffic needed when minimum end-to-end delay is achieved in the system, can be derived by solving the following optimization problem, where $K_{j,i}$'s, $h^l_{j,i}$'s, are optimization variables:*

$$\min D$$

$$\text{Subject to: Constraints } (6) - (8).$$

We can use sequential quadratic programming (SQP) to solve this non-linearly constrained optimization problem. SQP is the most successful method for solving non-linear optimization problems [18] (with convergence properties extensively studied), which has been implemented in many packages including MATLAB.

Corollary 2. *If we require that no inter-ISP traffic should be incurred, the minimum end-to-end dissemination delay in the entire system is achieved when the deployment of overall server capacity u_s among the ISPs satisfies $log_{C_1}\frac{N_1}{u_{s1}} = \ldots = log_{C_M}\frac{N_M}{u_{sM}}$.*

Corollary 2 can be illustrated as follows: When there is no inter-ISP traffic, we have $\mathscr{U}_{si} = u_{si}$ and $\sum_{i=1}^{M}\mathscr{U}_{si} = u_s$. If the end-to-end delay in ISP i, D_i, decreases, we know the effective server capacity $\mathscr{U}_{si}$ in ISP i must have been increased, according to Eqn. (7). Meanwhile, there must exist another ISP j, whose effective server capacity $\mathscr{U}_{sj}$ decreases, and thus its end-to-end delay D_j increases. Therefore, the minimum end-to-end delay in the entire system occurs when $D = D_1 = D_2 = \ldots = D_M$, which is equivalent to $log_{C_1}\frac{N_1}{u_{s1}} = \ldots = log_{C_M}\frac{N_M}{u_{sM}}$, *i.e.*, when the inter-ISP traffic is completely blocked, the minimum delay in the system occurs when the server capacity deployment in the ISPs satisfies the equations in Corollary 2.

We further discuss implications of our model based on the theorem and corollaries. From Eqn. (8), we derive the overall effective server capacity in the system as

$$\mathscr{U}_s = \sum_{i=1}^{M} \mathscr{U}_{si} = u_s + \sum_{i=1}^{M} \sum_{j=1,j\neq i}^{M} [\sum_{h\in\mathscr{H}_{j,i}} C_i^{(2-h-\frac{\tau_{j,i}}{t_h})} - \sum_{h\in\mathscr{H}_{i,j}} C_i^{(1-h)}].$$

Such a total effective server capacity $\mathscr{U}_s$ may not be equal to the overall deployed server capacity u_s. We illustrate this point by comparing the effective server capacity brought into ISP i, $\tilde{u}_{j,i}^l = C_i^{2-h_{j,i}^l-\frac{\tau_{j,i}}{t_h}}$, and the effective server capacity branching off from ISP j, $\bar{u}_{j,i}^l = C_j^{1-h_{j,i}^l}$, when an agent l pulls a chunk from ISP j to ISP i. The difference $\tilde{u}_{j,i}^l - \bar{u}_{j,i}^l = C_j^{1-h_{j,i}^l} \cdot [(\frac{C_i}{C_j})^{1-h_{j,i}^l} \cdot C_i^{1-\frac{\tau_{j,i}}{t_h}} - 1]$ may not be equal to 0, and its sign is decided by upload bandwidths of peers in ISPs i and j (as reflected by C_i and C_j), as well as latencies on inter-ISP links and intra-ISP links (*i.e.*, $\tau_{j,i}$ and t_h). We divide our discussions into two cases.

Case 1: Equal peer upload capacity in all ISPs, *i.e.*, $u_{p1} = u_{p2} = \ldots = u_{pM}$. We then know $C_1 = C_2 = \ldots = C_M$, $\tilde{u}_{j,i}^l - \bar{u}_{j,i}^l = C_j^{1-h_{j,i}^l}(C_i^{1-\frac{\tau_{j,i}}{t_h}} - 1) \leq 0$ (since $\tau_{j,i} \geq t_h$), and therefore $\mathscr{U}_s \leq u_s$. This gives us the following intriguing insights:

(i) When all peers have the same upload capacity, any cross-ISP chunk download will lead to decrease of the total effective server capacity, and thus increase of end-to-end chunk dissemination delay in the entire system. Therefore, the best streaming performance occurs when peers stream within their ISP boundaries (*i.e.*, minimum end-to-end dissemination delay and minimum inter-ISP traffic of zero occur concurrently), when server capacity deployment satisfies the condition in Corollary 2.

With $C_1 = \ldots = C_M$, the optimal server capacity distribution in Corollary 2 becomes $u_{si} = \frac{N_i}{N} u_s, \forall i$. It shows that no matter how peer population is distributed in different ISPs, as long as the server capacity is deployed proportionally to the number of peers in each ISP, the minimum end-to-end chunk dissemination delay, $D = 1 + t_h \cdot \lceil log_{C_i} \frac{N_i}{N} u_s \rceil$, and the minimum inter-ISP traffic, 0, occur concurrently.

Case 2: Different peer upload capacities in different ISPs. If $u_{pj} > u_{pi}$ (*i.e.*, $C_j > C_i$), we know $\frac{C_j}{C_i}^{(h_{j,i}^l-1)} \cdot C_i^{1-\frac{\tau_{j,i}}{t_h}}$ could be no smaller than 1 when $\tau_{j,i}$ is close to t_h, and then we can derive $\tilde{u}_{j,i}^l - \bar{u}_{j,i}^l = C_j^{1-h_{j,i}^l} \cdot [(\frac{C_i}{C_j})^{1-h_{j,i}^l} \cdot C_i^{1-\frac{\tau_{j,i}}{t_h}} - 1] \geq 0$. This shows us the following:

(ii) To achieve minimum end-to-end dissemination delay in the entire system, peers in ISPs with smaller upload bandwidth should try to download from ISPs with larger bandwidth if the inter-ISP link delay is small, such that the total effective server capacity can be increased. In this case, the larger the inter-ISP traffic is, the smaller the end-to-end dissemination delay in the system becomes.

(iii) When the inter-ISP link latency is large, even if peers in an ISP with small peer upload bandwidth stream from another ISP with larger peer bandwidth, $\frac{C_j}{C_i}^{(h_{j,i}^l-1)} \cdot C_i^{1-\frac{\tau_{j,i}}{t_h}}$ could still be smaller than 1, and the total effective server capacity decreases. In this situation, cross-ISP download should be discouraged, in order to achieve smaller chunk dissemination delay as well as lower inter-ISP traffic.

(iv) Server capacities are better deployed more into ISPs with large peer upload capacities and small inter-ISP link delays to other ISPs, rather than in ISPs otherwise, in order to achieve the smallest end-to-end dissemination delay in the entire system. If an ISP has a high inter-ISP latency with other ISPs, it is beneficial to deploy some server capacity in the ISP.

(v) Deploying more server capacity in ISPs with more peers decreases the volume of inter-ISP traffic, while it may simultaneously increase the end-to-end dissemination delay, when peer upload bandwidths in those ISPs are not high.

IV. EMPIRICAL STUDIES

We next investigate the relationship among volumes of inter-ISP traffic, streaming performance, and server capacity deployment, as captured by our models, using large-scale empirical studies under realistic settings. A discrete-event simulator is implemented, which can simulate tens of thousands of peers and multiple ISPs, as well as streaming servers deployed in different ISPs and a tracker server in the system which maintains information of chunk availability at the peers. In our default settings, we simulate $N = 10,000$ peers in the entire system, which are evenly distributed in 4 ISPs. The average peer upload capacities are 1.4 in ISP 1, 1.2 in ISP 2, 1.2 in ISP 3, and 1 in ISP 4, respectively. The 4 ISPs are all connected by peering links, so cross-ISP traffic may exist between any two ISPs. A total amount of $u_s = 100$ server capacity is deployed in the system, which is the sum of upload capacities at streaming servers in all ISPs. The tracker server provides a list of 50 candidate peers to each peer as neighbors. The active neighbors to whom a peer uploads chunks to are selected from the candidate peers, with a number of 7, 6, 6, and 5 for peers in the four ISPs, respectively. The delay on each intra-ISP link is 5 time units. All inter-ISP link delays are the same.

In order to control the numbers of copies of each chunk retrieved across ISP boundaries (*i.e.*, the amount of inter-ISP traffic), agents are simulated in our simulator (though they do not indeed exist in a practical system), each of which downloads chunks from one ISP and forwards them to another ISP. For example, agent $A_{j,i}$ forwards copies of streaming chunks from ISP j to ISP i: when we wish to simulate the retrieval of $K_{j,i}$ copies of a chunk from ISP j to ISP i, agent $A_{j,i}$ pulls $K_{j,i}$ copies of the chunk from peers in ISP j at time units $1+t_h(h^l_{j,i}-2), 1 \leq l \leq K_{j,i}$, respectively, and for each copy, it forwards it to a peer in ISP i that needs the chunk after a delay of $\tau_{j,i} - t_h$ time units (in order to simulate the effect of inter-ISP link delay). The tracker server logs the chunk availability in each peer. Thus, the tracker server can record the time units needed for all peers to obtain a chunk after the server pumps out it, which is the chunk dissemination delay.

A. End-to-End Chunk Dissemination Delay vs. Inter-ISP Traffic

We first fix the deployment of server capacity in the four ISPs, and study the relation between percentage of inter-ISP traffic (over the total amount of streaming traffic) and end-to-end chunk dissemination delay in Fig. 2. Different amounts of inter-ISP traffic are simulated by controlling the number of copies of chunks retrieved across ISP boundaries. Three cases of inter-ISP latency are investigated, where L represents the ratio of inter-ISP link delay over intra-ISP link delay: (1) low inter-ISP link delay with $L = 1$, (2) medium inter-ISP link latency with $L = 2$, and (3) high inter-ISP link latency with $L = 5$.

We observe that when all server capacity is deployed in ISP 1 with the largest peer upload bandwidth (the case in Fig. 2(a)), the larger the inter-ISP traffic is, the lower the end-to-end dissemination delay is, which validates insight (ii) in the previous section. The largest percentage of inter-ISP traffic is 75%, since each of the four ISPs has the same number of peers.

When server capacity is deployed in all ISPs (the case in Fig. 2(b)), the development of chunk dissemination delay with the increase of inter-ISP traffic is in general not monotonic; there exists an optimal volume of inter-ISP traffic with which the minimum dissemination delay is achieved. In addition, we observe that the larger the inter-ISP link delay is, the more inter-ISP traffic is required to achieve the best streaming performance.

B. Impact of Server Capacity Deployment

We next study how server capacity deployment affects the minimum end-to-end dissemination delay in the system and the required inter-ISP traffic to achieve this delay (which can be derived by solving the optimization problem in Corollary 1). In this set of experiments, different amounts of server capacity are deployed in four ISPs as follows: we vary u_{s1} (server capacity in ISP 1) from 0 to 100, and meanwhile vary $u_{s2} + u_{s3}$ (the total amount deployed in ISPs 2 and 3) between 0 and 100 as well; giving u_{s1} and $u_{s2} + u_{s3}$, u_{s4} will be fixed, as the total server capacity in the entire system is 100. Low inter-ISP link delays with $L = 1$ are used.

Fig. 3 (a) shows that when more server capacity is deployed in ISPs with lager peer upload bandwidths, the end-to-end delay in the system is smaller: the end-to-end delay is the lowest when all server capacity is deployed in ISP 1; when all capacity is deployed in ISP 2 and ISP 3, the delay is lower than that in the case when all is deployed in ISP 4. This is also what insight (iv) implies.

On the other hand, Fig. 3 (b) shows that when all server capacity is deployed in ISP 1, the volume of inter-ISP traffic is the largest. When all server capacity is deployed in ISP 2 and ISP 3, the volume of inter-ISP traffic is smaller but is larger than that in the case when all is deployed in ISP 4. This can also be concluded from insight (ii).

C. Impact of Peer Population Distribution

We further examine the minimum end-to-end chunk dissemination delay and the inter-ISP traffic required to achieve it, in four cases with different server capacity and peer population distributions:
(1) $(N_1, N_2, N_3, N_4) = (7000, 1000, 1000, 1000)$, $(u_{s1}, u_{s2}, u_{s3}, u_{s4}) = (100, 0, 0, 0)$;
(2) $(N_1, N_2, N_3, N_4) = (1000, 4000, 4000, 1000)$, $(u_{s1}, u_{s2}, u_{s3}, u_{s4}) = (0, 50, 50, 0)$;
(3) $(N_1, N_2, N_3, N_4) = (1000, 1000, 1000, 7000)$, $(u_{s1}, u_{s2}, u_{s3}, u_{s4}) = (0, 0, 0, 100)$;
(4) $(N_1, N_2, N_3, N_4) = (2500, 2500, 2500, 2500)$, $(u_{s1}, u_{s2}, u_{s3}, u_{s4}) = (25, 25, 25, 25)$.

From Fig. 4 (a), we observe that when the inter-ISP link latency is low ($L = 1$ or $L = 2$), the streaming performance in Case 1 (where server capacity is all deployed in ISP 1 with the largest peer capacity and most peers) is the best, which verifies insight (iv) in the previous section. When the inter-ISP link latency is large ($L = 5$), uniform server capacity deployment among all ISPs with even peer population distribution (Case 4) achieves the best streaming performance. This provides us another insight: to achieve minimum dissemination delay when inter-ISP link latencies are large, servers should be deployed in more ISPs with an amount relative to the peer populations.

In addition, Fig. 4 (b) further reveals that when most server capacity is deployed in ISPs with the majority of peer population and whose peer average upload bandwidth is not the highest in the system (Case 2 and Case 3), the volume of inter-ISP traffic is small. This verifies insight (v) in the previous section.

D. Impact of Different Inter-ISP Link Latencies

In our previous studies, the latencies on all inter-ISP links are the same. We next investigate how different inter-ISP link delays influence the server capacity deployment, inter-ISP traffic, and chunk dissemination delay in the system. In these set of experiments, the inter-ISP link delay is $L = 5$ between ISP 2 and each of the other ISPs, and is $L = 2$ between any two of ISP 1, ISP 3, and ISP 4. Server capacities are deployed in ISPs 1 and 2, but not in the other two. All other settings are the same as in previous experiments.

Fig. 5 (a) shows that the minimum dissemination delay is achieved when a small amount of server capacity is

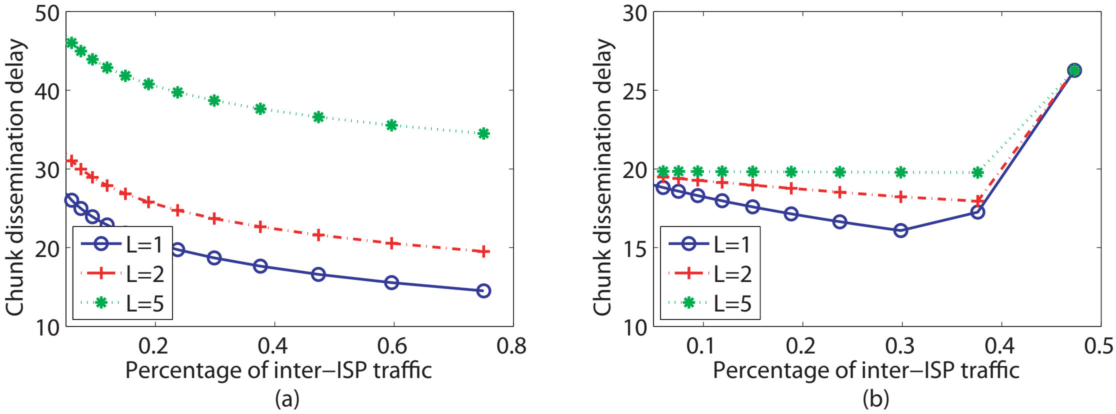

Figure 2. End-to-end chunk dissemination delay vs. inter-ISP traffic: (a) $(u_{s1}, u_{s2}, u_{s3}, u_{s4}) = (100, 0, 0, 0)$; (b) $(u_{s1}, u_{s2}, u_{s3}, u_{s4}) = (45, 10, 10, 35)$.

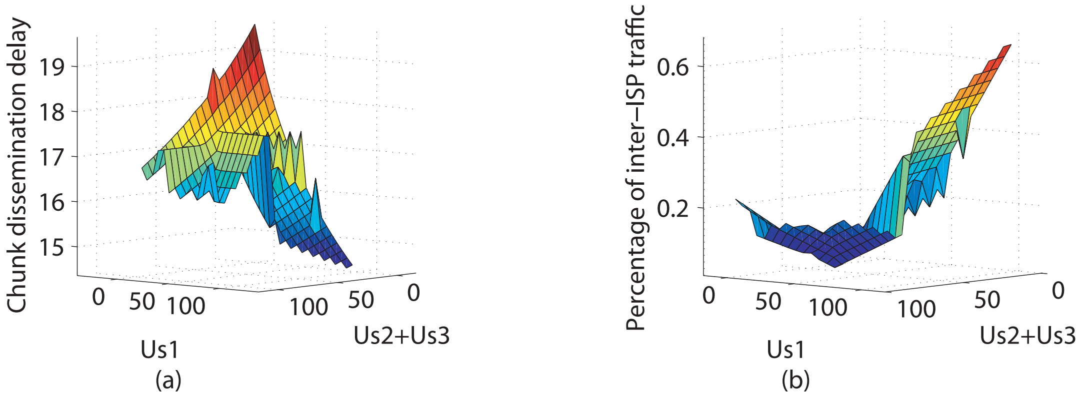

Figure 3. Minimum end-to-end chunk dissemination delay and the required inter-ISP traffic at different server capacity deployments.

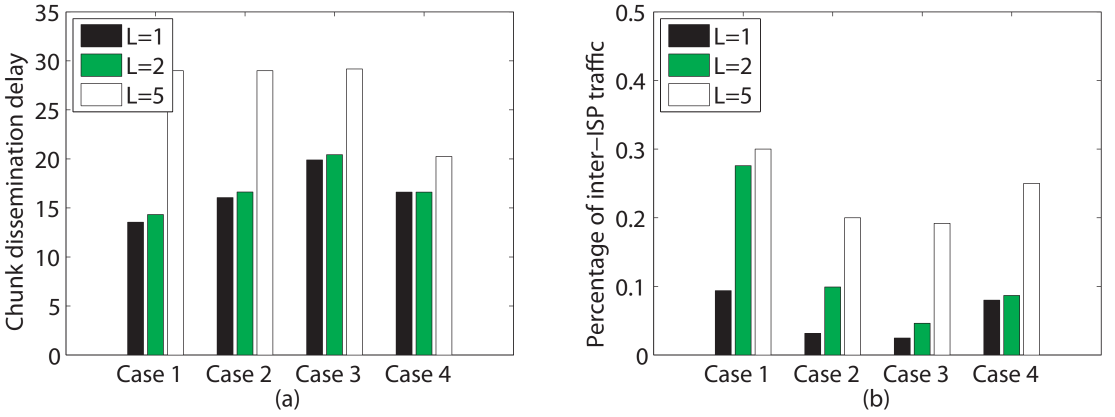

Figure 4. Minimum end-to-end chunk dissemination delay and the required inter-ISP traffic at different server capacity and peer population distributions.

deployed in ISP 2. As the server capacity in ISP 2 increases, the system performance becomes worse. These observations are consistent with insight (iv).

Fig. 5 (b) shows that the minimum inter-ISP traffic is achieved when server capacity is evenly distributed in ISP 2 and ISP 3, considering that both ISPs have the same peer upload bandwidths and peer population. When the percentages of server capacity in ISP 2 and ISP 3 become unbalanced, the inter-ISP traffic is larger. The decrease of inter-ISP traffic when most server capacity is delayed in ISP 2 can be explained by the large link delay between ISP 2 and each of the other ISPs, which discourages inter-

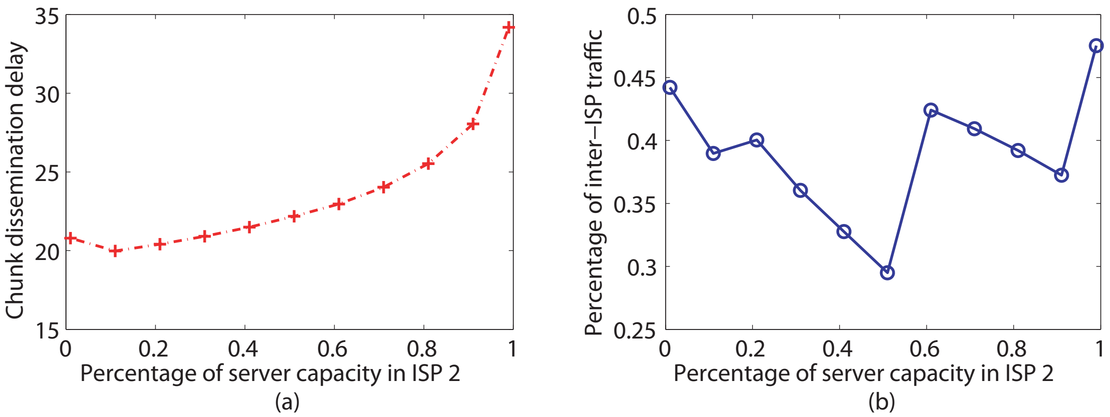

Figure 5. Minimum end-to-end chunk dissemination delay and the required inter-ISP traffic at different server capacity deployments under different inter-ISP link latencies.

ISP chunk retrieval.

V. RELATED WORK

A number of work have been conducted on theoretical analysis of P2P live streaming systems, in order to derive useful insights for performance improvement. Liu *et al.* [11] derive performance bounds of the streaming system, including minimum server load, maximum streaming rate, and minimum tree depth under different peer selection constraints, and also construct tree-based streaming systems to validate these bounds. Chen *et al.* [13] explore the streaming capacity of a streaming system with node degree bound. They propose a Bubble Algorithm for a system with bound on node out-degrees and a Cluster-tree Algorithm for systems with logarithmic bound on node's total degree. Liu [10] analyzes the chunk dissemination delay bound in P2P live streaming systems, based on a snow-ball algorithm. In Kumar *et al.*'s work [9], peers' upload bandwidth constraints are considered, and a stochastic fluid model is applied to calculate the maximum streaming rate that a churnless system can sustain and the probability of universal streaming at the streaming rate in a system with churns. Bonald *et al.* [16] treat chunk dissemination in a P2P live streaming system as a diffusion process, and analyze peer/chunk selection algorithms as diffusion schemes. They derive the diffusion rate and delay of chunks based on various diffusion schemes, and prove the qualitative result that the random peer/latest useful chunk selection algorithm can achieve the best diffusion at an optimal rate within an optimal delay. Zhou *et al.* [14] use recursive equations to model the buffer occupancy at peers under two chunk selection strategies, greedy and rarest first, and derive streaming continuity and start-up latency in the system. Massoulie *et al.* [12] propose a chunk dissemination heuristic for node-capacitated P2P networks, based on an optimal packet forwarding algorithm in edge-capacitated networks, and prove that the heuristic works well in complete graphs. All the above work does not take inter-ISP traffic into consideration in their analytical models.

There have been a number of algorithm/protocol designs to induce traffic locality in P2P systems. P4P [6] is a general solution for cooperative traffic control in P2P applications. In P4P, an iTracker portal is introduced as interfaces for network providers to provide information of the underlying network to P2P solution providers, in order to achieve traffic optimization. Picconi *et al.* [7] propose an algorithm to construct two levels of overlays: a primary overlay is created preferentially among nearby peers and a secondary overlay connects peers randomly across the network. Magharei *et al.* [8] construct a localized overlay by explicitly controlling the number of external connections that peers in one ISP could establish with peers in other ISPs. They propose an inter-ISP scheduling algorithm for delivering streams to individual ISPs and an intra-ISP scheduling algorithm to ensure the delivery of streams to all internal peers in an ISP. They also analytically prove the feasibility of streaming over such a localized overlay with limited external connections.

VI. CONCLUDING REMARKS

This paper targets at in-depth investigation of the impact of locality-aware protocol design on the achievable streaming performance in a P2P live streaming system. Towards this objective, we carefully model the relationship between volumes of inter-ISP traffic and the streaming performance, in P2P streaming systems over multiple ISPs at different bandwidth levels. In particular, we derive analytical formulas to derive the minimum end-to-end chunk dissemination delay in the system, as well as the amount of inter-ISP traffic needed to achieve this minimum delay. We also explore the best server capacity deployment strategies to achieve the best streaming performance, when minimum inter-ISP traffic is incurred. Our models and analyses provide useful insights on the design of efficient locality-aware P2P streaming protocols and effective server deployment strategies across multiple ISPs, which can achieve desired goals on inter-ISP traffic minimization or streaming performance optimization.

ACKNOWLEDGEMENT

This work has been supported in part by the University of Hong Kong under Small Project Funding, and the Research Grants Council of Hong Kong under RGC General Research Fund (Ref: HKU718710E).

REFERENCES

[1] *PPLive*, http://www.pplive.tv.

[2] *SopCast*, http://www.sopcast.org/.

[3] *Zattoo*, http://www.zattoo.com/.

[4] C. S. Inc., "Cisco Visual Networking Index: Forecast and Methodology, 2009-2014, White Paper," 2010.

[5] S. Sen, O. Spatscheck, and D. Wang, "Accurate, Scalable In-Network Identification of P2P Traffic Using Application Signatures," in *Proc. of WWW*, May 2004.

[6] H. Y. Xie, Y. R. Yang, A. Krishnamurthy, Y. B. G. Liu, and A. Silberschatz, "P4P: Provider Portal for Applications," in *Proc. of ACM SIGCOMM*, August 2008.

[7] F. Picconi and L. Massoulie, "ISP Friend or Foe? Making P2P Live Streaming ISP-Aware," in *Proc. of IEEE ICDCS*, June 2009.

[8] N. Magharei, R. Rejaie, V. Hilt, I. Rimac, and M. Hofmann, "ISP-Friendly Live P2P Streaming," in *Poster. of ACM SIGCOMM*, August 2009.

[9] R. Kumar, Y. Liu, and K. Ross, "Stochastic Fluid Theory for P2P Streaming Systems," in *Proc. of IEEE INFOCOM*, May 2007.

[10] Y. Liu, "On the Minimum Delay Peer-to-peer Video Streaming: How Realtime Can It Be?" in *Proc. of MULTIMEDIA*, September 2007.

[11] S. Liu, R. Zhang-Shen, W. J. Jiang, J. Rexford, and M. Chiang, "Performance Bounds for Peer-Assisted Live Streaming," in *Proc. of SIGMETRICS*, June 2008.

[12] L. Massoulie, A. Twigg, C. Gkantsidis, and P. Rodriguez, "Randomized Decentralized Broadcasting Algorithms," in *Proc. of IEEE INFOCOM*, May 2007.

[13] S. Liu, M. H. Chen, S. Sengupta, M. Chiang, J. Li, and P. A. Chou, "P2P Streaming Capacity under Node Degree Bound," in *Proc. of IEEE ICDCS*, June 2010.

[14] Y. P. Zhou, D. M. Chiu, and J. C. S. Lui, "A Simple Model for Analyzing P2P Streaming Protocols," in *Proc. of IEEE ICNP*, October 2007.

[15] C. Feng, B. C. Li, and B. Li, "Understanding the Performance Gap Between Pull-Based Mesh Streaming Protocols and Fundamental Limits," in *Proc. of IEEE INFOCOM*, April 2009.

[16] T. Bonald, L. Massoulié, F. Mathieu, D. Perino, and A. Twigg, "Epidemic Live Streaming: Optimal Performance Trade-Offs," in *Proc. of ACM SIGMETRICS*, June 2008.

[17] *CoralReef Suite*, http://www.caida.org/tools/measurement/coralreef/.

[18] P. T. Boggs and J. W. Tolle, "Sequential Quadratic Programming," *Acta Numerica*, vol. 4, pp. 1–51, 1995.

Jian Zhao. Jian Zhao received his B.Engr. degree in 2009 from Department of Electronic Engineering and Information Science, University of Science and Technology of China, China. He is currently a Ph.D. candidate in the Department of Computer Science, the University of Hong Kong, China. His research interests include peer-to-peer streaming and optimization of network traffic.

Chuan Wu. Chuan Wu received her B.Engr. and M.Engr. degrees in 2000 and 2002 from Department of Computer Science and Technology, Tsinghua University, China, and her Ph.D. degree in 2008 from the Department of Electrical and Computer Engineering, University of Toronto, Canada. She is currently an assistant professor in the Department of Computer Science, the University of Hong Kong, China. Her research interests include measurement, modeling, and optimization of large-scale peer-to-peer systems and online/mobile social networks. She is a member of IEEE and ACM.