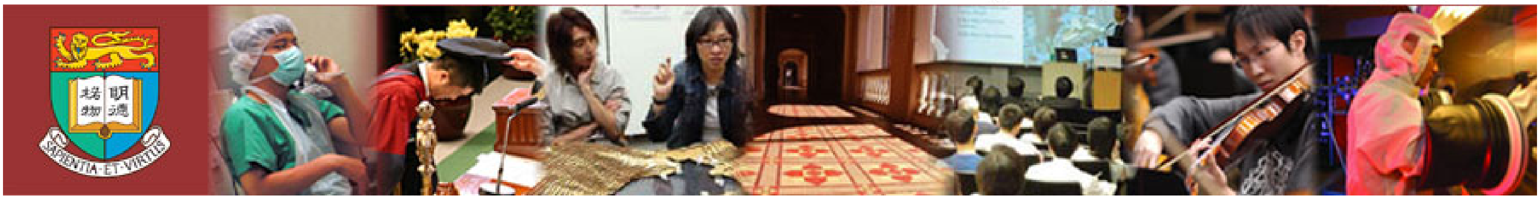


Title	Network aware P2P multimedia streaming: capacity or locality?
Author(s)	Jin, X; Kwok, YK
Citation	The 2011 IEEE International Conference on Peer-to-Peer Computing (P2P 2011), Kyoto, Japan, 31 August-2 September 2011. In Proceedings of P2P, 2011, p. 54-63
Issued Date	2011
URL	http://hdl.handle.net/10722/140240
Rights	IEEE International Conference on Peer-to-Peer Computing Proceedings. Copyright © IEEE.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE P2P 2011 proceedings.

Network Aware P2P Multimedia Streaming: Capacity or Locality?

Xin Jin and Yu-Kwong Kwok
Department of Electrical and Electronic Engineering
The University of Hong Kong
Pokfulam, Hong Kong SAR

***Abstract*—P2P content providers are motivated to localize traffic within Autonomous Systems and therefore alleviate the tension with ISPs stemming from costly inter-AS traffic generated by geographically distributed P2P users. In this paper, we first present a new three-tier framework to conduct a thorough study on the impact of various capacity aware or locality aware neighbor selection and chunk scheduling strategies. Specifically, we propose a novel hybrid neighbor selection strategy with the flexibility to elect neighbors based on either type of network awareness with different probabilities. We find that network awareness in terms of both capacity and locality potentially degrades system QoS as a whole and that capacity awareness faces effort-based unfairness, but enables contribution-based fairness. Extensive simulations show that hybrid neighbor selection can not only promote traffic locality but lift streaming quality and that the crux of traffic locality promotion is active overlay construction. Based on this observation, we then propose a totally decentralized network awareness protocol, equipped with hybrid neighbor selection. In realistic simulation environments, this protocol can reduce inter-AS traffic from 95% to 38% – a locality performance comparable with tracker-side strategies (35%) under the premise of high streaming quality. Our performance evaluation results provide valuable insights for both theoretical study on selfish topologies and real-deployed system design.**

***Index Terms*—Peer-to-peer multimedia streaming, traffic locality, inter-AS traffic, network awareness, unstructured overlays.**

I. Introduction

Peer-to-peer (P2P) overlays, such as PPLive [1], Skype [2], and PPStream [3], have gained their popularity as a decentralized and efficient content distribution architecture by disruptively occupying the Internet traffic. However, traffic generated without control by P2P systems may deteriorate the network performance and pose significant financial losses to Internet service providers (ISPs) with active P2P users. Moreover, traffic locality is one achieving goal of a network-friendly design by matching the topology between overlay P2P networks and underlay physical networks.

To alleviate the tension between ISPs and P2P content providers, P2P overlays with "locality awareness" – overlays favoring cooperations among peers from the same ISP, instead of oblivious to the underlying network topology (underlay) – have recently become an imperative achieving goal of efficient overlay construction and motivated numerous works aiming at understanding existing commercial P2P systems through on-field measurements or proposing locality enforcement solutions [4], [5]. For instance, measurement studies of PPLive, SopCast, and TVAnts in [4] imply that PPLive and especially TVAnts exploit some form of locality, such as Autonomous System (AS) awareness, that is, favoring peers within the same AS. SopCast, though, seems to be completely unaware of peer location. Nevertheless, the dominant metric of peer selection in all the studied systems is shown to be peer upload bandwidth, which represents node capacity.

Despite the imperativeness of traffic locality promotion, peers dynamically cooperate with each other and form a completely decentralized topology, which implies the difficulty of efficient overlay construction due to the lack of a centralized entity with dominant coordinating authority. Thus, directly applying the idea of peer locality to existing P2P streaming systems is non-trivial. Moreover, autonomous peers – selfish, rational, and seeking for their own utility maximization – judiciously establish their neighborhood composition to boost their streaming quality. What is the impact of network awareness on the entire system performance? Will rational peers take network aware peer selection strategies? We study two kinds of network awareness: locality awareness (i.e., favoring peers with lower AS hops) and capacity awareness (i.e., preferring nodes with higher capacities). Simulation results demonstrate that both pure capacity and pure locality awareness can hinder chunk dissemination due to the clustering of peers. This points to the necessity of incentive provision for selfish peers to enable capacity or locality awareness, constituting an essential part of the future work.

In this paper, we propose novel overlay construction, neighbor selection and chunk scheduling strategies to investigate the impact of both capacity and locality awareness. Specifically, we make three contributions:

- **Three Tier Framework.** To thoroughly study strategies taken by trackers and normal peers, we divide the strategy hierarchy into three tiers: tracker-tier neighbor selection, peer-tier neighbor selection and chunk scheduling, which we believe matches well with existing epidemic protocol designs. We then propose novel locality aware and capacity aware strategies to fit in the framework. To flexibly study the impact of node capacity and network locality, a novel hybrid neighbor selection strategy is in place by

Thanks are due to the anonymous reviewers (in particular, Reviewer D) for their constructive suggestions to improve the quality and presentation of this paper, and Dr. Wolf Siberski for his valuable suggestions on the final version of the manuscript. This work was supported in part by the HKU Small Project Fund (Ref: 201007176340) donated by HKU SPACE Research Fund.

selecting either type of network aware peers with different probabilities.

- **Decentralized Network Awareness.** From insights learned from the three tier framework, we propose a totally decentralized network awareness protocol – a scalable, lightweight and self-organizing design with the capacity to enforce traffic locality by active overlay construction and adopting hybrid neighbor selection. Our protocol can progressively refine peer neighborhood, without the assistance of any centralized entity.
- **Extensive Simulations.** Extensive simulations are conducted to evaluate and compare various locality aware and capacity aware strategies. The study on the three tier framework shows that the crux of locality enforcement is active overlay construction and hybrid neighbor selection can flexibly break the tradeoff between capacity awareness and locality promotion. Moreover, rarest first chunk scheduling couples best with all the studied neighbor selection strategies and pure network awareness in terms of both traffic locality and node capacity establishes a clustered topology, hindering efficient chunk dissemination. We further evaluate our proposed protocol under peer dynamics, diverse underlay topologies, different peer and bandwidth distributions, etc.

The remainder of this paper is organized as follows. We first present a preliminary description of unstructured mesh network architecture in Section II, followed by the three-tier framework to enforce network awareness in Section III. Section IV proposes our decentralized network awareness protocol. Then, performance evaluation and simulation results are, respectively, discussed in Section V and Section VI. Finally, Section VII details recent advances in traffic locality and we present conclusions and future work in Section VIII.

II. Preliminaries

Unstructured meshes have already been extensively deployed and experimentally verified as not only scalable but efficient and robust for large-scale media streaming in both industry and research communities [1], [6]. Four constituent entities – bootstrap servers, trackers, streaming servers, and autonomous peer clients – coordinate and interconnect with each other, and evolve into an overlay graph with specific desirable characteristics, driving P2P video streaming into a thriving business.

Newly arriving peers join in the streaming system by first contacting the bootstrap server, which provides peers with a list of content channels to enjoy. Each channel is coupled with a tracker maintaining a directory of peers currently residing in this channel and a streaming server (a.k.a., source node) segmenting the media content into chunks with the same size and pumping media chunks into the streaming overlay. After selecting a channel to enjoy, newly joining peers harvest a list of peers residing in the same channel as neighbors from the corresponding tracker. Of course, each peer can further update its neighbor list by asking the tracker or querying its neighbors. This neighbor selection process is commonly known as *overlay construction*, determining the connection relationships among peers.

Subsequent to overlay construction, peers then exchange their buffer maps with neighbors to indicate the chunk availability in their buffers and schedule chunks by adopting pull mode, push mode, or mixed push-pull scheme. To avoid redundant chunks incurred in push mode, we adopt the pull mode for *chunk scheduling* (i.e., which chunks to request from the neighborhood?) in which peers directly request chunks not within their own buffers but available in the neighborhood, and forward chunks to the requester upon receiving chunk requests. Frequently, several chunk holders may coexist in the neighborhood and neighbor selection is essential to determine to which to send the chunk requests. To avoid confusion, we term this neighbor selection as *peer-tier neighbor selection* and the neighbor selection taken by trackers for overlay construction as *tracker-tier neighbor selection*.

Therefore, in the design of locality aware strategies, we should focus on three tiers: tracker-tier neighbor selection for overlay construction, peer-tier neighbor selection and chunk scheduling.

III. Three Tier Framework: Overlay Construction, Neighbor Selection and Chunk Scheduling

In this section, we introduce tree tiers of strategies – tracker-tier neighbor selection, peer-tier neighbor selection and chunk scheduling – aiming at a thorough study on the impact of various capacity aware and locality aware strategies. Moreover, we propose a new capacity aware overlay construction strategy and a novel hybrid neighbor selection strategy.

A. Tracker-Tier Neighbor Selection

We first define strategies taken by trackers to recommend a list of neighbors for newly joining peers. After bootstrapping, only if there are not enough neighbors, will peers further contact the tracker and harvest some brand new neighbors.

1) Baseline Neighbor Selection – Random Neighbor Selection: As a baseline approach, random neighbor selection formulates a random mesh in overlay construction by randomly recommending M peers, to which peers can connect as neighbors. In our design, we limit the number of neighbors to M.

2) Pure Locality Aware Neighbor Selection: Trackers forward a list of neighbors with the smallest AS hop count (e.g., peers within the same AS) to newly joining peers or peers further contacting trackers due to the lack of neighbors to maintain enough streaming quality. Of course, in this case, we assume that trackers possesses the perfect information about AS hop count between any pair of peers. An obvious drawback of this method is the centralized maintenance of locality information on the tracker side.

3) Pure Capacity Aware Neighbor Selection: As indicated in [7], [8], if peers with high bandwidth capacity are placed closer to the source than peers with lower capacity, streaming performance can be improved in structured overlay topologies. Inspired by this, we propose the following pure capacity aware neighbor selection strategy: When peer i contacts trackers, trackers randomly select M peers from the set of concurrently online peers with capacity no smaller than peer i. Actually, this avoids the overload and cluster of high-capacity peers by also recommending peers with similar capacities and at the same time leverages the strength obtained from placing lower-capacity peers farther from the source.

4) Biased Neighbor Selection: Biased neighbor selection selects most peers from the same AS and others from different ASes. It is well-known that biased neighbor selection can mitigate the impact of clustering on system performance and at the same time maintain a high locality performance [9].

5) Hybrid Neighbor Selection: To integrate the strength of both network locality and node capacity into tracker-tier neighbor selection, hybrid neighbor selection recommends neighbors with bandwidth aware probability p utilizing the above stated pure capacity aware neighbor selection, while selecting neighbors via the above defined pure locality aware neighbor selection strategy with probability $q = 1 - p$. That is, a subset of neighbors recommended by trackers is selected based on capacity, while the complement subset is determined based on locality. This is remarkably different from biased neighbor selection by capturing the tradeoff between node capacity and node locality.

B. Peer-Tier Neighbor Selection

Denote by N_i^c the set of i's neighbors possessing chunk $c \notin C_i$, namely $N_i^c \subseteq N_i$:

$$N_i^c = \{p|(p \in N_i) \wedge (c \in C_p)\},$$

where N_p is the neighbor set of peer p and C_p is the set of chunks in p's buffer. The task of peer-tier neighbor selection is to choose neighbor j from N_i^c to send chunk requests to.

1) Random Neighbor Selection: Peer i randomly selects $j \in N_i^c$ to request chunk c.

2) Pure Locality Aware Neighbor Selection: When determining from which to request a specific chunk, peers select the closest neighbor in terms of AS hop count.

3) Pure Capacity Aware Neighbor Selection: In this study, we utilize the following capacity aware strategy [10] to capture the class of neighbor selection strategies aiming at load balancing: peer i selects j with the maximum value of outbound bandwidth over sending queue size. That is,

$$\arg\max_{j \in N_i^c} \frac{O_j}{l_j},$$

where l_j is the sending queue length of peer j.

4) Hybrid Neighbor Selection: Similar to tracker-tier hybrid neighbor selection, peer-tier hybrid neighbor selection selects j with bandwidth aware probability p utilizing the above peer-tier pure capacity aware neighbor selection and employing the above peer-tier pure locality aware neighbor selection strategy with probability $q = 1 - p$.

C. Chunk Scheduling

We utilize typical peer i and typical chunk c to illustrate the following three strategies. Denote by C_N and C_i the set of chunks available in i's neighborhood and the set of chunks in i's buffer. In chunk scheduling, we aim to select a chunk $c \in C_N \setminus C_i$ to request.

1) Random Scheduling: Peer i randomly selects chunk c.

2) Locality First Scheduling: Peer i selects chunk c with the lowest average AS hop count–the mean value of the AS hop counts of all i's neighbors holding chunk c.

3) Rarest First Scheduling: Peer i selects chunk c rarest in its neighborhood to promote chunk diversity.

In the simulation, we study the strength of the above algorithms with various strategy combinations. The default strategy for all three tiers is random, unless otherwise mentioned. Although various influencing design parameters (e.g., server bandwidth provision) exist for P2P multimedia streaming, in this study, we focus on the impact of the bandwidth aware probability p on system performance.

IV. The DNA Protocol

Tracker-tier neighbor selection needs to measure the routing hops between any pair of connected peers, which is prohibitively costly, even if feasible. This obviously deviates from our objective of scalability. Moreover, decentralization is a vital element of P2P design.

Motivated by this, our decentralized network awareness (DNA) protocol aims to solve this problem by adopting the random tracker-tier neighbor selection. Peers discover neighbors via trackers only for bootstrapping. After bootstrapping, each peer will autonomously adopt view exchange strategy to enhance one's neighborhood and promote network aware topology construction without querying trackers repeatedly. Each peer utilizes traceroute and a prefix-AS mapping table to obtain AS hop count to any connected neighbor [11]. The DNA protocol (cf. Fig. 1) incorporates an external membership management service, neighborhood refinement to further overlay topology construction, peer-tier neighbor selection and chunk scheduling into an integrated framework. Our protocol can inherently enforce network awareness in terms of either locality, capacity, or proximity. Due to page limit, we focus on locality enforcement in this study. Interested readers may refer to Xin Jin's doctoral dissertation in the future for the discussions on capacity and network proximity. In the following, we provide a detailed description of the architecture.

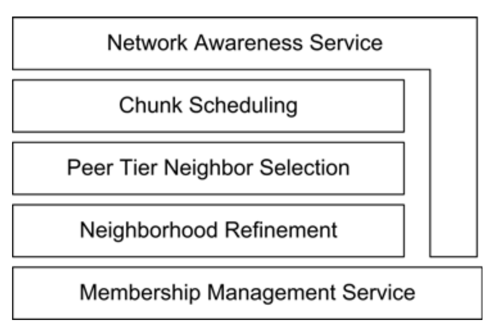


Fig. 1: DNA Architecture.

An external membership management service maintains all concurrently online peers on the tracker side. From simulation in Section VI, we learn that peer-tier neighbor selection and chunk scheduling cannot effectively enforce traffic locality. Thus, in our design, we adopt pure capacity aware peer-tier neighbor selection and rarest first chunk scheduling – which combination can improve streaming quality – and further enforce traffic locality by neighborhood refinement.

To promote network awareness, our DNA protocol incorporates neighborhood refinement, peer-tier neighbor selection and chunk scheduling. Different methods of peer-tier neighbor selection and chunk scheduling are discussed in the above section. Neighborhood refinement is composed of *view exchange* and *swap operation*. Peers exchange with each other knowledge of concurrently online peers to increase visibility towards the system. For swap operation, we first elect a candidate peer via a hybrid neighbor selection strategy and insert it into the neighbor set. If the neighbor set size exceeds the maximum number of neighbors that can be maintained in our protocol, an existing neighbor will be eliminated according to predefined strategies. The detailed DNA protocol is described in Alg. 1. In the following, we, respectively, describe neighbor view exchange and swap operation in detail.

Algorithm 1 DNA Protocol (peer i)

```
Internal data:
    nbr_list ⟵ ∅
    view ⟵ ∅
    init_nbr //Initial neighbor list selected randomly by the tracker

nbr_list ⟵ init_nbr
// hybrid neighbor selection
every △ T do
    if Size(view) ! = ∅ then
        p = bandwidth_aware_prob
        if Rand() < p then
            j ⟵ BestCapacityPeer(view)
        else
            j ⟵ BestLocalityPeer(view)
        Send(REQUEST(i),j)

// neighborhood update
upon Receive(REQUEST(j)) do
    if nbr_list ≥ NBR then
        if LOCALITYAWARE = true then
            e ⟵ WorstLocalityPeer(nbr_list)
        elseif CAPACITYAWARE = true then
            e ⟵ WorstCapacityPeer(nbr_list)
        eliminate(e)
    Insert(j,nbr_list)
    Send(ACCEPT(i), j)

upon Receive(ACCEPT(j)) do
    Insert(j,nbr_list)

// neighbor view exchange
every △ T do
    if Size(view) < N then
        j ⟵ random(nbr_list∪view)
        Send(REQUEST(NbrList,j,0))

upon Receive(REQUEST(NbrList,j,h) do
    nbr_exchg ⟵ nbr_list \ {j}
    Send(RETURN(nbr_exchg,i), j,h + 1)
    if h < 3 then
        Send(REQUEST(NbrList,j,h + 1))

upon Receive(RETURN(nbr_exchg,j,h))) do
    foreach k ∈ nbr_exchg
        if k ∉ nbr_list∪view
            Insert(k,view)
```

A. Neighbor View Exchange

In our DNA protocol, each peer maintains two peer sets: the neighbor set nbr_list and the known online peer set $\mathbb{V}$, called view ($nbr\_list \cap \mathbb{V} = \emptyset$). $\mathbb{V}$ enables peers to progressively refine the neighbor set. The purpose of neighbor view exchange is to periodically probe one's neighbors, harvest a list of online peers and consequently update its visibility towards the system.

The neighbor view exchange operation simply periodically sends neighbor list requests to randomly selected neighbors from peer set $nbr\_list \cup \mathbb{V}$ until obtaining a large enough view size. The request can be propagated to at most three hops in our implementation and each peer receiving requests returns a list of nodes to requesters. Obviously, this design is both lightweight and scalable.

B. Swap Operation

As described above, *candidate election* and *neighbor elimination* constitute the swap operation of our DNA protocol. For candidate election, we adopt a hybrid neighbor selection, similar to the tracker-tier hybrid neighbor selection except that candidate peers are selected from one's view set instead of all concurrently online peers maintained on the tracker side. We define two strategies for neighbor elimination: locality aware (literally eliminating the one with the largest AS hop count) and capacity aware (literally eliminating the one with the smallest outbound bandwidth). If tie exists, we adopt random strategy to break the tie. Unless otherwise mentioned, we adopt the locality aware strategy in our DNA protocol.

V. Performance Evaluation

We now proceed to evaluate the above proposed strategies. In this section, we first describe our simulation setup, followed by performance metrics.

A. Simulation Setup

We explore and evaluate the neighbor selection and chunk scheduling strategies stated above through simulations, based on the P2P media streaming simulator originally developed by Zhang [12]. This is an event-driven packet-level simulator with the capacity to simulate a maximum of $10,000$ peers simultaneously joining the system. In the following, we describe the simulation scenarios in detail.

Underlay topology. We modify the simulator to simulate the multi-AS underlay network with different topologies: star, mesh (completely connected), chain/line and ring. ASes interconnect with each other via inter-AS links. The star topology has one transit-AS and a number of stub-ASes, which are all connected with the transit-AS but not directly connected with each other [13]. ASes in the mesh topology are all directly connected with each other. ASes in the ring topology are all sequentially connected with each other according to the AS ranks. The chain/line topology is different from the ring topology only with the respect that the ASes with the lowest and highest ranks are not connected with each other.

Peer heterogeneity. We utilize the 3-class scenario to build an overlay upon a heterogeneous network. The 3-class scenario is based on a 3-class bandwidth distribution: low-bandwidth peers with outbound bandwidth 128 Kbps and inbound bandwidth 768 Kbps, medium-bandwidth peers with outbound bandwidth 384 Kbps and inbound bandwidth $1,500$ Kbps, and high-bandwidth peers with outbound bandwidth $1,000$ Kbps and inbound bandwidth $3,000$ Kbps.

Resource index. Denote by $\mathcal{P}$ the set of all online peers. O_s and O_p are respectively the outbound bandwidth of the streaming server and peer $p \in \mathcal{P}$. Outbound bandwidth supply is essential to the system performance. We define resource index ρ to evaluate outbound bandwidth supply in the system [14]:

$$\rho = \frac{O_s + \sum_{p \in \mathcal{P}} O_p}{|\mathcal{P}| \cdot r},$$

where r is the channel streaming rate. In the study of pure tracker-tier and peer-tier neighbor selection strategies, We evaluate the performance under different resource indices by varying the fractions of the 3-class peers defined above.

Peer distributions among ASes. In the uniform distribution, peer populations residing in different ASes are the same. However, peer distribution follows a Mandelbrot-Zipf (MZipf) distribution in real-deployed systems [15]. Thus, we utilize MZipf distribution with $\alpha = 1.33$ and $q = 10$ to simulate the skewed peer distribution among ASes:

$$p(k) = \frac{1/(k+q)^{\alpha}}{\sum_{k=1}^{k=K} 1/(k+q)^{\alpha}},$$

which defines the probability of each peer distributed within the AS with rank k. K is the total number of ASes.

Bandwidth distributions. Here, by bandwidth distribution, we mean the bandwidth distribution of peers within different ASes. We consider two scenarios: all high capacity peers are distributed among ASes with top AS ranks (top 6 ASes) by mimicking the fact that some ISPs enjoy technological advantage and offer their customers higher access speeds (*skewed bandwidth distribution scenario*); and peers with high and low access bandwidth capacities are equally distributed among various ASes (*uniform bandwidth distribution scenario*). Since the skewed bandwidth distribution scenario captures the realistic situation to a greater extent, unless otherwise mentioned, we utilize skewed bandwidth distribution scenario.

Peer dynamics. To evaluate the performance of our DNA protocol under network dynamics, we implement the following peer dynamics scenario: peers join and leave the overlay repeatedly with the peer arrival process following Poisson process. The user arrival rate is 10 peers per second and the expected lifetime is 15 mins.

Unless otherwise mentioned, 400 nodes – constituting an overlay with the resource index of 1.2 and uniformly distributed among ASes with a skewed bandwidth distribution – simultaneously participate in a static overlay network with 20 ASes, formulating a star underlay AS topology with an average node degree of 15.

B. Performance Metrics

Performance metrics are either network awareness oriented or QoS oriented. To evaluate the streaming performance in terms of QoS, we utilize the following metrics:

- **Distribution delay.** By distribution delay, we mean the elapsed time from the instant a chunk is generated by the source to the moment it is received by a peer. The average distribution delay of peer p is the average distribution delay of all chunks received by p. The average distribution delay of peer set $\mathbb{S}$ is the average delay of all peers within $\mathbb{S}$.
- **Delivery ratio.** Aside from distribution delay, to quantitatively evaluate the streaming quality, we define delivery ratio to represent the number of chunks arriving at each node before or on playback deadline over the total number of chunks that each node should receive.Then, the average delivery ratio of the system is defined as

$$\rho = \frac{\sum_{p \in \mathcal{P}} |\mathcal{C}_p|/|\mathcal{C}|}{|\mathcal{P}|},$$

where $\mathcal{C}_p$ and $\mathcal{C}$ respectively represent the set of chunks received by peer $p \in \mathcal{P}$ by playback deadline and the set of chunks generated by the source node.

On the other hand, to evaluate the graph stability and traffic locality, we define the following metrics:

- **Clustering coefficient.** "Clustering Coefficient" (CC) can be utilized to evaluate the stability and the randomness of self-organizing systems [16]. CC has become an important metric to study whether a random graph or "small-world" network is formulated in P2P overlay construction [17]. In this study, we utilize CC to investigate the effects of neighbor selection strategies on the randomness of the overlay topology. Denote by C_g and C_r the CC of the overlay graph formulated and the random graph respectively. To determine whether the graph is a small world or a random graph, we need to compare C_g and C_r. If C_g is orders of magnitude larger than C_r, the graph is a small world; otherwise, it is an indication of some graph randomness. Therefore, we utilize C_g/C_r to evaluate the overlay graph randomness. Specifically, we calculate CC in the manner as indicated in [17]. Firstly, for each online node i, we randomly select two neighbors j and k of i, and verify whether j is the neighbor of k and vice versa. If so, Cnt, initialized as 0, is increased by 1. Then,

$$C_g = (Cnt)/(2 * |\mathcal{P}|),$$

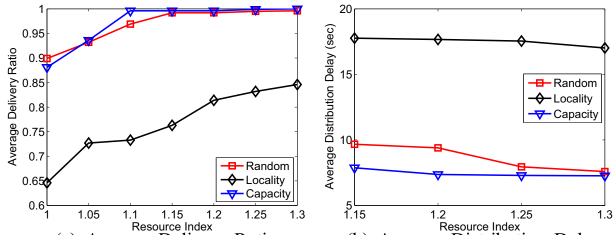
where $\mathcal{P}$ is the set of all peers within the channel we study. Secondly, C_r is calculated as

$$C_r = NBR/|\mathcal{P}|,$$

where NBR is the average node degree.
- **Inter-AS traffic.** In the course of alleviating the tension between ISPs and P2P content providers, we utilize inter-AS traffic to evaluate the efficacy of traffic locality enforcement. If the source and destination of a chunk share the same AS domain, it is counted as intra-AS traffic. Otherwise, it is deemed as inter-AS traffic.

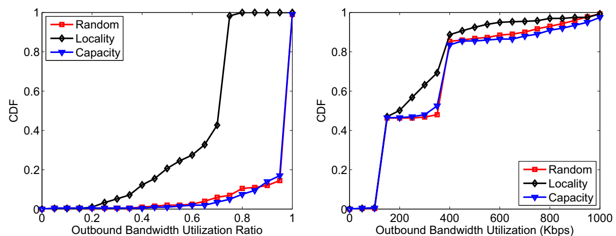
Outbound bandwidth is widely assumed as the only bottleneck for video streaming. To evaluate the efficiency of bandwidth resource utilization, we define the following two metrics:

- **Outbound bandwidth utilization ratio.** Outbound bandwidth utilization ratio of peer p is defined as the average



(a) Average Delivery Ratio (b) Average Distribution Delay

Fig. 2: Comparison of Tracker Tier Strategies.



(a) Bandwidth Utilization Ratio (b) Absolute Bandwidth Utilization

Fig. 3: Outbound Bandwidth Utilization.

upload rate to its neighbors over its outbound bandwidth capacity.

- **Signaling overhead.** Periodical control packets (i.e., packets other than content traffic) are inevitable for overlay construction and chunk scheduling. In this context, signaling overhead is calculated as the total control traffic over the total traffic existing in the overlay. Low overhead is preferred for more efficient outbound bandwidth utilization and better system design.

VI. Simulation Results

In this section, we present simulation results of the above proposed neighbor selection and chunk scheduling strategies.

A. Tracker Tier Neighbor Selection

We first compare the streaming and locality performance of the above defined tracker-tier neighbor selection strategies.

1) Pure Strategies: Fig. 2 shows the performance comparison of random, pure capacity aware and pure locality aware tracker-tier neighbor selection strategies. The results (cf. Fig. 2(a)) show that the capacity aware strategy can achieve better performance in terms of average delivery ratio. Since only when the resource index is above 1.1, can the random and pure capacity aware strategies achieve an acceptable streaming quality, we merely compare the distribution delay with resource indices above 1.1 in Fig. 2(b). Obviously, the capacity aware strategy outperforms the other two strategies with the highest streaming quality.

However, what is the source of the performance disparity? Fig.3 shows that the performance degradation of the locality aware strategy is due to the inefficiency of the bandwidth utilization. However, compared to the random strategy, the capacity aware strategy does not significantly pose more load on peers, which points to the dissemination efficiency of the overlay formulated by pure capacity aware strategy.

Further, what results in the low bandwidth utilization of the pure locality aware strategy? To verify our conjecture that the locality aware overlay clusters peers within the same AS and hinders the dissemination of chunks, we compare the average C_g/C_r of the three tracker-tier neighbor selection strategies in Table I. We learn that the clustering effect of locality aware overlay is much greater than the overlays formulated by the other two strategies, the inter-AS traffic, though, can be significantly reduced.

TABLE I: Tracker Tier Strategies

Tracker Tier Strategy	Average C_g/C_r	Inter-AS Traffic
Random	0.9329	95%
Capacity	1.2186	95%
Locality	16.89	9%

TABLE II: Channel Population (Pure Tracker Tier Locality)

Channel Population	Inter-AS Traffic	Average C_g/C_r	Average Delivery Ratio
50	92%	0.98	99.8%
100	78%	1.77	99.8%
150	62%	3.38	98.4%
200	50%	5.09	98.1%
250	39%	8.41	92.8%
300	17%	13.53	89.2%
350	12%	16.30	84.9%
400	9%	16.89	81.4%

To study which chunk scheduling strategy couples best with the pure locality aware strategy, we compare the performance of the three chunk scheduling methods under the resource index of 1.2. The inter-AS traffic is almost the same, but the average distribution delays are respectively $15672ms$ for rarest first strategy, $17063ms$ for locality first strategy, and $17663ms$ for random strategy. Obviously, rarest first strategy couples better with locality aware neighbor selection strategies.

Then, we study the impact of channel population on pure locality aware strategy in Table II. We find that it is much more difficult to enforce inter-AS traffic locality for unpopular channels and that the enforcement of traffic locality counters our goal of better streaming quality due to the clustering of peers.

2) Hybrid Neighbor Selection: We vary the bandwidth aware probability of the hybrid neighbor selection strategy to obtain performance evaluation in terms of streaming quality, traffic locality and clustering effects. Fig. 4 shows that the tracker-tier hybrid neighbor selection cannot achieve a near-optimal performance by coupling with peer-tier random neighbor selection and random chunk scheduling, although the streaming quality improves with the increase of capacity aware peers. Thus, in Section VI-C, we will discuss the coupling of the three-tier strategies.

Fig. 4 also presents system designers hints on how to choose p for neighbor selection. Specifically, two kinds of system designers may exist: locality aware designers to strike a balance between traffic locality and streaming quality, and quality aware designers only caring about streaming performance. Thus, a locality aware designer may choose $p = 0.4$ as the sweet spot because lower p will degrades the system performance and higher p will increase the inter-AS traffic with no much quality gain. However, a quality aware designer may always choose $p = 1.0$ because (s)he does not care about traffic locality at all.

Fig. 5 shows the inverse relationship between C_g/C_r and

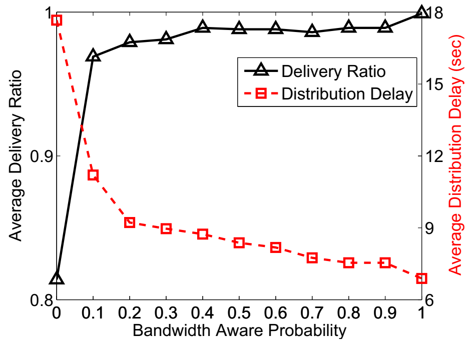


Fig. 4: Streaming Quality of Tracker Tier Hybrid Neighbor Selection.

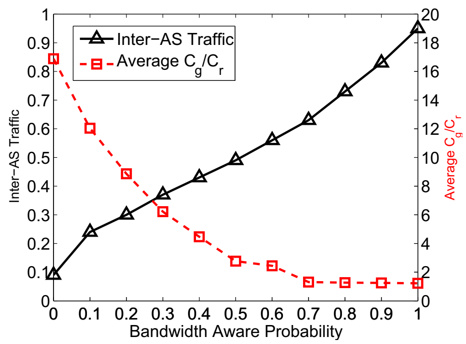


Fig. 5: Traffic Locality and Clustering Effects of Tracker Tier Hybrid Neighbor Selection.

inter-AS traffic. However, streaming quality increases with the decrease of C_g/C_r. To this end, we learn that capacity awareness can boost streaming quality, while locality awareness can enforce traffic locality but with low QoS performance. Thus, we needs to make a tradeoff between capacity awareness and locality awareness, which indicates the necessity of hybrid neighbor selection.

To further evaluate the performance of our proposed hybrid neighbor selection strategy for traffic locality enforcement, we also simulate the *biased neighbor selection* strategy. The biased neighbor selection strategy by selecting peers outside one's residential AS with probability 0.1 produces an inter-AS traffic of 26% and $C_g/C_r = 9.73$, while the hybrid neighbor selection strategy with bandwidth aware probability of 0.1 promotes an inter-AS traffic of 24% and $C_g/C_r = 12.04$. This demonstrates that our hybrid neighbor selection strategy can obtain comparable locality performance with the biased neighbor selection strategy.

B. Peer Tier Neighbor Selection

Here, we compare diverse peer-tier neighbor selection strategies. Because peer-tier neighbor selection does not affect the overlay topology and only decides to which peer to send the request, there is no need to study the metric C_g/C_r.

1) Pure Strategies: Fig. 6 shows us that, again, pure capacity aware strategy obtains the best streaming quality. Since pure locality aware strategy cannot achieve an acceptable delivery ratio, there is no need to study the distribution delay to show its inferiority in terms of streaming quality. With

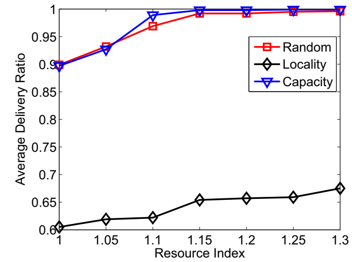
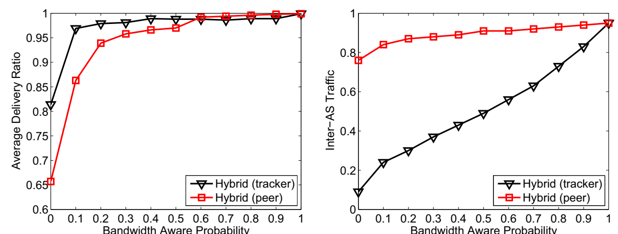


Fig. 6: Average Delivery Ratio of Pure Peer Tier Strategies.



(a) Streaming Quality (b) Traffic Locality

Fig. 7: Peer Tier Hybrid Neighbor Selection.

respect to traffic locality, both random and pure capacity aware strategies produce an inter-AS traffic of 95%, while pure locality aware strategy promotes an inter-AS traffic of 76%, which improvement is not as significant as the tracker-tier pure locality aware strategy.

2) Hybrid Neighbor Selection: Fig. 7 shows the simulation results with tracker-tier hybrid neighbor selection as a comparison. This shows peer-tier neighbor selection can enforce locality but not as effective as the tracker-tier neighbor selection. Therefore, for an effective locality enforcement, it is necessary to adopt active overlay construction (e.g., tracker-tier neighbor selection), instead of peer-tier neighbor selection – an extremely challenging task if without the aid of tracker. This is one design goal of our DNA protocol.

C. Chunk Scheduling

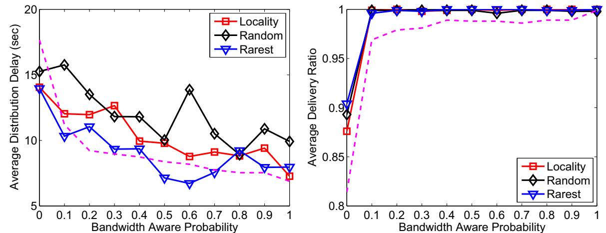
Since chunk scheduling does not affect the overlay topology, we only study the performance in terms of streaming quality and traffic locality.

1) Pure Strategies: In this part, the tracker-tier and peer-tier neighbor selection both adopt random strategies to compare various chunk scheduling strategies. The average distribution delays are respectively $9394ms$ for random strategy, $7604ms$ for rarest first strategy, and $9792ms$ for locality first strategy. All can achieve near optimal delivery ratio (no less than 0.99) with inter-AS traffic 95%.

We learn that locality first chunk scheduling cannot significantly enhance the performance in terms of locality. This again implies that the crux of locality aware scheme design is active overlay construction, instead of chunk scheduling strategies. At the same time, rarest first strategy performs best in terms of QoS.

2) Neighbor Selection and Chunk Scheduling Coupling: From the above analysis, we learn that tracker-tier pure capacity aware, peer-tier random neighbor selection and random chunk scheduling combination cannot achieve a near optimal streaming quality, and that the crux of locality enforcement is

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE P2P 2011 proceedings.



(a) Distribution Delay (b) Delivery Ratio

Fig. 8: Neighbor Selection and Chunk Scheduling Coupling.

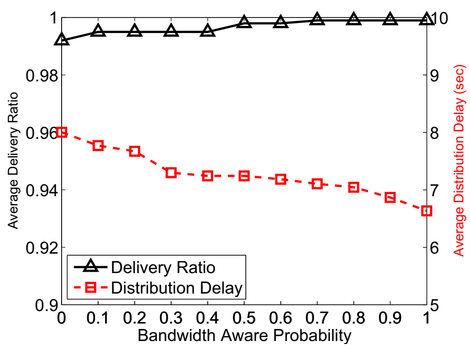


Fig. 9: Streaming Quality of The DNA Protocol.

active overlay construction by tracker-tier neighbor selection. Also, peer-tier capacity aware strategy can lift streaming quality. Thus, we vary chunk scheduling methods, to study whether rarest first chunk scheduling still performs best under tracker-tier hybrid and peer-tier pure capacity aware neighbor selection combination.

Fig. 8 shows the simulation results. The dashed line repeats results of tracker-tier pure capacity aware, peer-tier random neighbor selection and random chunk scheduling combination. The inter-AS traffic is almost the same with the tracker-tier hybrid, peer-tier random neighbor selection and random chunk scheduling combination under different chunk scheduling scenarios. We can easily observe that the rarest first chunk scheduling performs best despite the chaos of the distribution delay. This is why we adopt rarest first chunk scheduling in our DNA protocol design.

Surprisingly, when the bandwidth aware probability is 0.6 with rarest first chunk scheduling, we can achieve better performance in terms of both traffic locality and streaming quality than the tracker-tier pure capacity aware strategy.

D. The DNA Protocol

In this section, unless otherwise mentioned, the DNA protocol adopts locality aware neighbor elimination strategy with view size 50. The simulation is performed by varying the bandwidth aware probability of hybrid neighbor selection utilized for candidate election. From results shown in Fig. 9 and Fig. 10, we can easily observe the impact of bandwidth aware probability on both streaming quality and traffic locality.

The impact of view size. We vary the view size from 10 to 90 and hold constant bandwidth aware probability of 0. As shown in Fig. 11(a) and Table III, with the increase of view size, streaming quality degrades, while inter-AS traffic decreases. This is due to the increasing visibility and the

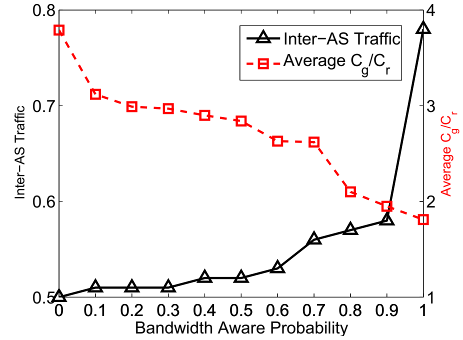
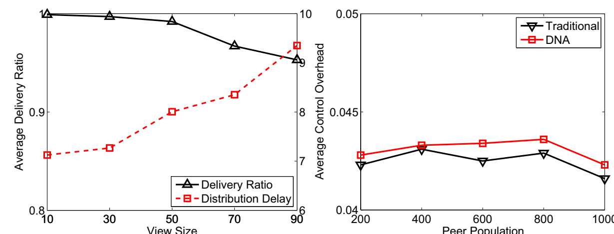


Fig. 10: Traffic Locality and Clustering Effects of The DNA Protocol.



(a) Impact of View Size (b) Control Message Overhead

Fig. 11: Impact of View Size and Control Message Overhead.

consequent clustering effects. Obviously, the view size of 50 makes a fair tradeoff between streaming quality and traffic locality enforcement. This is why we utilize this view size as the default setup.

TABLE III: Impact of View Size (DNA Protocol)

$\|view\|$	Average Inter-AS Traffic	Average C_g/C_r
10	78%	1.72
30	61%	2.26
50	50%	3.79
70	43%	4.72
90	35%	7.82

Signaling overhead. Because the traceroute messages grow linearly with the number of nodes maintained by each peer, the overhead we study here does not include traceroute messages. We term the strategies other than DNA protocol as traditional and compare the signaling overhead under different node population scenarios. As shown in Fig. 11(b), there is no significant performance degradation in terms of overhead. The increase in overhead mainly stems from view exchange messages and connection requests generated by neighborhood refinement.

Peer dynamics. Here, we study the effect of peer dynamics on our DNA protocol with bandwidth aware probability of 0 and performing neighbor elimination with locality awareness. The setup for peer dynamics is defined in Section V. We can still achieve a near optimal average delivery ratio of 0.995 but the inter-AS traffic is increased to 66%.

The impact of neighbor elimination strategy. If the neighbor elimination is capacity aware by eliminating neighbors with lowest outbound bandwidth, we can achieve a near optimal average delivery ratio of 0.999 with an average distribution delay of $6778ms$. However, the inter-AS traffic is

TABLE IV: Locality Promoting Peer Fractions

Locality Fraction	Average Inter-AS Traffic	Average C_g/C_r	Distribution Delay (ms)
0.0	79%	1.48	7303
0.1	75%	1.67	7835
0.2	70%	1.76	7071
0.3	68%	1.93	7461
0.4	65%	2.00	6780
0.5	63%	2.19	6973
0.6	62%	2.27	7363
0.7	58%	2.43	7397
0.8	56%	2.61	7336
0.9	52%	2.96	7945
1.0	50%	3.79	8004

65%, showing that it is more difficult to enforce traffic locality due to the de-clustering effect (C_g/C_r is decreased to 1.31).

The impact of pure network awareness. From simulation results of both tracker-tier pure locality aware strategy and our DNA protocol, we can easily learn that locality awareness ($p = 0$) hinders chunk dissemination due to significant clustering effects. Then, what if peers are capacity aware by favoring neighbors with high capacity? That is, the DNA protocol performs candidate election with bandwidth aware probability of 1 and employs capacity aware neighbor elimination. The inter-AS traffic is 95%, with almost no traffic locality.

Capacity awareness also invokes a thought on fairness and efficiency [18]. The overall average delivery ratio is 0.945 and the average delivery ratio of low capacity peers, medium capacity peers and high capacity peers are respectively 0.878, 0.982 and 0.995. We also find that low and medium capacity peers contribute almost all their outbound bandwidth with an average outbound bandwidth utilization ratio of 0.999, while the average outbound bandwidth utilization ratio of high capacity peers is 0.612 (still highest absolute bandwidth contribution among 3-class peers). This is a fair situation if we evaluate fairness utilizing absolute bandwidth contribution, yet, an unfair result in terms of effort based contribution (i.e., relative contribution). This also reveals that it is better to enforce effort based fairness to boost system efficiency.

The fact that C_g/C_r is 1.97 further points that the performance degradation stems from the clustering of peers with similar capacities – the intrinsic characteristic of the strategies taken by each peer.

The fraction of locality promoting peers. Next, we assume that two kinds of peers coexist in the overlay. Peers with bandwidth aware probability of 1 and locality promoting peers adopting our DNA protocol with bandwidth aware probability of 0. Both groups perform neighbor elimination with locality awareness, considering the fact that network aware peers defined above cannot achieve a reasonable streaming quality. We vary the fraction of locality promoting peers and obtain the simulation results shown in Table IV. In all scenarios, we can achieve average delivery ratio around 0.995.

Different AS topologies. Here, we study the performance of our DNA protocol under various AS topologies. The performance in terms of distribution delay, C_g/C_r and inter-AS traffic is different as shown in Table V. We can easily see that if the underlay topology (e.g., star and mesh) is well-connected, efficient chunk dissemination can be attained.

TABLE V: AS Topologies

Underlay Topology	Average Inter-AS Traffic	Average C_g/C_r	Distribution Delay (ms)
Star	50%	3.79	8004
Mesh	50%	4.06	7866
Chain	61%	6.06	13722
Ring	62%	5.95	13481

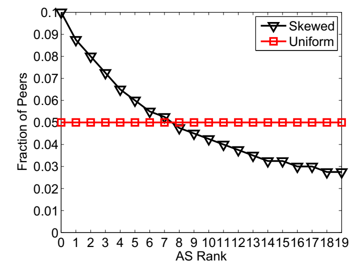


Fig. 12: Peer Distributions among ASes.

However, if the network is highly clustered with isolated “domains” as in the chain and ring topologies, it will incur longer delay for chunks to be distributed across the network.

Skewed peer distribution. Here, we study the impact of MZipf peer distribution among ASes, as shown in Fig. 12, but with a uniform bandwidth distribution.

For DNA protocol with bandwidth aware probability of 0, we cannot achieve a near optimal average delivery ratio (0.982). With bandwidth aware probability of 0.1, we can achieve an average delivery ratio of 0.996, an average distribution delay of $6752ms$, an inter-AS traffic of 51% and $C_g/C_r = 3.93$. To evaluate our DNA protocol, we also study the tracker-tier neighbor selection strategy under this scenario as a comparison.

Surprisingly, we cannot achieve a near optimal delivery ratio until the bandwidth aware probability is 0.3. With bandwidth aware probability of 0.3, we can achieve an average delivery ratio of 0.999, an average distribution delay of $8675ms$, an inter-AS traffic of 35% and $C_g/C_r = 6.02$.

Finally, we raise a question: Can our DNA protocol achieve a comparable traffic locality performance with the tracker-tier strategy? The answer is yes. When the view size is 90, with bandwidth aware probability of 0.4, we can achieve an average delivery ratio of 0.996, an average distribution delay of $6936ms$, an inter-AS traffic of 38% and $C_g/C_r = 5.07$.

VII. Related Work

Several alternatives exist for ISPs to reduce inter-ISP traffic, including bandwidth throttling, gateway peers or caches deployment. Bandwidth throttling limits inter-domain traffic or even closes inter-ISP connections (e.g., Comcast [19]) with the sacrifice of user satisfaction. Gateway peers and caches store blocks sent by external peers and redistribute them when requested by internal peers instead of fetching blocks repeatedly from external peers. Obviously, both gateway peers and caches face the problem of scalability.

R. Bindal *et al.* [9] propose biased neighbor selection (BNS) to improve traffic locality in BitTorrent (BT) via topology construction, in which the majority of one’s neighbors are from those within the same ISP, while retaining k neighbors from other ISPs. In [20], a locality aware peer selection strategy redirects new joining peers to a selected set of peers, with the assistance of so-called "oracle" – a centralized service, offered by the ISP and ranking potential neighbors according to preferential metrics. P4P [21] proposes to create an iTracker to recommend peer lists. Ono plugin [22] guides the recommendation of peers in the neighbor set, utilizing the similarity of the redirection ratio of CDN servers without any centralized entity, which is different from the above strategies.

B. Liu *et al.* [5] go further to construct an autonomous system (AS) level map via measurement study. To promote traffic locality, the random neighbor selection is replaced with AS hop count metric minimization (tracker locality), and a choker locality strategy which unchokes the four neighbors with minimum AS hop count. Accordingly, a locality first piece selection strategy favoring pieces with lower distance is in place. In reality, these locality aware policies are in accordance to BNS.

In [23], an approach very similar to BNS is investigated and shows that BitTorrent locality enforced with the neighbor set containing almost only local peers enables significant saving on inter-ISP traffic without degrading peer download completion time. As a complementary mechanism to BNS, biased unchoking (BU) [24] divides the neighbor set into two subsets according to locality value and chooses optimistically choked peers from the subset of preferred peers with probability q and with probability $1 - q$ from the other subset. In this literature, the simulation is performed with a star AS underlay topology.

In contrast to the above mentioned works, our study on unstructured live streaming overlays with epidemic protocol is novel by balancing the tradeoff between capacity awareness and traffic locality. Moreover, we propose a totally decentralized network awareness protocol.

VIII. Conclusions and Future Work

We have presented a new three tier framework, namely tracker-tier neighbor selection, peer-tier neighbor selection, and chunk scheduling, to thoroughly study the impact of capacity awareness and network locality. Specifically, diverse capacity aware and locality aware strategies are proposed to fit in this framework. Then, a novel decentralized network awareness protocol is proposed to promote traffic locality without the assistance of any centralized service.

In this paper, extensive simulations are conducted to evaluate various three-tier strategies and our DNA protocol. We arrive at several insightful conclusions. Firstly, the crux of traffic locality is active overlay construction, instead of peer-tier neighbor selection or chunk scheduling. Secondly, both pure locality and capacity awareness hinder efficient chunk dissemination due to clustering effects. Moreover, the promotion of traffic locality and capacity awareness is conflicting. We can break the tradeoff between streaming quality and traffic locality utilizing our novel hybrid neighbor selection strategy. Finally, our DNA protocol can achieve comparable locality performance under realistic simulation environments.

Our study still faces two constrains. From the evaluation perspective, simulations cannot capture all dynamic factors in a realistic network. Thus, an on-field experiment on PlanetLab to obtain a deeper understanding of the tradeoff between node capacity and node locality and the effectiveness of our traceroute assisted DNA protocol is in prospect. Also, game theoretic strategy interactions among trackers and peers offer another interesting research arena, considering their selfish behaviors. We are currently working on both directions.

References

[1] PPLive, 2011, http://www.pptv.com/.

[2] Skype, 2011, http://www.skype.com.

[3] PPStream, 2011, http://www.ppstream.com.

[4] D. Ciullo, M. A. Garcia, A. Horvath, E. Leonardi, M. Mellia, D. Rossi, M. Telek, and P. Veglia, "Network Awareness of P2P Live Streaming Applications: a Measurement Study," *IEEE Transactions on Multimedia*, vol. 12, no. 1, pp. 54–63, January 2010.

[5] B. Liu, Y. Cui, Y. Lu, and Y. Xue, "Locality-Awareness in BitTorrent-Like P2P Applications," *IEEE Transactions on Multimedia*, vol. 11, no. 3, pp. 361–371, April 2009.

[6] L. Zhao, J.-G. Luo, M. Zhang, W.-J. Fu, J. Luo, Y.-F. Zhang, and S.-Q. Yang, "Gridmedia: A Practical Peer-to-Peer Based Live Video Streaming System," in *Proc. of IEEE MMSP*, November 2005.

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

[8] J. Noh, P. Baccichet, A. Mavlankar, and B. Girod, "Un-Leeching P2P Streaming by Active Overlay Management," in *Proc. of IEEE GLOBECOM*, December 2008.

[9] R. Bindal, P. Cao, and W. Chan, "Improving Traffic Locality in BitTorrent via Biased Neighbor Selection," in *Proc. of IEEE ICDCS*, July 2006.

[10] Y. Yang, A. L. Chow, L. Golubchik, and D. Bragg, "Improving QoS in BitTorrent-like VoD Systems," in *Proc. of IEEE INFOCOM*, March 2010.

[11] S. Ren, E. Tan, T. Luo, S. Chen, L. Guo, and X. Zhang, "TopBT: A Topology-Aware and Infrastructure-Independent BitTorrent Client," in *Proc. of IEEE INFOCOM*, March 2010.

[12] M. Zhang, "Peer-to-Peer Streaming Simulator," 2011, http://media.cs.tsinghua.edu.cn/∼zhangm/.

[13] F. Lehrieder, S. Oechsner, T. Hoßfeld, Z. Despotovic, W. Kellerer, and M. Michel, "Can P2P-Users Benefit from Locality-Awareness?" in *Proc. of IEEE P2P*, August 2010.

[14] C. Liang, Y. Guo, and Y. Liu, "Investigating the Scheduling Sensitivity of P2P Video Streaming: An Experimental Study," *IEEE Transactions on Multimedia*, vol. 11, no. 3, pp. 348–360, April 2009.

[15] H. Wang, J. Liu, and K. Xu, "On the Locality of BitTorrent-based Video File Swarming," in *Proc. of USENIX IPTPS*, April 2009.

[16] D. J. Watts and S. H. Strogatz, "Collective Dynamics of 'Small-World' Networks," *Nature*, vol. 393, no. 6684, pp. 440–442, June 1998.

[17] L. Vu, I. Gupta, J. Liang, and K. Nahrstedt, "Measurement and Modeling of a Large-scale Overlay for Multimedia Streaming," in *Proc. of QShine*, August 2007.

[18] R. Rahman, M. Meulpolder, D. Hales, J. Pouwelse, D. Epema, and H. Sips, "Improving Efficiency and Fairness in P2P Systems with Effort-Based Incentives," in *Proc. of IEEE ICC*, May 2010.

[19] Comcast Throttles BitTorrent Traffic, 2007, http://torrentfreak.com/comcast-throttles-bittorrent-traffic-seeding-impossible/.

[20] V. Aggarwal, A. Feldmann, and C. Scheideler, "Can ISPs and P2P Users Cooperate for Improved Performance?" *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 3, pp. 29–40, July 2007.

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

[22] D. R. Choffnes and F. E. Bustamante, "Taming the Torrent: A Practical Approach to Reducing Cross-ISP Traffic in P2P Systems," in *Proc. of ACM SIGCOMM*, August 2008.

[23] S. L. Blond, A. Legout, and W. Dabbous, "Pushing BitTorrent Locality to the Limit," *Computer Networks*, vol. 55, no. 3, pp. 541–557, February 2011.

[24] S. Oechsner, F. Lehrieder, T. Hoßfeld, F. Metzger, D. Staehle, and K. Pussep, "Pushing the Performance of Biased Neighbor Selection through Biased Unchoking," in *Proc. of IEEE P2P*, September 2009.