# Router-aided Approach for P2P Traffic Localization

A DISSERTATION SUBMITTED TO THE
GRADUATE SCHOOL OF ENGINEERING AND SCIENCE OF
SHIBAURA INSTITUTE OF TECHNOLOGY

by

**HOANG VAN HIEP**

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

**DOCTOR OF ENGINEERING**

SEPTEMBER 2014

To my parents for their inspiration and motivation to meet my dreams.

To my wife, Pham Thi Nhai. Thanks to her love and sacrifice I can keep my mind on my research.

To my children, Hoang Dang Khoa and Hoang Mai Chi.

# Acknowledgments

First of all, I would like to express my sincere thanks to my supervisor, Professor Takumi Miyoshi for his valuable guidance and support throughout my research. I have learned from him many things related to my research direction. I have also learned from him many other things, especially the professional attitude on working. Every time I discuss with him, I can obtain a lot of knowledge. Thanks to his kindly support and many instructive comments at every stage of the dissertation process, I can finish my Ph.D dissertation sooner than expected.

Next, I would like to thank all my committee members, Prof. Yutaka Hirakawa, Prof. Eiji Kamioka, Prof. Hiroaki Morino, and Prof. Takuya Asaka for participating in my defense and their valuable comments. Their suggestions and discussions during my presentations help me to significantly improve my results. I wish to thank Prof. Yoshiaki Tanaka, Waseda University, for his suggestions and helpful comments in IEICE conferences, workshops, and summer camps.

I would also like to extend my gratitude to the faculty and staff members of Shibaura Institute of Technology for their support. Thanks to all the Multimedia Information Network Laboratory (MINET) members who are very kind to me, I strongly appreciate their helps for my life in Japan.

Finally, I would like to thank all my friends including academic and non-academic ones, who have always been by my side to overcome

even the most difficult time. Thank you all for your friendship and encouragement.

Saitama, Jun 1, 2014

Hoang Van Hiep

# Abstract

Most peer-to-peer (P2P) applications including file sharing and streaming applications form overlay networks for communicating among peers that are oblivious to the underlay network topology. As a result, a large quantity of unpredictable traffic is generated on the Internet. In particular, the unwanted cross-domain traffic proves to be costly for the ISPs. This raises the problem of P2P traffic localization.

ISPs or network operators often control P2P traffic by bandwidth throttling or limiting and/or even blocking P2P systems in their network. However, this is not an overall solution for the fundamental concern of the ISPs, which is to reduce the cross-ISP/AS traffic. A variety of methods have been introduced to solve the problem, and many works proposed that the consideration of peer location would reduce the cross-domain traffic and conserve the bandwidth. To realize traffic localization, P2P systems must be essentially equipped with locality-aware neighbor peer-selection mechanisms. Almost all of existing approaches, however, focus on solving the problem on the application layer. Several modifications of the existing P2P systems are therefore inevitable as one of the following reasons:

- The enhancement of trackers to efficiently gather information of the underlay network and to provide this information to the P2P applications. On the P2P application side, an appropriate protocol to communicate with the enhanced trackers must be implemented.

- The modification of the P2P application software to upgrade the current neighbor peer-selection procedures because P2P applications currently only employ random and/or round-trip time (RTT)-based strategies.
- Both of the above.

In this dissertation I propose a novel approach for P2P traffic localization without any peer reaction. The proposed approach requires neither dedicated servers, nor collaboration between ISPs and P2P users, nor modification of P2P application software. In particular, this dissertation offers the following main contributions.

First, I proposed a peer list modification method for traffic localization. The peer lists are modified for localizing before they arrive at the application. The experiments evaluating on a popular P2PTV, namely PPStream, prove the effectiveness of the proposed method on the problem of traffic localization.

Secondly, I proposed a video request packet redirection method for traffic localization. In this method, video data request packets that are sent to the peers not contained in the localized list are modified to redirect to peers in the localized list. Experimental results show that the method successfully realizes traffic localization on PPStream.

Thirdly, I proposed a novel method for localizing P2P traffic hierarchically with multiple levels including AS level, ISP level, and country level. The method is completely independent of P2P applications. The idea is that, if we intentionally degrade the quality of connection paths of inter-domain traffic, we can turn the inter-domain traffic into the intra-domain traffic since a querying peer will tend to remove the inter-domain connections and select the local connections instead. To achieve this idea, I proposed three different schemes including delay insertion, forcing packet loss, and bandwidth limitation. Experiments on different P2P streaming applications indicate that the hierarchical traffic localization method not only reduces significantly the inter-domain traffic but also maintains a good performance of P2P applications. This method is also the most significant contribution of this dissertation.

Finally, I proposed a router collaboration scheme to combine with the peer list modification scheme for traffic localization. In this method, the local peers are collected at not only one router but also many routers. Clearly, the peer list modification scheme will become much more effective in combining with router collaboration scheme because we will have more local peers in hand. This has been proven by experimental evaluation on PPStream.

# Contents

# List of Figures

# List of Tables

# List of Algorithms

# Chapter 1

# Introduction

This chapter starts out by describing the P2P traffic localization problem. Then, I discuss the specific challenges and why current approaches are not up to meeting these challenges, which will later be used to motivate my proposal. The end of this chapter presents the main contributions of this dissertation and an outline of its organization.

## 1.1 P2P Traffic Localization Problem

P2P file sharing traffic used to be the dominant portion of traffic on the Internet in the last decade. This situation has changed dramatically with the tremendous growth of multimedia content delivery, especially the increasing deployment of video streaming services in the last few years. It is reported that the sum of all forms of video including TV, video on demand (VoD), Internet, and P2P will be approximately 86 percent of global consumer traffic by 2016 [1]. In the video streaming field, P2P is still a promising model because it can distribute the transmission load on video servers across user terminals. Currently, P2P video streaming applications (P2PTV) such as PPTV [2], PPStream [3], SopCast [4], and Zattoo [5] have become increasingly popular. Figure 1.1 presents an example of neighbor peer distribution of a node running an on-demand video channel on PPStream. We can see that the peers are distributed in many places all over

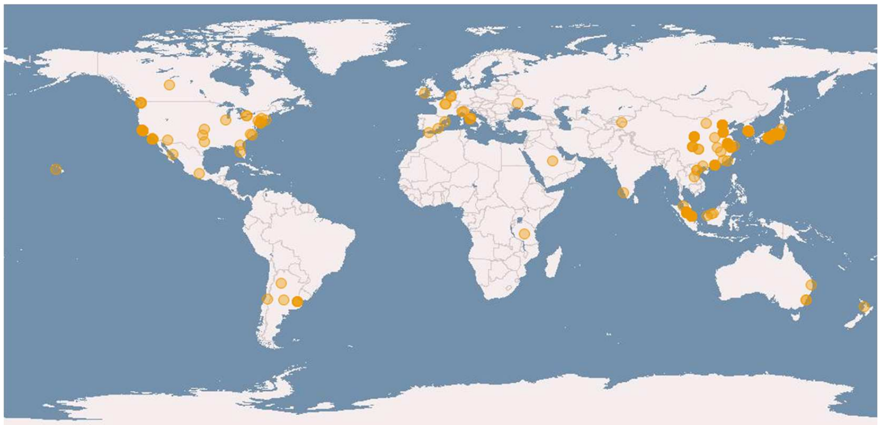


Figure 1.1: An example of peer distribution for PPStream.

the world. Therefore, controlling the traffic generated by P2P systems will be mandatory for internet service providers (ISPs) as well as the research community.

In P2P communications, routing functions for communicating among peers are implemented based on the overlay topologies built on top of the Internet. The problem is that the overlay networks are generally constructed without considering locality on the underlay network. Figure 1.2 shows an example of overlay network, in which peers establish their connections based on not the network stability but the resource availability. Therefore, although some peers are belonging to the same ISP in the underlay network, they are not neighbors on the overlay network. For this reason, P2P systems generate a large amount of unwanted traffic on the Internet. The unwanted inter-domain traffic is especially costly for the ISPs. This raises the problem of P2P traffic localization.

Figure 1.3 illustrates an example of the problem. In many cases, *Peer 2* located in Japan tries to connect to *Peer 3* in America to download the desired resource while the resource is also available at *Peer 1* located in the same ISP as *Peer 2*. This is because *Peer 2* lacks information about the underlay network topology. Clearly, it is better if a sender and a receiver are closer together since there are at least two advantages: the download time can be shorter, and the costs at ISPs can be saved due to reduction of traffic crossing various ISPs. The goal of traffic localization is to ask *Peer 2* to download the data from *Peer 1*, not

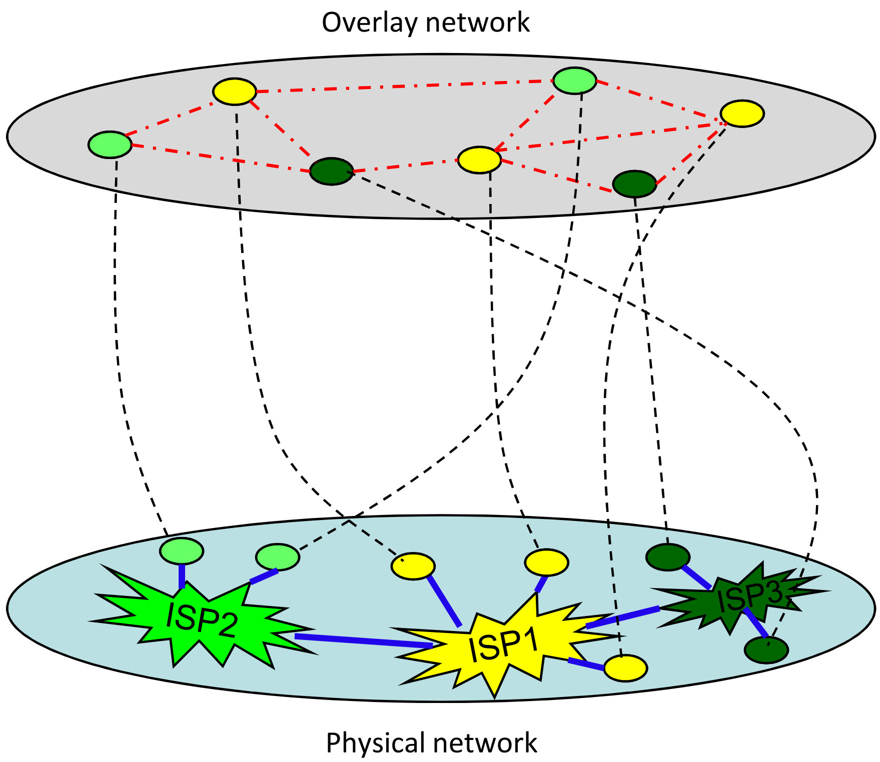



Figure 1.2: The overlay network is usually independent of physical network topology.

*Peer 3*.

## 1.2 Challenges

### 1.2.1 Various and Proprietary P2P Applications

There are various P2P applications, and they can be divided into two categories including file sharing and P2P streaming systems (P2PTV). In file sharing systems, BitTorrent is one of the most popular protocols for transferring large files. On the other hand, in the P2PTV systems, although they have been draw attention of ISPs, the internal algorithms and protocols are not public. The development of P2PTV systems bring uneasiness to ISPs since the traffic they generate may extremely grow out of control. Recently, one of the goals of the NAPA-

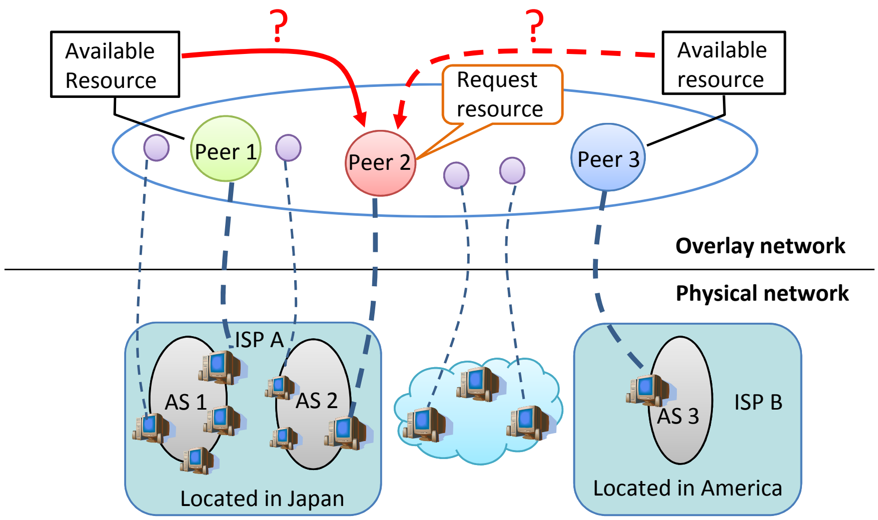



Figure 1.3: An example of P2P traffic localization problem.

WINE project (Network-Aware P2PTV Application over WIse NEt-works), is to provide a thorough analysis of the impact that a large deployment of P2PTV services may have on the Internet, through a detailed characterization of the traffic they generate [21].

Localizing P2PTV traffic may have the following difficulties: (1) reverse-engineering is required to investigate the protocols of P2PTV applications; (2) For almost existing locality-enhancing strategies, P2P systems must be essentially equipped with locality-aware neighbor peer selection mechanisms in order to realize P2P traffic localization. Therefore, several modifications of existing P2P application software are inevitable. This is sometimes very hard, even if not impossible, due to a closed design or proprietary problem of commercial software.

### 1.2.2 Relationship between ISPs and P2P applications

To reduce the cost of handling cross-ISP/AS traffic, ISPs might implement bandwidth throttling or limits, and/or even block P2P systems in their networks. However, this is not an overall solution, only a temporary fix. In response, P2P applications may change the design and try to hide from the network operators, e.g., applying dynamic port strategies. This makes P2P traffic control problem more challenging. Many previous works propose that ISPs and P2P users should cooperate with each other for improving the network efficiency as well as P2P application performance. However, this approach still has to face the following difficulties: (1) the need of dedicated servers or enhancement of trackers to efficiently gather information of the underlay network and to provide this information to the P2P applications. On the P2P application side, an appropriate protocol to communicate with the enhanced trackers must be implemented; (2) the need of trust and good cooperation between ISPs and P2P users.

### 1.2.3 Impact of Over Localization

There is a trade-off between the localization and P2P application performance. Thus, an excessive localization (over-localization) of the traffic might cause partitioning in the overlay interconnecting these peers, which will negatively affect the performance experienced by the peers themselves. Therefore, finding the reasonable balance between localization and randomness in neighbor peers selection is another issue.

Besides, many other issues of P2P systems remain such as security, privacy, etc. These issues are beyond the scope of this dissertation.

## 1.3 Objectives

Motivated by the challenges mentioned above, the main objectives of the dissertation are as follows:

- **Extensibility**. This dissertation aims to localize the P2P traffic without any modification of existing P2P application software. The proposed schemes should be easily applied for all types of P2P applications, especially for P2PTV services, which are predicted to be much more popular in the very near future.

- **User transparence**. The proposed method must ensure that it is completely transparent to the users. Therefore, it should not require any dedicated servers, or collaboration between ISPs and P2P users.

- **Win-no lose situation**. The proposed method must ensure a "win" situation for ISPs by reducing the cross-domain traffic, and a "no lose" situation for P2P users by maintaining a good performance of P2P applications.

- **Simplicity**. The proposed method should be easily introduced into the current network.

## 1.4 Contributions of this Dissertation

The overall objective of the schemes proposed in this dissertation is to localize the P2P traffic without requirement of dedicated servers, or collaboration between ISPs and P2P users, or modification of P2P application software. To achieve all the above requirements, I proposed router-aided approach for solving the problem. The contributions of the dissertation are as follows:

- First, I proposed a peer list modification method for traffic localization. The peer lists are modified for localizing before they arrive at the application. The experiments evaluating on a popular P2PTV, namely PPStream, prove the effectiveness of the proposed method on the problem of traffic localization.

- Secondly, I proposed a video request packet redirection method for traffic localization. In this method, video data request packets that are sent to the peers not contained in the localized list are modified to redirect to peers in the localized list. Experimental results show that the method successfully realizes traffic localization on PPStream.

- Thirdly, I proposed a novel method for localizing P2P traffic hierarchically with multiple levels including AS level, ISP level, and country level. The method is completely independent of P2P applications. The idea is that, if we intentionally degrade the quality of connection paths of inter-domain traffic, we can turn the inter-domain traffic into the intra-domain traffic since a querying peer will tend to remove the inter-domain connections and select the local connections instead. To achieve this idea, I proposed three different schemes including delay insertion, forcing packet loss, and bandwidth limitation. Experiments on different P2P streaming applications indicate that the hierarchical traffic localization method not only reduces significantly the inter-domain traffic but also maintains a good performance of P2P applications. This method is also the most significant contribution of this dissertation.

- Finally, I proposed a router collaboration scheme to combine with the peer list modification scheme for traffic localization. In this method, the local peers are collected at not only one router but also many routers. Clearly, the peer list modification scheme will become much more effective in combining with router collaboration scheme because we will have more local peers in hand. This has been proven by experimental evaluation on PPStream.

## 1.5 Structure of this Dissertation

This dissertation is composed of eight chapters. Chapter 1 provides the problem. Then, the specific challenges and a question why the current technology is not meeting these challenges are discussed.

Chapter 2 summaries previous works in four key areas related to the dissertation including P2P streaming systems, conventional approaches for P2P traffic management, application-layer traffic optimization problem, and router-aided approaches.

Chapter 3 describes the protocol of P2PTV in detail. Based on the observation of the protocol, I give solutions for the P2P traffic localization problem, which will be explained in latter chapters.

Chapter 4 introduces the peer list modification method. In this chapter, I focus on the step of obtaining the peer list of P2PTV. A router-aided scheme is proposed to modify all the peer lists for localizing before they arrive at the application.

Chapter 5 presents the video request packet redirection method. In this chapter, I focus on the step of sending video request packets of P2PTV. A router-aided scheme is proposed to redirect all request packets (sent to foreign peers) to the local peers.

Chapter 6 describes a router-aided method for localizing P2P traffic hierarchically with multiple levels such as AS level, ISP level, or country level. Three different schemes including hierarchical delay insertion scheme (HDIS), hierarchical forcing packet loss (HPLS), and hierarchical bandwidth limitation scheme (HBLS) are also described in detail in this chapter.

Chapter 7 proposes a router collaboration scheme to combine with the peer list modification scheme for traffic localization. This chapter is to demonstrate that router collaboration can make contribution on the problem of P2P traffic localization.

The dissertation ends with conclusions and future works in the chapter 8.

# Chapter 2

# Literature review

This chapter summarizes previous works in four key areas related to my research: (1) P2P streaming systems; (2) Conventional approaches for P2P traffic management; (3) Application-layer traffic optimization; and (4) Router-aided approaches.

## 2.1 P2P streaming systems

Existing P2P streaming systems can be largely classified into three categories including tree-based, mesh-based, and hybrid structures.

**(1) Tree-based structure**: peers are organized into a tree structure, rooted at the source of content. In principle, the content is "pushed" from higher levels to lower levels, i.e., each node only receives data from its parent node, which may be the source or a peer. The push-based content delivery is suitable for steady systems where peers do not change too often. Such systems require little overhead, since packets are forwarded from node to node without the need of extra signal messages. However, in high churn environments, the tree must be continuously destroyed and rebuilt. As a consequence, in order to avoid disruption, nodes have to buffer enough data for at least time required to reconstruct the tree. Therefore the tree-based structure has less reliability. A few examples of P2P applications such as PeerCast [6] and Conviva [7], follow the tree-based architecture.

**(2) Mesh-based structure**: in contrast to the tree-based structure, mesh-based systems implement a mesh distributed graph, where each node contacts a subset of peers to obtain a number of chunks. In the mesh-based structure, each node "pulls" the chunks it needs from other peers. This pull-based content delivery involves very high overhead due to the exchange of buffer maps between nodes (a buffer map message indicates which video chunks a peer has already buffered and can share with other peers). Since each node relies on multiple neighbors to retrieve the content, mesh-based systems offer good resilience to node failures, and thus have high reliability. Many popular P2P applications follow the mesh-based structure such as PPTV [2], PPStream [3], SopCast [4], TVants [8], Zattoo [5], etc.

**(3) Hybrid structure**: this structure combines the tree-based and mesh-based structures, that is, pull-based and push-based content delivery to use the advantages of both structures. In the hybrid structure, all the peers are organized into a mesh-based topology. The process of data delivery combining push and pull schemes is as follows: (1) a requested node first subscribes to a sub-stream by connecting to one of its partners via a single request (pull); (2) the selected partner, i.e., the parent node, will continue pushing all chunks in need of the sub-stream to the requested node. New Coolstreaming [34] is an application applying a hybrid structure.

In this dissertation, I focus on mesh-based P2P streaming applications because they are the most popular ones that are attracting millions of users. The general protocol of these applications will be described in chapter 3.

## 2.2 Conventional approaches for P2P traffic management

### 2.2.1 Over-provisioning

In a P2PTV application, a requested chunk must come into the application before the chunk deadline. If the chunk comes later than the deadline, it will be lost and the quality of P2PTV is therefore degraded. If bandwidth is insufficient, real-time traffic will suffer from congestion. Over-provisioning, or acquiring more bandwidth, is a straightforward way to avoid congestion on the Internet and solves the QoS issue. Clearly, it is more difficult to control a network that does not have enough bandwidth than a well-provisioning one. In addition, over-provisioning also leaves enough room for future traffic growth. In network planning with over-provisioning, the rules-of-thumb for backbone links is to upgrade at 40% or 50% of the link utilization and ensure that the maximum usage of the link does not exceed 75% under failure scenarios [47].

Considering the extremely increasing the amount of video traffic as reported by Cisco [1], bandwidth over-provisioning becomes mandatory. However, this method does not solve the fundamental concern of the ISPs, which is to reduce the cross-domain traffic, i.e., to localize the traffic.

### 2.2.2 Blocking P2P Traffic

Blocking P2P traffic is another way to avoid network performance degradation of non-P2P traffic. This method is to save bandwidth for other applications and to avoid legal problems for ISPs caused by illegal distribution of copyrighted contents via their networks. However, blocking P2P traffic is just suitable for college or university networks because of the following reasons: (1) P2P systems try to hide themselves from the network by changing their design, e.g., applying dynamic port strategies. It makes the recognition of the P2P traffic more challenging; (2) Many legal contents delivered by P2P systems such as copyright-free music, free movies, games, and Linux distributions, should not be blocked; (3) Blocking P2P systems would lead to the decrease of customer satisfaction, i.e., reduces sharply the demand of end users, and therefore reduces revenue and market share.

### 2.2.3 Bandwidth caps

As reported in [9], in North America, 1% of the subscriber population generates about 50% of upstream and 25% of downstream traffic. This means that ISPs may need to do something with such kind of heavy users to save the bandwidth for other users. Implementing bandwidth caps is one of the methods to discourage users from consuming excessive amount of bandwidth (the cap). For instance, if a user exceeds the bandwidth cap, the ISP can restrict the users connection speed for a certain time. This method may have effectiveness with P2P file sharing systems because almost all P2P file sharing systems are bandwidth hungry. However, for real-time streaming traffic such as game consoles and P2PTV, tracking the total bandwidth usage to avoid exceeding traffic quotas is not trivial task. This is because applying the bandwidth caps may cause many disruptions of the real-time traffic and thus trouble the customers. With the rapid increase of smartphones and tablets, bandwidth caps have become quite common in wireless networks.

### 2.2.4 Deep packet inspection (DPI)

Deep packet inspection (DPI) becomes a mandatory method for differentiating services in the Internet since examining just the packet headers and port numbers does not help for traffic classification anymore. Techniques for DPI include scanning a specific string in the header or payload, behavioral analysis, statistical analysis, etc. This approach avoids the drawback of blocking all P2P traffic. As a consequence, it is widely used by ISPs to prioritize certain applications. However, this method requires scanning for a certain patterns in the first few packets of each flow: 1-3 packets for unencrypted protocols and 3-20 packets for encrypted ones [38]. Therefore, certain processing delay will be added for all types of traffic because of overhead.

As described above, all the conventional approaches are mandatory and already utilized by ISPs today. However, they are not enough to solve all P2P problems. Especially, the conventional approaches did not concern the problem how to reduce inter-domain traffic generated by P2P systems, which is thorough solved in this dissertation.

## 2.3 Application-layer traffic optimization

Many approaches have been proposed to solve the interaction between application-layer overlays and the underlay networks. The available literature in this field can be divided into two main categories: end systems estimate the topology by themselves (measurement-based approaches) and operators or third parties provide the topology (Operator-provided topological information approaches).

### 2.3.1 Measurement-based approaches

The first approach tried to find mechanisms for topology estimation by end systems. Francis et al. proposed IDMaps, the first solution for network distance prediction [26]. IDMaps is a system where the distance between two arbitrary hosts A and B can be computed via some special hosts called tracers. The distance between A and B is estimated by summing the distance between A and its closest Tracer T1, plus the distance between B and its closest Tracer T2, and the distance of shortest path from T1 to T2. This idea was implemented in client/server architecture by supporting HOPS servers, where end hosts can query to obtain the network distance.

Deriving from this idea, Ng et al. have explored an architecture for network distance prediction based on P2P called global network positioning (GNP) [40]. It is a two-part architecture: In the first part, hosts in a small set known as Landmarks compute their own relative locations in a geographic space by simply measuring the round-trip time among these Landmarks. In the second part, ordinary hosts can estimate its own coordinates by measuring the round-trip time to the Landmarks.

Both GNP and IDMaps require some tracers or landmarks whose locations must have been known in advance. Therefore, it is difficult to reach high accuracy in large-scale networks. To overcome this weakness, Costa et al. proposed PIC, a practical coordinate-based mechanism to estimate round-trip time (RTT) or network hops between two arbitrary nodes on the Internet [24]. PIC maps each node to a point in a $d$-dimension Euclidean space. When a new node joins the system, it computes the coordinates of its corresponding point based on a set of landmarks, $L$, where $L \geq d + 1$ and their coordinates must be computed in advance. The idea of PIC is similar to GNP [40], but the number of landmarks in PIC is not fixed for all nodes that join the system, and therefore PIC has more practicality.

Wong et al. proposed a Meridian framework for performing node selection based on network location [51]. In Meridian, each node maintains a track of small fixed number of neighbors and organizes them into multi-resolution rings according to the distance from the node. When a Meridian node receives a request to find the closest node to a given target node, it first computes the latency $d$ between itself and the target, and then forwards the request to one of its ring members if the latency between the ring member and the target is less than $\beta \times d$, where $\beta$ is an acceptance threshold. The above steps are repeated with the new node. If no new nodes meet the threshold, then the routine stops and the currently closest node is chosen.

Although P2PTV can adapt to the above approaches for developing neighbor peer selection strategies, this is not a trivial task due to a requirement of many modifications of existing application software. In particular, each peer must be equipped with a location coordinate to estimate the locations of other peers. This may require a heavy computation. Moreover, when some overlay networks are running on one node simultaneously, such measurement-based approaches are very difficult to reach high accuracy.

### 2.3.2 Operator-provided topological information approaches

Over-provisioning and deep packet inspection (DPI)-based bandwidth management are considered the best conventional strategies to deal with P2P traffic [25]. However, they do not solve the fundamental concern of the ISPs, which is to reduce the cross-domain traffic, i.e., to localize the traffic. The idea of implementing "better-than-random" peer selection and traffic localization were originally proposed for P2P file sharing. Karagiannis et al. analyzed BitTorrent trace logs and concluded that about 50 percent of the files could be downloaded from active peers located in the same ISP [31]. Plissonneau et al. introduced their study on eDonkey file sharing and reported that 99.5 percent of traffic traversed nationwide or international networks [41]. It is also noted that about 40 percent of the traffic could be localized if locality-aware peer-selection mechanisms were integrated in the P2P protocol.

Bindal et al. proposed biased neighbor-selection scheme applying for BitTorrent in which a peer only selects $k$ external peers from other ISPs and the majority, $35 - k$ internal peers from the same ISP, where $k$ is a parameter [20]. This biased neighbor-selection scheme can reduce the cross ISP traffic significantly without an increase in download time. The idea can be implemented in two ways: the modification of trackers and clients and the use of P2P traffic shaping devices. The former certainly requires a lot of software modification, whereas the latter, similar to our approach, requires no modification of trackers and clients. However, it will be difficult to apply this idea to other type of P2P applications such as P2PTV because the peer list format has to be known in advance. In other words, the biased neighbor-selection scheme is dependent on the P2P applications.

To efficiently localize the P2P traffic, ISPs and P2P users should cooperate together for improving the performance. Aggarwal et al. proposed the so-called “oracle” service that could be provided by ISPs [16]. The ISPs, by having complete information of their own networks such as physical topology, bandwidth, and geographical information of peers, maintain an oracle service to help P2P systems make a better selection of neighbor peers. Deriving from the oracle idea, P4P is a promising framework [52]. P4P is a flexible architecture that allows network providers to provide more useful information to P2P systems. In P4P, each network provider, e.g., an ISP maintains an iTracker in its own network. The iTracker provides the p-distance interface, representing the logical distances and costs among PIDs (aggregation nodes) based on physical network information such as topology, routing cost, and provider policy. The P2P applications can query the interface to obtain underlay network information for choosing their neighbor peers more efficiently. Recently, the Internet Engineering Task Force (IETF) has formed a working group for standardizing a query/respond protocol to help P2P applications easily obtain network information provided by ISPs, known as Application Layer Traffic Optimization (ALTO) [18, 42]. Although the above approaches improve not only the network efficiency but also the P2P application performance, such kind of oracle-based approaches has the following requirements: (1) to open some detailed and/or sensitive information to external entities for efficient traffic localization, which raises the problem of security; (2) some dedicated servers for gathering underlay network information and providing this information to the applications; (3) several modifications of existing P2P application software for implementing an additional module to communicate with the dedicated servers; and (4) the trust and good cooperation between ISPs and P2P users.

Choffnes and Bustamante introduced another approach, which requires no cooperation between ISPs and P2P applications [22]. They claimed that the information necessary for peer selection is already gathered by content distribution networks (CDNs). Therefore, the presence of the oracle service provided by ISPs is redundant. By using DNS redirection, they hypothesized that if two peers are sent to a similar set of replica servers, they are recognized as being close to the servers, and more importantly close to each other. The idea is implemented as a java plugin, named "Ono" to Azureus BitTorrent client. This work might operate inefficiently without the support from many subscribing peers distributed worldwide. Furthermore, to apply this method for other types of P2P applications such as P2P streaming applications (P2PTV), I believe that some modifications must be required.

Liu et al. analyzed three locality-aware policies for BitTorrent-like system including tracker locality, choker locality and piece picker locality [35, 36]. They introduced AS hop-count to both tracker locality and choker locality level. For piece picker locality, they proposed a locality-first policy to encourage a peer to download pieces from the closest peers to itself. The modification of trackers and clients is surely required for this scheme. In a similar manner to that used in [35, 36], Jin et al. introduced three-tier framework applying for P2P multimedia streaming including tracker-tier neighbor selection, peer-tier neighbor selection and chunk scheduling [30].

While many researches focus on BitTorrent, Sheng et al. presented a traffic localization mechanism on another P2P file-sharing system named eMule [44]. They proposed to modify the eMule client for measuring the distance from it to other peers by itself. For calculating the distance, TTL and RTT were used.

NAPA-WINE project group (Network Aware Peer-to-peer Application over WIse NEtwork) proposed a network-aware architecture in which the application overlay layer and underlay network layer interoperate to optimize the service offered to end users [21]. The main goal of NAPA-WINE project is to propose a P2P architecture for P2PTV. The NAPA-WINE architecture regulates a total design that consists of user, overlay, messaging, monitoring, repository modules, and data communication and signaling among the modules. It means that the existing P2PTV might be modified to follow the protocol model.

## 2.4 Router-aided approaches

As described above, majority of the existing P2P locality-aware mechanisms require a lot of modifications in the clients and/or trackers for implementing biased neighbor peer selection. This is sometimes very hard, even if not impossible, due to a closed design and license problem of commercial software. Lee and Nakao introduced a new kind of P2P traffic localization technique applying to BitTorrent, called Netpherd, which does not require any modification of the application software [32, 33]. They proposed to turn the inter-domain traffic into intra-domain traffic by adding artificial delay to the inter-domain traffic. The idea of delay insertion is the same as our work. However, they focused on BitTorrent, a file sharing system. In addition, the artificial delay time is constant for all inter-AS traffic, e.g., 100ms. Netpherd thus only localizes the traffic at AS level.

Miyoshi et al. proposed P2P-DISTO (P2P Delay Insertion Scheme for Traffic Optimization), a router-aided approach, that focuses on P2P streaming applications [37]. According to the geographical location of peers, the packets transferred to foreign peers were inserted with a fixed length of additional delay, e.g., 500ms, 1000ms. P2P-DISTO thus only localized the traffic at country level. Furthermore, inserting a constant delay without taking into account the number of peers existing in the same area might cause the degradation of quality of service, e.g., P2P-DISTO will not work well if no peer exists in Japan.

In this study, I continue the work of P2P-DISTO, but try to localize the traffic hierarchically with multiple levels: AS level, ISP level, and country level. The hierarchy of localization realizes deeper traffic localization and also maintains the performance of the P2P applications. In addition, to achieve the idea of degradation of network performance of inter-domain traffic, I proposed three different schemes including delay insertion, forcing packet loss, and bandwidth limitation.

# Chapter 3

# Solutions for P2PTV traffic localization problem

As described above, this dissertation focuses on solving the traffic localization problem for P2PTV services. This chapter first reviews the protocol of P2PTV in general. Based on the protocol, all proposed solutions for P2P traffic localization problem are then introduced. Finally, I review the protocol in detail of some popular P2PTV applications including PPTV, PPStream, and SopCast.

## 3.1 P2PTV protocol

There are a lot of P2PTV applications, the protocols are therefore various [17, 27, 28, 29, 39, 43, 45, 49, 50]. However, in this dissertation I focus on some mesh-based P2PTV applications such as PPTV, SopCast, and PPStream. They have become more and more popular recently.

Figure 3.1 shows the protocol of P2PTV in general. In P2PTV systems, once a user initializes the playback of a channel, the peer will join an overlay network constructed by all the peers playing the same channel. The protocol in detail is as follows:

- Step 1. A querying peer sends a request to a peer list server to obtain the

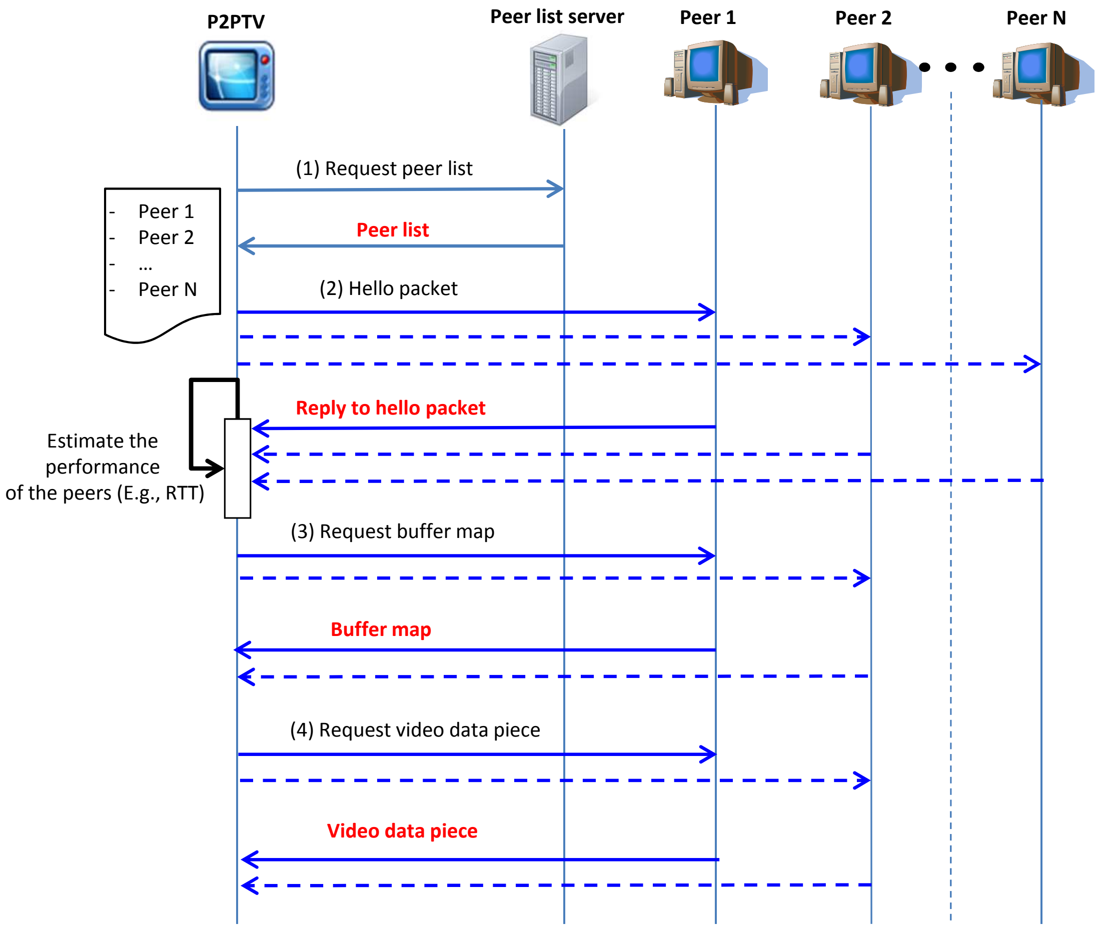



Figure 3.1: Protocol of P2PTV application in general.

list of candidate destinations where the desired resource resides.

- Step 2. After obtaining the peer list, the peer then sends hello packets to some candidate peers in the peer list to know if they are now active or not.

- Step 3. To increase the download speed, the querying peer estimates the network performance of the candidate peers, e.g., measures the RTT of hello packets, to eliminate the delayed peers.

- Step 4. The querying peer sends requests to some active peers to ask buffer-map messages; a buffer-map message indicates which video chunks a peer has already buffered and can share with other peers.

- Step 5. The querying peer sends requests to available peers for video data pieces.

- Step 6. The querying peer starts to exchange video data pieces with the candidate peer. During video data exchange, the querying peer sometimes queries the root servers to update the peer list as well.

## 3.2 Proposed solutions

I found three steps of the P2PTV protocol where I can intervene to make the traffic localized.

- The first one is at the step to obtain the peer list. If all the peer list packets are modified for localizing before they arrive at the P2PTV, the application then only connects to some peers in the modified peer list. In other words, the traffic can be localized. I thus propose the peer list modification method, which is described later in chapter 4. Furthermore, the peer list modification method can be combined with router collaboration for improving the performance of P2P traffic localization as shown later in chapter 7.

- The second one is at the step to request video data pieces. If all the video data requests sending to locality-unaware peers are redirected to local peers, the traffic can also be localized because only the local peers respond to the video requests. Therefore, I propose the video request packet redirection method, which is described later in chapter 5.

- The other is at the step 3. Based on an observation, the querying peer tends to select a candidate peer who has better performance, e.g., shorter RTT than others. Since the network performance is affected by various factors, communication with peers across network domains is sometimes better than the local communication. This leads to the increasing of cross-domain traffic. From the observation, if we intentionally degrade the network quality of connection paths of inter-domain traffic, the querying peer will tend to remove the inter-domain connections and select the local connections instead. In other words, we can turn the inter-domain traffic into the intra-domain traffic. To achieve this idea, I propose the degrading network performance of inter-domain connections method, as shown in detail in chapter 6.

  With the peer list modification method and the video request packet redirection method, the protocol of P2PTV is directly intervened. Therefore, both methods are not independent of P2P applications since they require to know the format of peer list packets in advance. For degrading the network performance of inter-domain connections method, the protocol of P2PTV is indirectly intervened. Therefore it is completely independent of P2P applications.

Next, I will review the protocol in detail of three very popular P2P streaming applications that will be used in this dissertation to evaluate my proposed methods.

## 3.3 PPTV

PPTV is the update version of PPLive which was first developed by Huazhong University of Science and Technology in 2004. PPTV system comprises the fol-

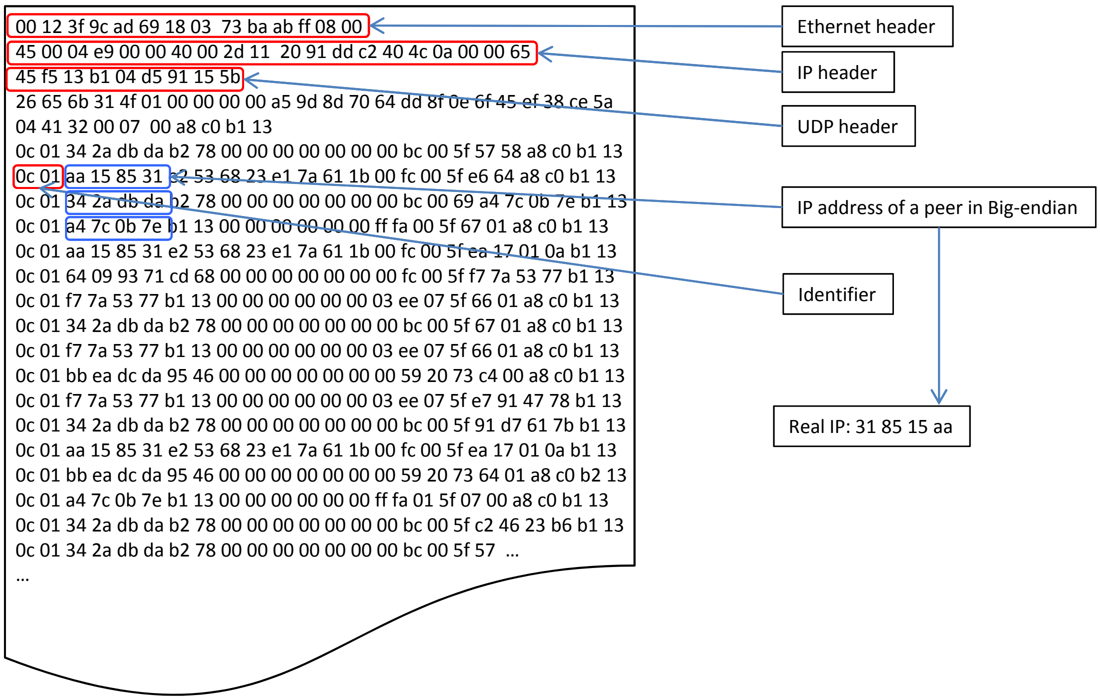



Figure 3.2: An example of a peer list packet of PPTV.

lowing components:

- Peer: a node that downloads video content from other peers and also uploads its own video content to other peers.
- Video streaming server: this is the source of video content.
- Channel server: this provides the list of available channels to the peers.
- Tracker server (peer list server): this provides a list of only peers that are watching the same channel as the querying peer.

According to a previous study, a PPTV peer connects to a constant number of peers to download video chunks. Furthermore, the top-ten peers contribute a major part of the download traffic, even one peer can be the only video provider [45].

The peer protocol of PPTV includes peer discovery and chunk distribution process. For the peer discovery, after obtaining the list of online peers watching the same channel from tracker server, a querying peer probes the active peers from the list. Some active peers may also return their own list of active peers to help the querying peer accelerate its peer discovery process. Chunk discovery process is based on buffer-map exchanging among peers. A peer advertises the available chunk that it has to other peers. To avoid video disruption due to download rate variation, PPTV deployed double buffer structure.

In PPTV, the peer list packet is sent in clear-text without any encoding. Figure 3.2 shows the format of a peer list packet of PPTV. It is interesting that the IP addresses are store in big-endian order. Therefore, to find the peers IP address, we need to use the inversed order.

## 3.4 PPStream

The system architecture of PPStream is very similar to that of PPTV. The system also includes a channel server, a peer list server, streaming sources and PPStream peers. Once a peer joins the overlay network, it receives the list of channels from the channel server. After selecting a channel to watch, the querying peer can receive a list of available peers that are watching the same channel from peer list server. For video chunk discovery process, PPStream also exchanges buffer-maps among peers. In particular, a PPStream peer selects peers to download video data chunks based on a rate-based algorithm in order to maximize the utility of uplink and downlink bandwidth.

PPStream tends to get the video data from many peers simultaneously. The peer list packet in PPStream is also sent without any encoding. Figure 3.3 shows an example of peer list packet.

## 3.5 SopCast

The system architecture and working flows of SopCast are similar to those of PPTV. SopCast transfer data mainly using UDP, occasionally TCP. The size of

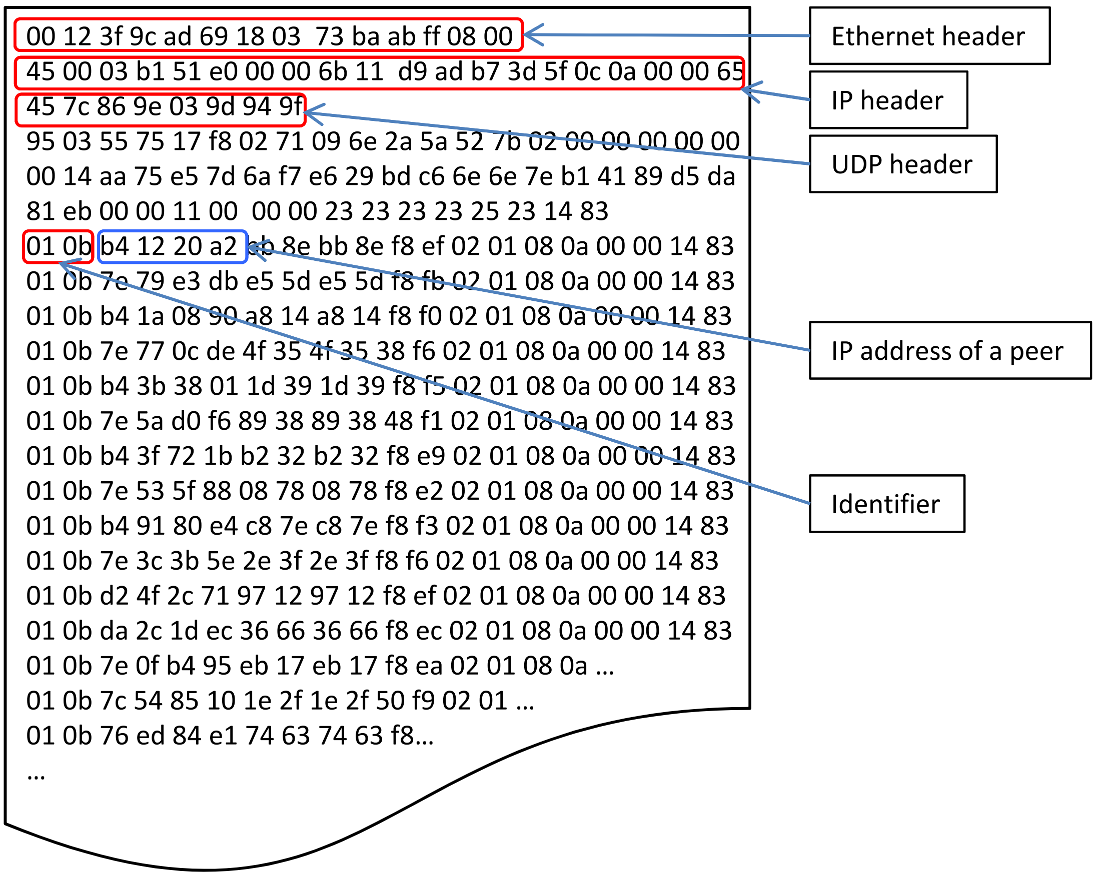



Figure 3.3: An example of a peer list packet of PPStream.

packet carrying video data of SopCast is normally larger than that of PPTV or PPStream.

In SopCast, top-ten peers contribute to almost all of total download traffic. SopCast tends to switch periodically among provider peers. However, it normally requires more than one peer to get the video data.

I could not check the format of peer list packet of SopCast. It might be sent with encoding.

## 3.6 Behavior of P2PTV in laboratory network environment

To understand the behavior of PPTV, PPStream, and SopCast in the laboratory network environment, I ran three applications one by one five times on a measurement host, with five minutes for each time. On SopCast, a live Chinese channel, CCTV-2, was selected to play. An on-demand drama popular in Japan and a Chinese drama were selected for the experiment on PPStream and PPTV, respectively.

Table 3.1 presents the number of connections during five minutes of experiments for each P2P application. We can see that, during five minutes, PPTV connects to too many peers (288 peers in average) while SopCast just connects to a small number of neighbor peers (60 in average).

Table 3.1: Number of connections of three P2PTV applications in five minutes.

| **No. experiment** | **PPTV** | **PPStream** | **SopCast** |
|---|---:|---:|---:|
| 1st | 289 | 183 | 38 |
| 2nd | 327 | 192 | 43 |
| 3rd | 173 | 180 | 78 |
| 4th | 325 | 185 | 80 |
| 5th | 327 | 190 | 64 |

Table 3.2 shows the percentage of downloaded traffic of top-ten peers in the total download traffic. Based on the results, top-ten peers of SopCast contribute to almost all total download traffic, more than 98%. In contrast, top-ten peers of PPStream and PPTV contribute only 36% and 42% of total download traffic in average, respectively.

Table 3.2: Traffic contribution of top-ten peers of three P2PTV applications in five minutes.

| **No. experiment** | **PPTV** | **PPStream** | **SopCast** |
|---|---:|---:|---:|
| 1st | 42.5% | 31% | 98% |
| 2nd | 36% | 40% | 99% |
| 3rd | 42% | 42% | 99% |
| 4th | 43% | 35% | 97% |
| 5th | 45% | 36% | 98% |

# Chapter 4

# Peer list modification method

## 4.1 Introduction

This chapter proposes a novel approach to localize the P2PTV traffic. In P2PTV, a peer generally receives a list of available peers as its neighbors from some peer list servers. The peer then contacts some peers in the peer list for exchanging video data by sending video data request packets. Xiao Su et al. reported that the peer list is sent in clear text without any encoding [46]. By deep packet inspection, we analyze every packet going through a gateway router to determine which packet contains the peer list, and to parse the list of IP addresses of all peers in the peer list. The geographical locations of all peers are easily resolved by using several IP-to-location database services. This study proposes to modify the peer list packets for localizing according to geographical location at network routers before they arrived at the application. Since the application then connects to some peers in the localized peer list, the traffic will therefore be localized.

Since the peer list modification scheme is implemented on the gateway routers, it requires neither modification of existing application software, nor enhancement of trackers, nor new protocol for communicating between trackers and applications. Therefore, the proposal can be easily applied for all P2P applications.

## 4.2 Proposed scheme

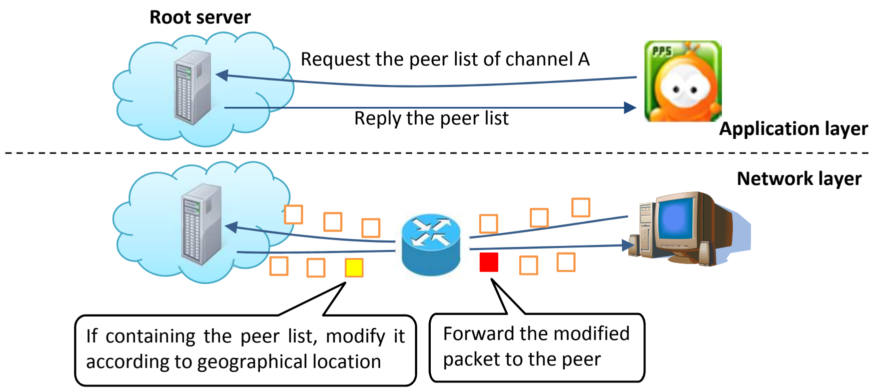



Figure 4.1: The concept of peer list modification scheme.

In P2PTV system such as PPStream, once a user chooses a channel for watching, the peer will join an overlay network formed by all the peers watching the same channel. At the beginning, a peer asks some root servers to obtain the list of available peers that have its desired data. During video data exchange, the peer sometimes queries the root servers to update the peer list as well. As mentioned in Chapter 3, the protocol of a P2PTV can be intervened at the step of obtaining the peer list for traffic localization. Figure 4.1 illustrates the idea of the peer list modification scheme. The process is simple as follows: (1) every packet flowing through a gateway router is first examined; (2) Packets that contain the peer list sent by trackers or peer list servers will be modified for localizing; (3) Finally, the modified peer list is forwarded to PPStream. For instance, to localize the traffic inside Japan, the scheme replaces foreign peers by Japanese peers in the peer list. The list of Japanese peers is not only collected at current peer list but also accumulated from previous peer lists.

To deploy the proposed method without any software modifications, the peer list modification should be implemented from outside of the existing applications. Therefore, I introduce a router-aided approach. Figure 4.2 presents the router architecture for the proposed peer list modification method. Two modules, a packet monitoring module and a peer list modification module, are added to a common router. The packet monitoring module inspects every packet going

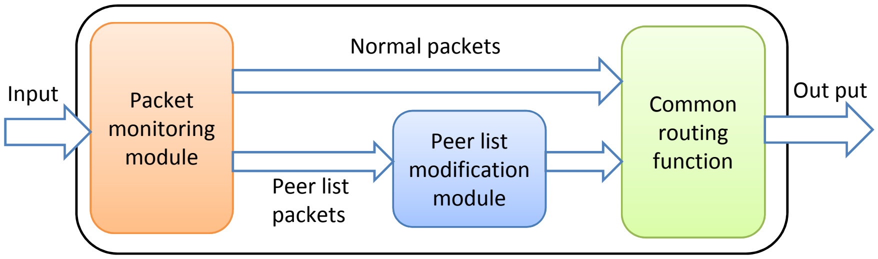



Figure 4.2: A router architecture for peer list modification scheme.

through the router to check whether it contains the peer list or not. If the peer list is found in the packets, the IP addresses of all peers in the peer list are collected, and the packets are passed to the peer list modification module, otherwise the packets are forwarded directly to the common routing function. In the peer list modification module, the obtained IP addresses are first mapped to their countries by using several IP-to-location database services. Then, the peer list is modified as follows: keep only $k$ foreign peers in the peer list, where $k$ is a parameter with value from 0 to the number of peers.

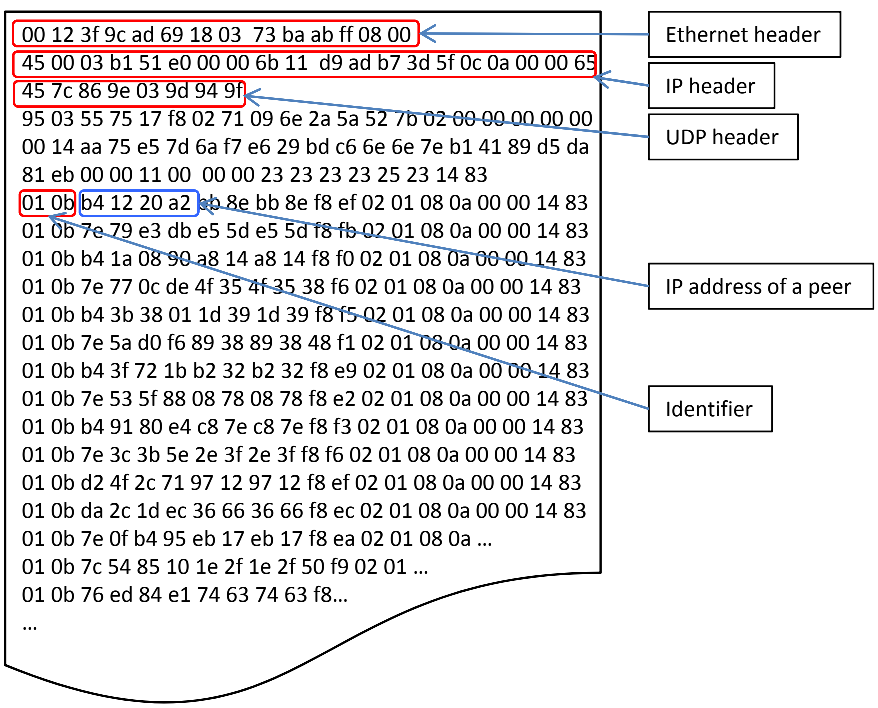



Figure 4.3: The format of a peer list packet of PPStream.

To determine which packet contains the peer list, Wireshark [10], a well-known packet-sniffer, was utilized to reverse-engineer the protocol of PPStream. The format of packet that contains the peer list was found as shown in Fig. 4.3. It was an UDP packet. After knowing the format of peer list packet, it is easy to recognize the peer list packet by finding the "identifier, e.g., "01 0b in the packet.

Since different applications may have different formats of the peer list packet, the proposed method has a drawback that it depends on the peer list format of applications. However, it is not difficult to find the peer list format of other P2P applications by deep packet inspection.

## 4.3 Implementation

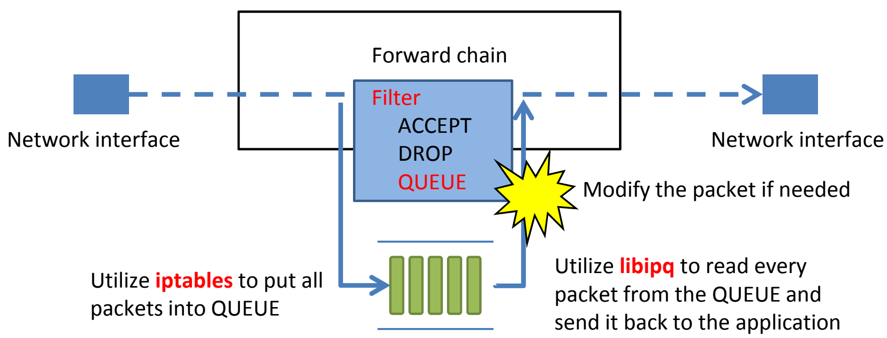



Figure 4.4: The procedure of the implementation for peer list modification scheme.

To evaluate the effectiveness of the peer list modification scheme, a desktop PC was set up as a software router. The hardware configuration of the router is as follows: Intel Core i7-2600 3.4GHz CPU, 12 GB of DDR3 memory, and two 1 Gbps Ethernet network interface cards, operated under Linux Ubuntu 12.04 with 3.2.0-29 generic kernel.

Figure 4.4 shows procedure the implementation. In this implementation, `libipq` [11], a library for iptables packet queueing, was utilized to capture all packets flowing through the router. `Libipq` provides a mechanism to intervene and modify the packets before they arrive at the application. First, all packets are pushed into QUEUE by using `iptables` command, e.g., `iptables -A FORWARD -i eth0 -o eth1 -j QUEUE`. Then libipq is used to read every packet from the QUEUE and to send it back to user space. The packet can be modified by calling an API function provided by `libipq` library: `ipq_set_verdict` with reasonable parameters. In partucular, the data payload of the peer list packet will be modified. Note that all the check sum fields should be recomputed in modifying the content of packets.

For the IP-to-location mapping, this implementation only checks the countries where peers are located by utilizing GeoLite Country database, a free IP geolocation database created by MaxMind [12].

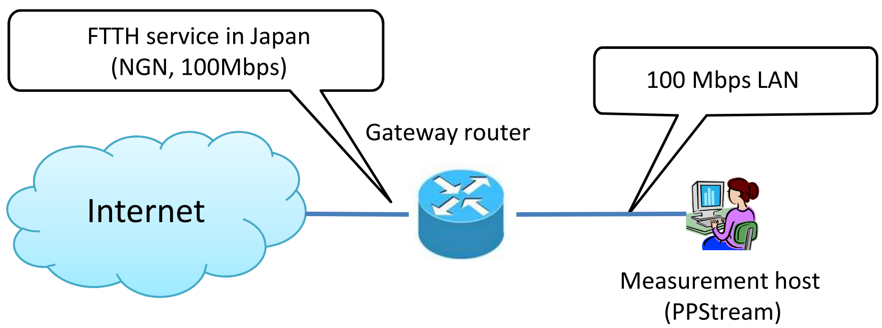



Figure 4.5: Experimental network environment setting.

## 4.4 Experimental results

### 4.4.1 Experimental Setting

The experiments were performed using an existing P2PTV application, namely PPStream on the Internet. Figure 4.5 shows the experimental network environment. For the network connection, we subscribed to FLET'S HIKARI NEXT, a 100 Mbps optical access service on the next generation network (NGN), and plala HIKARI Mate with FLET'S as an ISP in Japan. The proposed router is placed as a gateway router. A measurement host connects to the Internet via the gateway router. The hardware configuration of the measurement host is as follows: Intel

Core i5-2440 CPU 3.1 GHz, 4GB of memory, and a 100 Mbps network interface card, operated under 64-bit Windows 7.

An on-demand drama was selected to run on the measurement host. For the statistical information, `Wireshark` was installed on the measurement host. All the measurement were conducted in January 2013.

### 4.4.2 Results of peer list modification mechanism

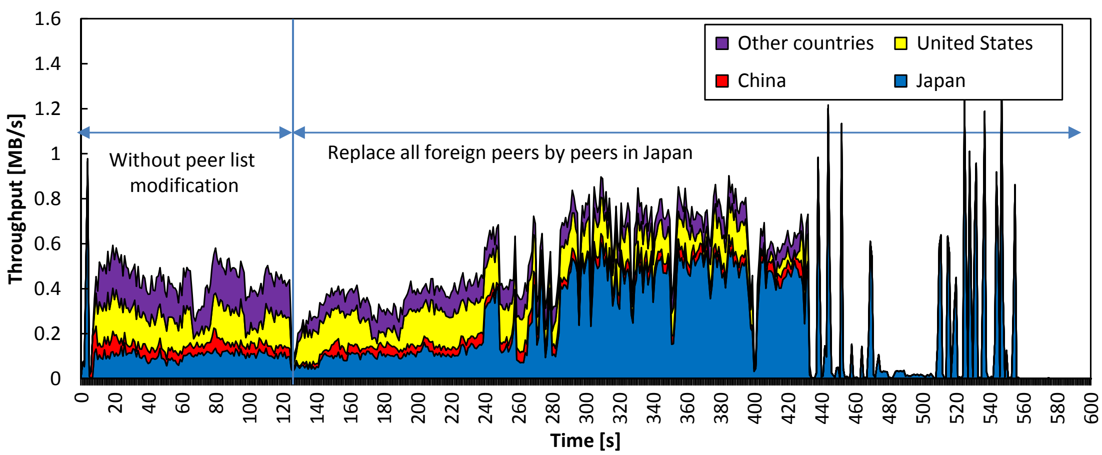



Figure 4.6: Temporal change of throughput for PPStream with real-time the peer list modification while playing a channel.

To prove the efficiency of the peer list modification mechanism, two experiments were set up. In the first experiment, the peer list was in real time modified while playing a predefined channel. The channel was played totally for 10 minutes. During the first two minutes, the peer list was not modified. After 2 minutes elapsed, the peer list was modified as follows: all foreign peers were replaced by peers in Japan that had been accumulated from all the previous peer lists, i.e., the parameter $k = 0$. The value of 2 minutes was selected as the time period to modify the peer list because 2 minutes is enough for PPStream to start exchanging video data with some peers in some different countries. In the Fig. 4.6, we can see the change of throughput in the connection of PPStream. During the first two minutes, without modifying the peer list, the application connects to peers located in many countries including China, Japan, the United States and other countries, which are bundled into one group. After the second minute, the application tends to change the connection to peers in Japan, and the traffic from Japan is gradually increasing in particular. From the 8th minute, all the traffic is downloaded from Japan. We can also see that the traffic from Japan does not change immediately right after applying the method, but gradually increases. It is possible to infer that PPStream remains the connection with some peers for few minutes before switching to other peers in the new peer list.

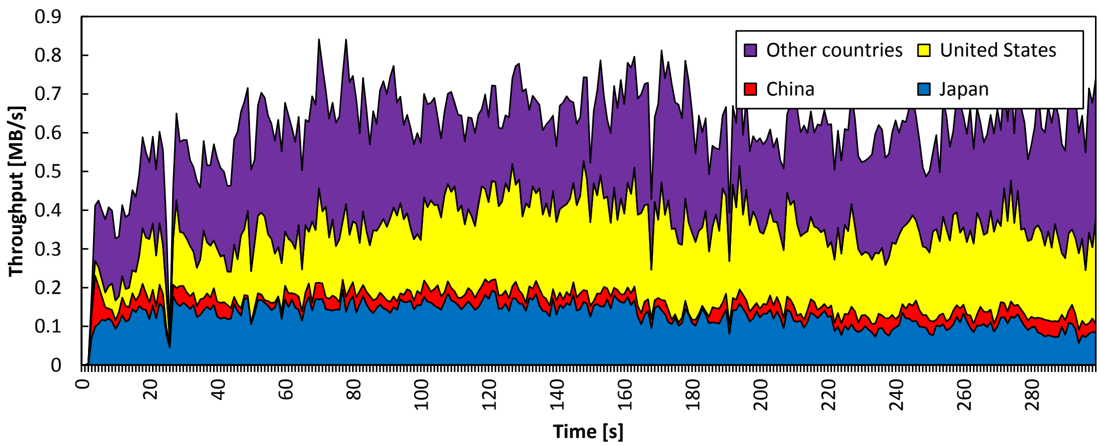



Figure 4.7: Temporal change of throughput by regions for original behavior of PPStream.

In the second experiment, the same part of the video on demand was played three times on the measurement host. The peer list is modified according to different scenarios as shown in the Figs. 4.7, 4.8 and 4.9. In scenario 1, without any modification of the peer list, i.e., keep the original behavior of PPStream, the traffic comes from many countries including China, Japan, the United States, and other countries.

In scenario 2, the peer list was modified by replacing all foreign peers by peers in Japan, i.e., the parameter $k = 0$. As expected, almost all traffic comes from Japan as shown in the Fig. 4.8.

In scenario 3, the negative case of the localization problem was evaluated by removing all Japan peers in the peer list, i.e., the parameter $k = N$, where $N$ is the number of peers in the peer list. The traffic from Japan is almost zero as

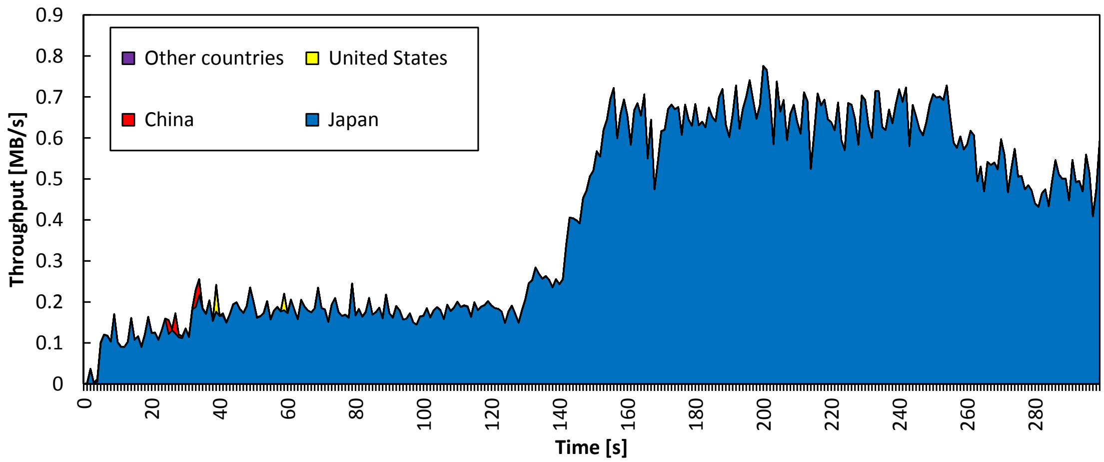



Figure 4.8: Temporal change of throughput by regions when replacing all foreign peers in the peer list by peers in Japan.

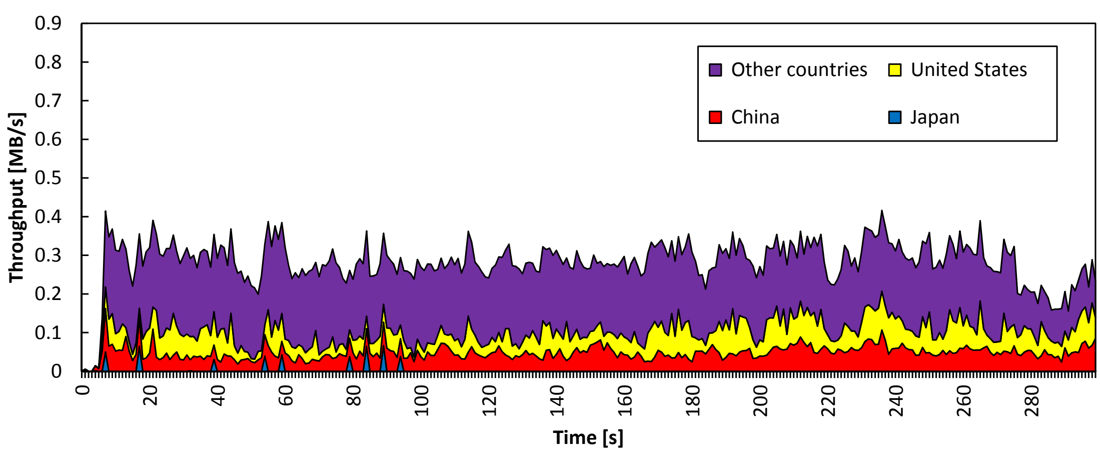
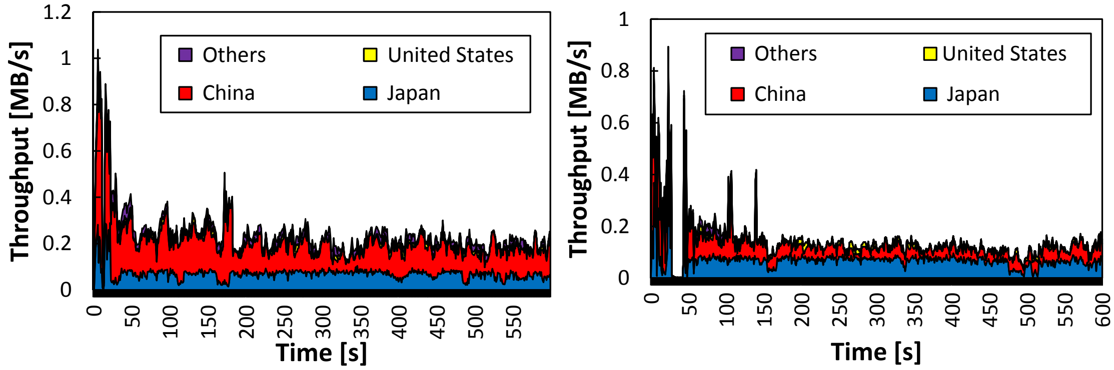
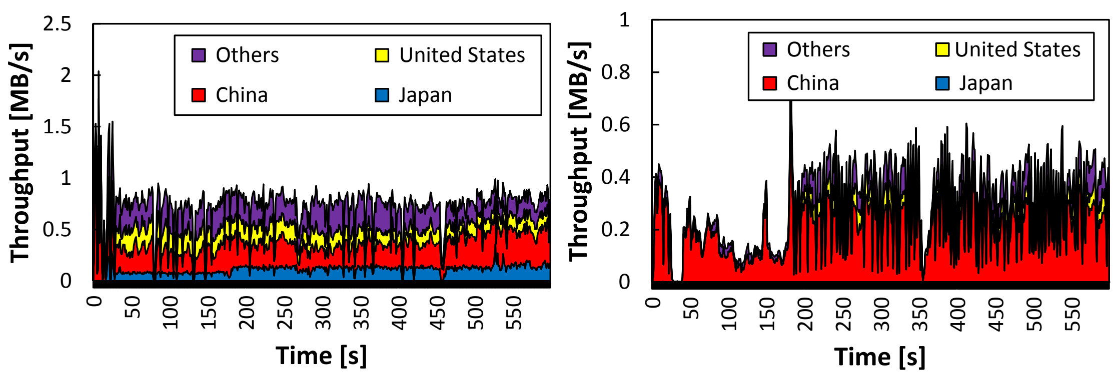



Figure 4.9: Temporal change of throughput by regions when removing all Japanese peers from the peer list.




(a) Keep original behavior, host 1. (b) Replace foreign peers by Japanese peers, host 2.

Figure 4.10: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying the peer list modification on the measurement host 2




(a) Keep original behavior, host 1. (b) Remove Japanese peers, host 2.

Figure 4.11: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying the peer list modification on the measurement host 2

a consequence; in particular, the blue color corresponding to traffic from Japan almost disappeared in the Fig. 4.9. Note that three scenarios watch the same part of the video during the same period, 5 minutes, but the amounts of traffic in three scenarios are quite different. The experiments were performed five times and the data cached by PPStream for five minutes was checked. It is remarked that the data cached in scenario 1 was usually larger than that of the other scenarios. It means that the peer list modification affects the download speed of the video. Therefore, the selection of the parameter $k$ with a reasonable value should be taken into account. Nevertheless, the above results prove that the peer list modification mechanism can be successfully applied for addressing the traffic localization problem.

### 4.4.3 Update results with newest version of PPStream

Figures 4.10 and 4.11 present the updated results of experiment 2 with the newest version of PPStream (version 3.6). The experiment was conducted in July 2014. This time, I perform the experiment on two hosts simultaneously. The original behavior of PPStream tends to download video data pieces from many countries as shown in Figs. 4.10 (a) and 4.11 (a). When foreign peers in the peer list are replaced by Japanese peers, we can see that the traffic inside Japan increases as shown in Fig. 4.10 (b). However, there is some traffic comes from China in Fig. 4.10 (b) even if all foreign peers have been replaced by Japanese peer in the peer list. This proves that PPStream version 3.6 has some updates compare to version 2.7. The application also contacts with some peers that do not appear in the plain-text peer list. PPStream still contacts primarily with Japanese peers that exist in the modified peer list though. In figure 4.11 (b), since all Japanese peers are remove from the peer list, no traffic inside Japan appears in the result

## 4.5 Conclusion

This chapter proposed a router-aided approach for P2P traffic optimization. The proposed scheme modifies the peer list packets for localizing according to geographical location of the destinations at network routers before they arrived at the application. This study also presented an implementation method for peer list modification scheme by utilizing `iptables`, `libipq`, GeoLite Country database on a PC-based router. Experiments evaluated on a popular P2PTV, PPStream proved that the proposed method successfully realizes traffic localization. Since the method is implemented from outside of the peer, it does not require any modification of existing application software. This proposal can be easily deployed on traffic-shaping devices to help ISPs control the P2P traffic.

There remain several problems in the current implementation. First, the format of the peer list packet must be known in advance. This makes the method dependent on the protocol of the applications. Secondly, using only country information of peers is insufficient, and might cause performance degradation, e.g. the degradation of video quality. Further study will improve this method using more information of the peers such as ISP, AS, and quality of connection links to achieve finer-grained result. In the future, this method will be apply to other P2PTV applications such as PPTV [2], SopCast [4], and Zattoo [5].

# Chapter 5

# Video request packet redirection method

## 5.1 Introduction

This chapter proposes another router-aided method to localize the P2PTV traffic by directly intervene to the protocol of P2PTV. As mentioned in previous chapters, in P2PTV protocol, a peer generally receives a list of available peers as its neighbors from some peer list servers. The peer then sends video data request packets to some peers in the peer list for asking video data pieces. The peer finally exchanges video data with some active peers. This study proposes to intervene at the step of sending video data requests to make the traffic localized. The process in detail is as follows: (1) at gateway routers, a list of local peers from all the obtained peer list packets are collected, e.g., a list of peers in Japan, and are marked as a localized list; (2) All video data request packets sent to the peers that do not exist in the localized list will be modified to redirect them to the peers exist in the localized list. In other words, the control information of the packets, the destination IP addresses in particular, will be modified. Since the application only sends queries to some peers in the localized list, only local peers will response the video data pieces, i.e., the traffic will therefore be localized.

Since the video request packet redirection scheme is also implemented on the

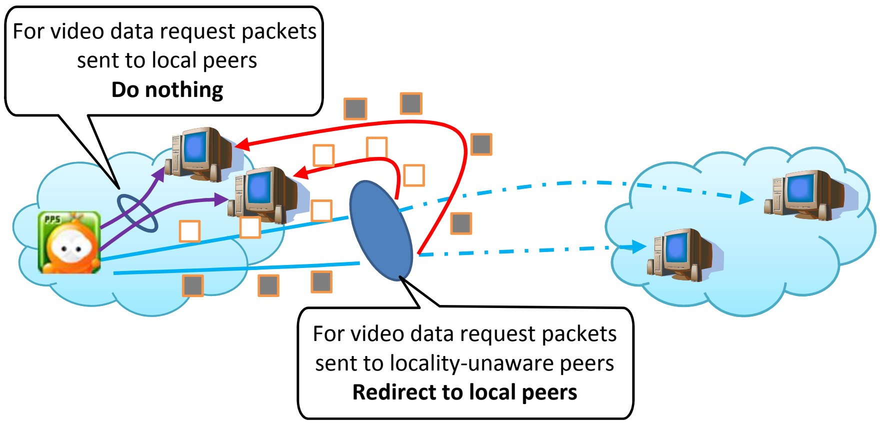



Figure 5.1: The concept of packet redirection scheme.

gateway routers, it requires neither modification of existing application software, nor enhancement of trackers, nor new protocol for communicating between trackers and applications.

## 5.2 Proposed scheme

Here after, we refer to scheme to redirect the video data request packets as packet redirection scheme for short. Figure 5.1 presents the concept of packet redirection scheme. In particular, the scheme does nothing with the video data request packets sent to local peers, but redirects all request packets sent to locality-unaware peers into local peers.

This method requires knowing which packet is the video data request packet for the redirection. By analyzing Wiresharks packet capture files of PPStream, we can found many similar UDP packets that have large data payloads, more than 1000 bytes. These packets are likely to carry the video data chunks. I assume that the video data request packets must be sent to some peers in the peer list and followed by the packets carrying video data chunks. Such kinds of packets are found in UDP format, and have very small data payloads, less than 100 bytes. Figure 5.2 shows an example of the video data request packet.

Figure 5.3 presents the router architecture for the proposed packet redirection

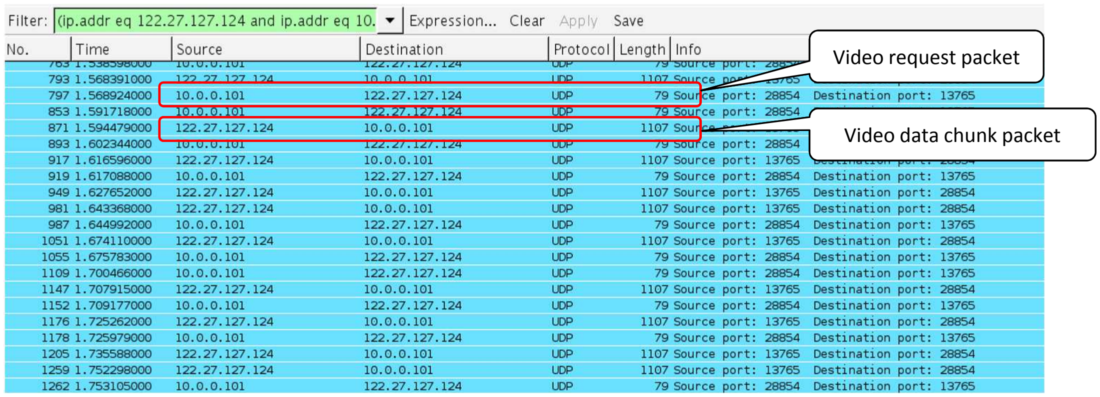



Figure 5.2: An example video data request packet.

scheme. It is quite similar to the architecture of the peer list modification scheme, but the peer list modification module is replaced by a packet redirection module. The packet monitoring module inspects every packet going through the router: if the packet contains the peer list, the IP addresses of local peers in the peer list, e.g., all IP addresses in Japan are collected. If the packet is the video data request type, it is forwarded to the redirection module, otherwise the packet is sent directly to the common routing function. In the redirection module, the destination IP address of the packet is modified to redirect it to one of the local peers.

Because of deep packet inspection, we just intervene in P2P traffic, and therefore pose no threat to the bandwidth availability of other applications. On the other hand, the redirection scheme has a drawback that the format of peer list packet and video request packet must be known in advance. This makes the proposed method dependent on the protocol of P2P applications.

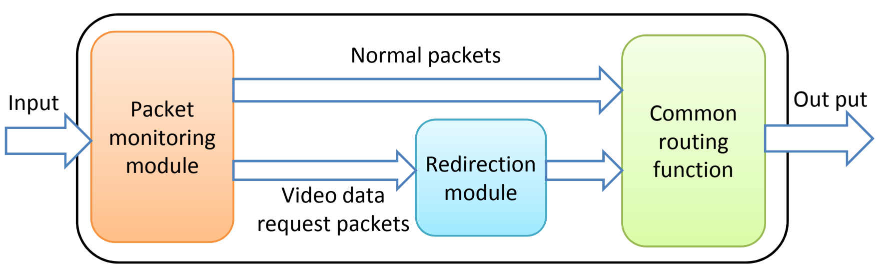



Figure 5.3: A router architecture for packet redirection scheme.

## 5.3 Implementation

To evaluate the effectiveness of the packet redirection scheme, a desktop PC was set up as a software router. The hardware configuration of the router is as follows: Intel Core i7-2600 3.4GHz CPU, 12 GB of DDR3 memory, and two 1 Gbps Ethernet network interface cards, operated under Linux Ubuntu 12.04 with 3.2.0-29 generic kernel.

The implementation of packet redirection scheme is very similar to that of peer list modification scheme described in previous chapter. The procedure is as follows: (1) All packets are pushed into QUEUE by using `iptables` command; (2) Libipq is then used to read every packet from the QUEUE and to send it back to user space; (3) The video request packets will be modified by calling an API function provided by `libipq` library: `ipq_set_verdict` with reasonable parameters. In particular, the control information of the packet is modified; i.e., the destination IP address of the packet is changed. For the IP-to-location mapping, this study only checks the countries where peers are located by utilizing GeoLite Country database.

## 5.4 Experimental results

Two experiments were set up in a similar manner to those of the peer list modification mechanism described in previous chapter. Figure 5.4 shows the result of the first experiment, the temporal change of throughput for PPStream with real-time redirection of video data request packets while playing a channel. It is easy to see that the result is very similar to those of the peer list modification mechanism. In particular, during the first two minutes, without packet redirection, the application downloads the video data from many countries including China, Japan, the United States and other countries, which are bundled into one group. After the second minute, since every video request packet sent to foreign peers was redirected into peers in Japan, the application tends to change the connection to peers in Japan, and the traffic from Japan is gradually increasing in particular.

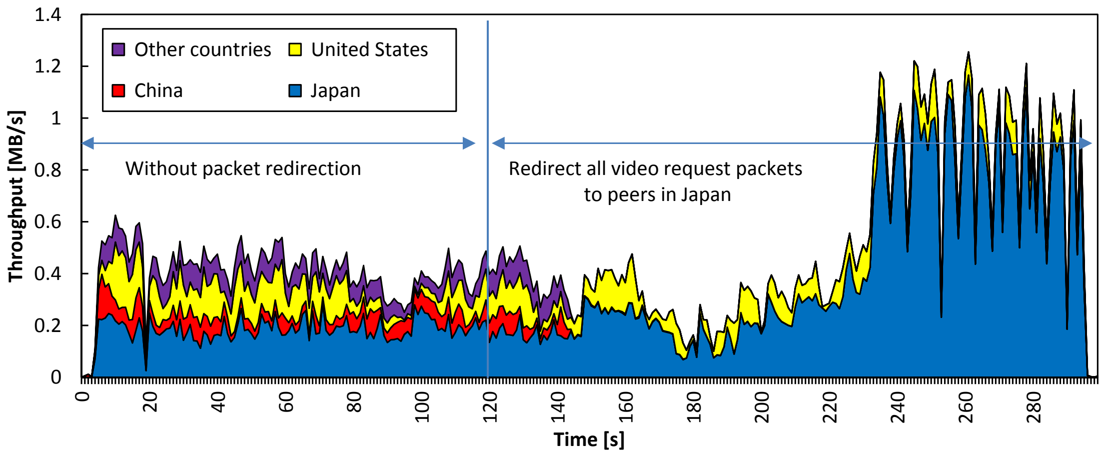



Figure 5.4: Temporal change of throughput by regions for PPStream with real-time redirection of video request packets.

In the second experiment, the same part of the video on demand was played on the measurement host with three different scenarios as follows:

- Scenario 1: Keep the original behavior of PPStream.
- Scenario 2: Redirect all video request packets to peers in Japan.
- Scenario 3: Redirect all video request packets to peers in the United States.

In scenario 1, without applying the packet redirection, video data were exchanged with peers in some countries such as Japan, the United States, and other countries, as shown in Fig. 5.5. In scenario 2, all video data request packets sent to foreign peers were redirected to the peers in Japan. The list of Japan peers had been collected from all the previously obtained peer lists. As expected, Fig. 5.6 shows that almost all traffic is presented in green color corresponding to the traffic from Japan. In scenario 3, to test the negative side of the localization, all video data request packets were redirected to peers in the United States, which is very far from Japan. Figure 5.7 shows that almost all traffic comes from the United States.

The experimental results indicate that the packet redirection scheme realizes traffic flow control on PPStream application.

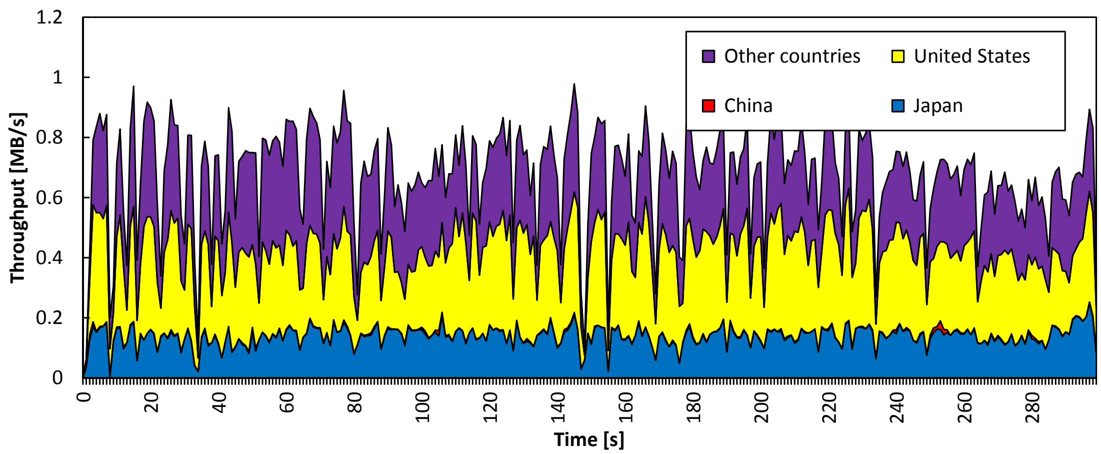



Figure 5.5: Temporal change of throughput by regions for original behavior of PPStream.

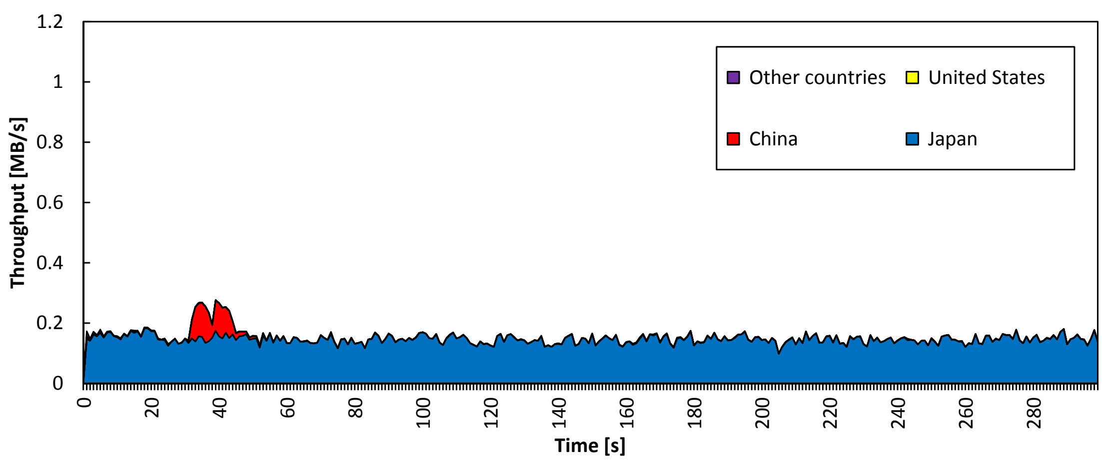



Figure 5.6: Temporal change of throughput by regions for PPStream when redirecting all video request packets to peers in Japan.

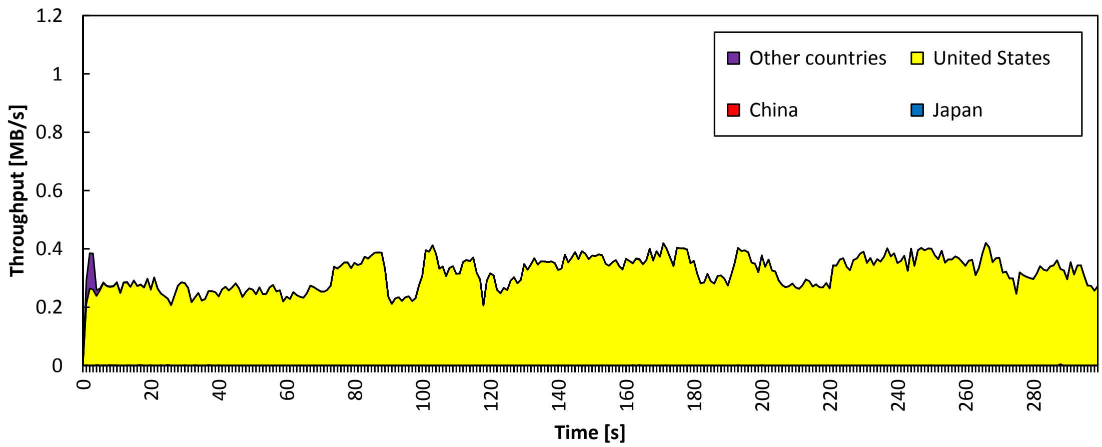
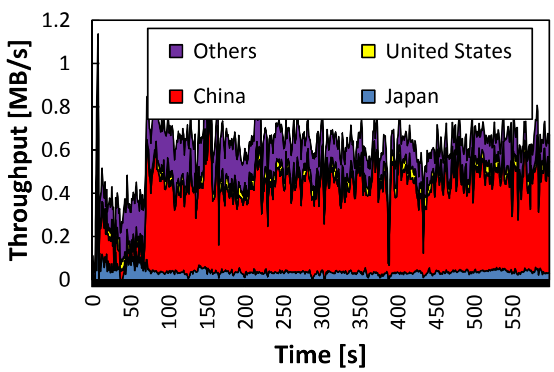
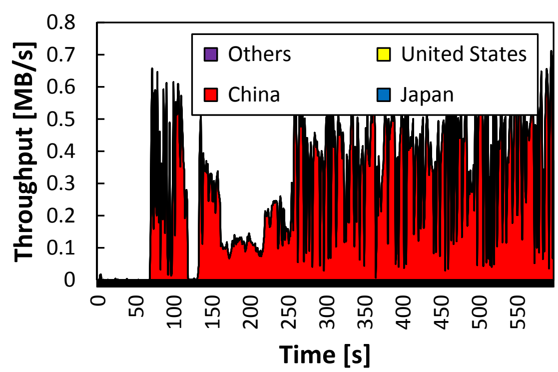



Figure 5.7: Temporal change of throughput by regions for PPStream when redirecting all video request packets to peers in the United States.




(a) Keep original behavior, host 1.




(b) Redirect video request packets to peers in China, host 2.

Figure 5.8: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying the packet redirection on measurement host 2

### 5.4.1 Update results with newest version of PPStream

Figure 5.8 presents the updated results of experiment 2 with the newest version of PPStream (version 3.6). The experiment was conducted in July 2014. This time, I performed the experiment on two hosts simultaneously. We can see that the results is very similar to those of older version (PPStream 2.7). In particular, the original behavior of PPStream on host 1 tends to download video data pieces from many countries as shown in Fig. 5.8 (a). When apply packet redirection on host 2, by redirecting all video request packets to peers in China, almost all traffic comes from China as shown in Fig. 5.8 (b).

## 5.5 Conclusion

This chapter proposed a router-aided approach for P2P traffic localization. To make the traffic localized, the proposed scheme redirects all video request packets sent to locality-unaware peers to the local peers. An implementation method for packet redirection scheme is also introduced by utilizing `iptables`, `libipq`, GeoLite Country database on a PC-based router. Experiments evaluated on a popular P2PTV, PPStream proved that the proposed method successfully realizes traffic localization. Since the method is implemented at gateway routers, it does not require any modification of existing application software. This proposal can be easily deployed on traffic-shaping devices to help ISPs control the P2P traffic.

Several future challenges remain. First, the format of the peer list packet and the video request packet must be known in advance. This makes the method dependent on the protocol of the applications. Second, using IP-to-country mapping database in the current implementation causes the traffic localization is only coarse-grained. Further study will test other mapping database services such as IP-to-AS number and IP-to-city in attempts to achieving finer-grained traffic awareness. Furthermore, the investigation of the degradation of video quality when applying the method should be also taken into account.

# Chapter 6

# Degrading Network Performance of Inter-domain Connections Method

## 6.1 Introduction

As described in chapters 4 and 5, both the peer list modification method and the video request packet redirection method are not completely independent of P2P applications. Although both methods require no modification of existing P2P application software, they require to known the format of peer list packets and/or video request packets in advance. In other words, they depend on the protocol of P2P applications. This chapter introduces a novel approach to localize P2P traffic completely independent of P2P traffic applications. As described in chapter 3, we can also make traffic localized by indirectly intervene the protocol of P2P applications.

I exploit an important feature of P2P applications that a querying peer will select a candidate peer as its neighbor if the candidate peer is likely to provide better performance. For instance, the querying peer tends to select a candidate peer who has shorter RTT than others. Since the network performance is affected by various factors, communication with peers across network domains is sometimes better than the local communication. This leads to the increasing of cross-domain traffic. Based on this observation, if we intentionally degrade the quality of connection paths of inter-domain traffic, the querying peer will tend to remove the inter-domain connections and select the local connections instead. In other words, we can turn the inter-domain traffic into the intra-domain traffic. To achieve this idea, this chapter proposes three following schemes:

- Delay insertion scheme (DIS): Each P2P packet will be inserted additional delay according to geographical locations of the destinations at network routers. In particular, for farther peers, longer delay will be inserted than closer ones. Clearly, a packet with an additional delay takes longer to travel than an undelayed one, the corresponding response definitely arrives later, and the RTT of the delayed communication is therefore increased. Since the P2P application tends remove the poor quality peers, the traffic can be localized.

- Packet loss scheme (PLS): Each P2P packet will be forced to drop with a given probability according to geographical locations of the destinations at network routers. In particular, for farther peers, a higher probability of packet loss will be forced than closer ones. By discarding packet probabilistically, the P2P traffic can be also localized for the following reasons: (1) Packet loss might cause a failure of transmitting the hand shaking packets with farther peers, and closer peers are therefore likely to be selected instead; (2) If no failure occurs in the hand shaking phase, the peer might download video data from a peer located in a different ISP/AS. However, the subsequent packet loss will make this connection un-stable. Moreover, P2P streaming applications tend to close unstable connections to improve the quality of the video. Therefore, cross-ISP/AS connections will be reduced; in other words, the traffic can be localized.

- Bandwidth limitation scheme (BLS): The bandwidth of each P2P connection will be limited according to geographical locations of the destinations at network routers. In particular, for farther peers, a lower bandwidth will be allocated than closer ones. Clearly, a higher bandwidth connection is definitely better than a lower bandwidth one in term of both latency and packet loss rate. Therefore, the P2P traffic can be also localized by applying bandwidth limitation scheme.

In addition, this chapter also proposes to localize the P2P traffic not only at a single level but also multiple levels. In particular, the traffic is first localized at AS level if some candidate peers exist inside the same AS. The scope of localization will change from AS level to ISP level if no candidate peer exists in the same AS. Similarly, the scope can be change from ISP level to country level if no candidate peer exists in the same ISP. The hierarchy of localization can solve the problem of the trade-off between traffic localization and service quality of P2P applications. In other words, the proposed hierarchical framework can reduce the cross-domain traffic as well as conserve the performance of P2P applications.

In summary, this work provides the following contributions:

- Compared with existing locality-enhancing approaches focusing on the application layer, the proposed method requires neither dedicated servers, nor collaboration between ISPs and P2P users nor modification of P2P application software.
- Compared to random and/or RTT-based peer-selection strategies, the proposed method significantly reduces cross-ISP/AS traffic.
- This work provides a hierarchical localization approach to balance the trade-off between traffic localization and service quality of P2P applications.
- To implement the idea of degrading the quality of connection paths of inter-domain traffic, this work introduces three different schemes including delay insertion, forcing packet loss, and bandwidth limitation.

## 6.2 Proposed Method

Let us consider an overlay network in which a querying peer receives a list of candidates located in different areas. Without a locality-aware mechanism, in general, the querying peer often randomly selects a set of candidates to contact with. To increase the download speed, P2P applications including P2PTV currently tend to eliminate the delayed peers based on the RTT measured before starting downloading the data pieces. In consideration of this feature, this study proposes to degrade the quality of connection paths of inter-domain traffic in order to turn the inter-domain traffic into intra-domain traffic.

Given a querying peer, $\text{peer}_0$, and a list of $N$ candidate peers, $\{\text{peer}_1, \text{peer}_2, ..., \text{peer}_N\}$, let $(\text{as}_0, \text{isp}_0, \text{cc}_0)$ be denoted AS number, ISP name, and country code of the querying peer, respectively, and $(\text{as}_i, \text{isp}_i, \text{cc}_i)$ be denoted AS number, ISP name, and country code of $\text{peer}_i$, respectively. The main goal is to compute the delay length, the packet loss rate, and the limited bandwidth assigned to a candidate $\text{peer}_i$.

### 6.2.1 Fixed-length Degrading Network Performance of Inter-domain Connections Schemes

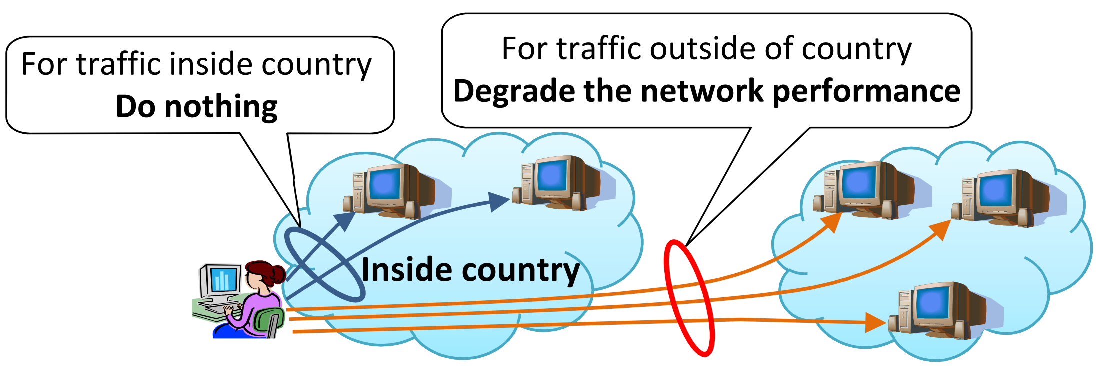



Figure 6.1: The concept of fixed-length degrading network performance of inter-domain connections method.

To prove the effectiveness of degrading network performance of inter-domain connections method in P2P traffic localization problem. I first introduce fixed-length degrading method including fixed-length delay insertion scheme (FDIS), fixed-length packet loss scheme (FPLS), and fixed-length bandwidth limitation scheme (FBLS). Figure 6.1 presents the general concept of the fixed-length degrading network performance method. The method does nothing with local traffic inside the same country as the querying peer, but degrade network performance of traffic that goes out to or come in from different countries. By doing that, the RTTs of the foreign peers is increased. The querying peer will then prefer to connect with local neighboring peers that have shorter RTTs.

**(1) Delay Length Computation for FDIS**

The delay length assigned to a candidate peer$_i$ is computed by [ms], as follows:

$$\text{delay\_length}_i = F_1(\text{cc}_i, \text{cc}_0) = \begin{cases} 0, & \text{if } \text{cc}_i = \text{cc}_0 \\ C_1, & \text{if } \text{cc}_i \neq \text{cc}_0 \end{cases} \tag{6.1}$$

where $C_1$ is a constant number.

**(2) Packet Loss Rate Computation for FPLS**

The packet loss rate assigned to a candidate peer$_i$ is computed by [%], as follows:

$$\text{plr}_i = F_2(\text{cc}_i, \text{cc}_0) = \begin{cases} 0, & \text{if } \text{cc}_i = \text{cc}_0 \\ C_2, & \text{if } \text{cc}_i \neq \text{cc}_0 \end{cases} \tag{6.2}$$

where $C_2$ is a constant number.

**(3) Limited Bandwidth Computation for FBLS**

The limited bandwidth assigned to a candidate peer$_i$ is computed by [kbps], as follows:

$$\text{bw}_i = F_3(\text{cc}_i, \text{cc}_0) = \begin{cases} +\infty, & \text{if } \text{cc}_i = \text{cc}_0 \\ C_3, & \text{if } \text{cc}_i \neq \text{cc}_0 \end{cases} \tag{6.3}$$

where $C_3$ is a constant number.

### 6.2.2 Hierarchical Degrading Network Performance of Inter-domain Connection Schemes

Since localizing the traffic at country level only is surely not enough in real situation, the objective of this dissertation is to localize the traffic hierarchically with AS level, ISP level, and country level. Figure 6.2 illustrates the concept of the proposed hierarchical method at AS level. The method does nothing with local traffic within the same AS, but degrade the network performance the connections to/from different ASes, ISPs, or countries. For farther peers, the network performance is forced to degrade with higher amplitude than closer ones. The scope of localization will be changed from AS level to ISP and country level if no candidate peer exists in the same AS and ISP, respectively. The behavior of the ISP level is as follows: do nothing with the traffic within the same ISP, but degrade the network performance the connections to/from different ISPs or countries. Similarly, for the country level, the proposed method does nothing with the traffic inside the country but degrade the network performance of oversea connections. To intentionally degrade the network performance of inter-domain traffic, this study proposes three different mechanisms including delay insertion, forcing packet loss, and bandwidth limitation.

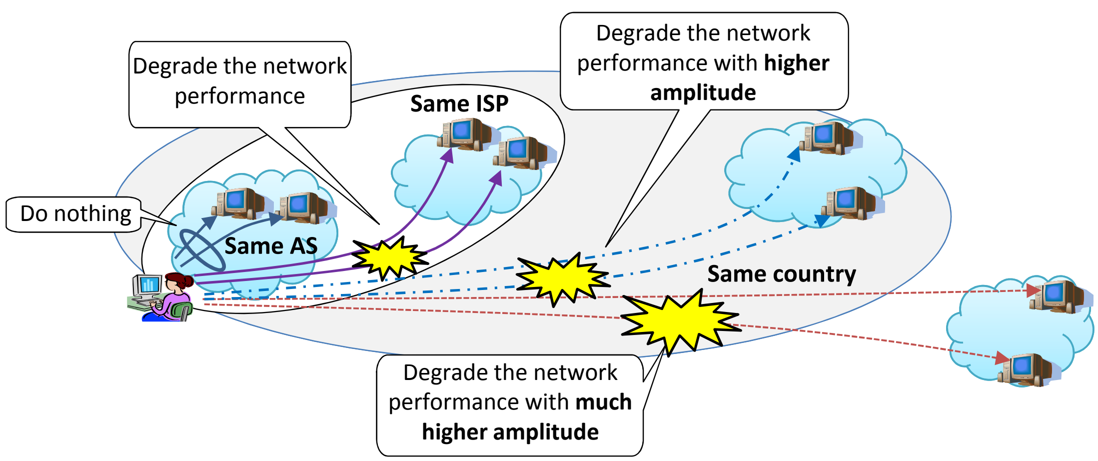



Figure 6.2: The concept of hierarchical traffic localization at AS level.

#### (1) Logical Distance Between Peers

To realize the concept of hierarchical traffic localization mentioned above, I first define a logical distance representing an distance adjustment factor between the candidate $\mathrm{peer}_i$ and the querying peer $\mathrm{peer}_0$ as follows:

$$\begin{aligned} D_i &= f_1(\mathrm{as}_i, \mathrm{as}_0)e^{-\frac{1}{n_1+\varepsilon}} + f_2(\mathrm{isp}_i, \mathrm{isp}_0)e^{-\frac{1}{n_2+\varepsilon}} \\ &\quad + f_3(\mathrm{isp}_i, \mathrm{cc}_i, \mathrm{isp}_0, \mathrm{cc}_0)e^{-\frac{1}{n_3+\varepsilon}}, \end{aligned} \tag{6.4}$$

where $n_1$, $n_2$, and $n_3$ are the total numbers of peers in the same AS, ISP, and country as $peer_0$, respectively, $\varepsilon$ is a very tiny constant to ensure the denominators of all fractions never come to zero, and

$$f_1(\mathrm{as}_i, \mathrm{as}_0) = \begin{cases} 0, & \text{if } \mathrm{as}_i = \mathrm{as}_0 \\ \theta_1, & \text{if } \mathrm{as}_i \neq \mathrm{as}_0 \end{cases} \tag{6.5}$$

$$f_2(\mathrm{isp}_i, \mathrm{isp}_0) = \begin{cases} 0, & \text{if } \mathrm{isp}_i = \mathrm{isp}_0 \\ \theta_2, & \text{if } \mathrm{isp}_i \neq \mathrm{isp}_0 \end{cases} \tag{6.6}$$

$$f_3(\mathrm{isp}_i, \mathrm{cc}_i) = \begin{cases} 0, & \text{if } \mathrm{isp}_i = \mathrm{isp}_0 \\ d(\mathrm{peer}_i, \mathrm{peer}_0), & \text{if } \mathrm{isp}_i \neq \mathrm{isp}_0, \\ & \text{and } \mathrm{cc}_i = \mathrm{cc}_0 \\ \theta_3 + d(\mathrm{peer}_i, \mathrm{peer}_0), & \text{if } \mathrm{cc}_i \neq \mathrm{cc}_0 \end{cases} \tag{6.7}$$

Since ISPs, including ASes, have to manage their own networks, the information that the querying peer connects to a peer exists inside or outside the AS/ISP is the most important. Hence, $\theta_1$ and $\theta_2$ are coefficients to differentiate the inter-AS/ISP traffic from the intra-AS/ISP traffic, respectively. To ensure that the logical distances of farther peers will be higher than those of closer ones, I define $d(\mathrm{peer}_i, \mathrm{peer}_0)$ as the physical distance between $\mathrm{peer}_i$ and $\mathrm{peer}_0$. Even though some nearby physical locations might be far apart from each other in terms of network connectivity in some specific cases, the physical distance is still a reasonable estimation in most cases. The coefficient, $\theta_3$, is to make the distances of foreign peers sufficient higher than those of local ones.

From above equations, the logical distance is based on not only the physical distance but also the number of peers in the same area as the querying peer.

This enables the method to realize the hierarchy of localization. For instance, if no candidate peer exists in the same AS or ISP, i.e., $n_1 = n_2 = 0$, the first two exponential functions in Eq. (6.4) will come to zero, the logical distance will therefore depend only on the country information. In the worst case, if no candidate peer exists in the same country as the querying peer, $n_1 = n_2 = n_3 = 0$, the logical distance will be almost zero, which means that the method will not degrade the network performance of the inter-domain traffic. Therefore, the performance of P2P applications will not be affected when applying the proposed method even if no local peer exists.

**(2) Delay Length Computation for HDIS**

Since longer delay should be inserted into connections with farther peers, the delay length for each candidate peer$_i$ is computed by simply normalizing the logical distance $D_i$ as follows:

$$\mathrm{delay\_length}_i = \frac{D_i}{20000} \times T \quad [\mathrm{ms}], \tag{6.8}$$

where T is a delay unit. I expect that almost all logical distances will not exceed 20000 km, approximating the half of the circumference of the Earth.

**(3) Packet Loss Rate Computation for HPLS**

The the packet loss rate for each candidate peer$_i$ is simply computed as follows:

$$\mathrm{plr}_i = \frac{D_i}{20000}, \tag{6.9}$$

**(4) Limited Bandwidth Computation for HBLS**

Since lower bandwidth should be allocated for farther peers, we compute the limited bandwidth for a candidate $peer_i$ as follows:

$$\mathrm{bw}_i = \frac{1}{\mathrm{D}_i + \varepsilon_1} \times \mathrm{B} \quad [\mathrm{kbps}], \tag{6.10}$$

where B is a bandwidth unit, and $\varepsilon_1$ is a tiny constant to ensure the denominator of the fraction never come to zero.

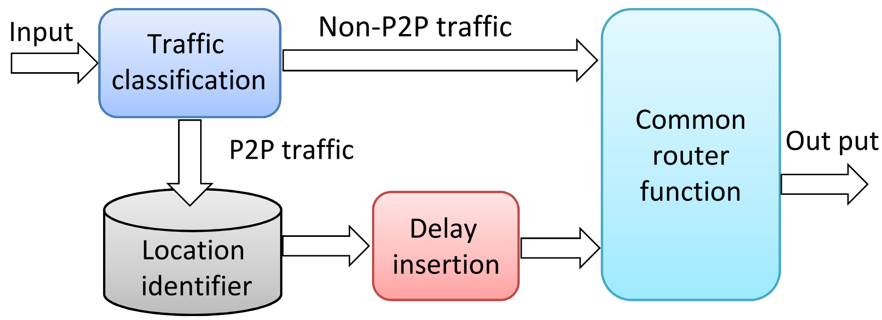



Figure 6.3: The router architecture for delay insertion schemes.

### 6.2.3 Proposed Router Architecture

**(1) Router architecture for delay insertion schemes (HDIS and FDIS)**

This study introduces a router-aided approach to implement the proposed method independently of the P2P applications. Figure 6.3 shows the architecture of the router for delay insertion scheme. Three modules, a traffic classification, a location identifier, and a delay insertion module, are added into a common router. The traffic classification module classifies the input traffic into P2P or non-P2P traffic. To avoid the degradation of service quality of non-P2P applications, the non-P2P traffic goes directly to the common router function. In the location identifier, the destination IP address of every P2P packet is first examined. Next, the location information of the destination such as AS number, ISP name, country code, and geographical location (latitude and longitude) are resolved by using several IP-to-geographic-location database services. At this step, the numbers of the peers in the same AS, ISP, and country as the querying peer are also updated. The delay insertion module holds the packet for a delay length period. Note that the additional delay length for a candidate peer is computed from Eqs. (6.1) and (6.8) for FDIS and HDIS, respectively.

**(2) Router architecture for forcing packet loss schemes (HPLS and FPLS)**

Figure 6.4 shows the architecture of the router for forcing packet loss scheme. The router architecture for packet loss rate scheme is very similar to that of delay insertion scheme. The difference is that the delay insertion module is replaced

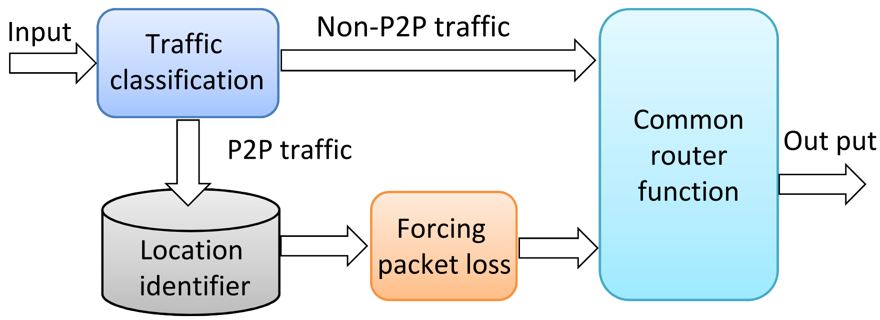



Figure 6.4: The router architecture for forcing packet loss schemes.

by the forcing packet loss module. Instead of adding delay into the packet, the packet loss module forces the packet to drop with a given probability (packet loss rate). The packet loss rate is calculated from Eqs. (6.2) and (6.9) for FPLS and HPLS, respectively.

**(3) Router architecture for bandwidth limitation scheems (HBLS and FBLS)**

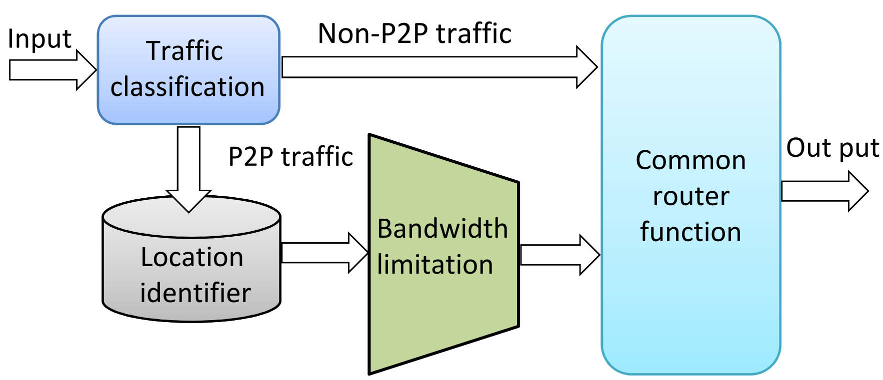



Figure 6.5: The router architecture for bandwidth limitation schemes.

Figure 6.5 shows the architecture of the router for bandwidth limitation scheme. Similarly, the router architecture for bandwidth limitation scheme is also very similar to that of delay insertion scheme. However, the delay insertion module is replaced by bandwidth limitation module. The bandwidth limitation module limits the bandwidth of connection between the querying peer and the candidate peer. The limited bandwidth for each candidate peer is computed from Eqs. (6.3) and (6.10), respectively.

## 6.3 Implementation of Proposed Method

The proposed method does not require any specific network architecture. It can be applied to the current Internet by replacing the conventional routers by the proposed routers. This dissertation tries to implement the proposed scheme on a home gateway router for the following reasons: Firstly, the implementation and the experiment can be easily performed on the local side. A low-power router, such as a PC-based router, will be enough to process the proposed delay insertion since the amount of traffic from a home network is not so large. Secondly, I would like to prove that the proposed method demonstrably localizes P2PTV traffic even at a single network edge. On the other hand, the proposed router could be also deployed on any routers in the Internet. In that case, multiple proposed routers will cooperatively work with some kind of distributed functions among them. As described later in Sect. 6.6, such an extended applicability of the proposed method will be an issue to be addressed in the future.

A desktop PC was setup as a software router. The hardware configuration is as follows: Intel Core i7-2600 3.4GHz CPU, 12 GB of DDR3 memory, and two 1 Gbps Ethernet network interface cards, operated under Linux Ubuntu 12.04 with 3.2.0-29 generic kernel.

### 6.3.1 Implementation of Traffic Classification

For the traffic classification module, many methods have been proposed. For example, to block P2P traffic, ISPs usually apply deep packet inspection and session-based classification with 5 tuples (IP addresses, port numbers, and protocol type). Recently, Valenti et al. introduced "Abacus", an accurate behavioral classification method for P2P traffic relying only on the count of packets and bytes that peers exchange during small fixed-length time windows [48]. In addition, as described in previous chapters, some P2P streaming applications such as PPStream and PPTV send the peer list packets in clear text without encoding. Therefore, all traffic transferred with the peers existing in the peer list can be recognized as P2P traffic. I can easily utilize such types of classification methods to implement the module in the proposed router. In this study, however, I assume that such the classification module is beyond the scope of this dissertation, and thus focus only on the implementation of the delay insertion module for HDIS, the forcing packet loss module for HPLS, and the bandwidth limitation module for HBLS to verify the effectiveness of traffic localization by three schemes in a real network.

### 6.3.2 Implementation of Location Identifier

The implementation of location identifier includes two main steps:

- *Packet monitoring*: `libpcap`, a well-known packet capture library is utilized to examine all packets coming into the router [13]. The headers of the packets are analyzed to read their source and destination IP addresses. The list of peers is updated at this step.
- *IP-to-location mapping*: the obtained IP addresses are then mapped to their locations by using IP-to-location services as described above. In this implementation, we utilize GeoLite database services including GeoLite ASN, GeoLite City, and GeoLite Country, which are free IP geolocation databases created by MaxMind [12].

### 6.3.3 Implementation of Delay Insertion Module for FDIS and HDIS

The implementation of delay insertion module includes two main steps:

- *Computation of delay length*: the delay length for each IP address is computed according to Eqs. (6.1) and (6.8) for FDIS and HDIS respectively in the proposed method section.
- *Delay insertion*: to insert additional delay in a real network, I utilize `dummynet`, a flexible tool for simulating packet filtering, bandwidth management, packet delay, and packet loss [14]. `dummynet` has been originally

developed in FreeBSD, but is now also available for other frameworks including Linux and Windows. By using `ipfw` firewall, a user interface provided by `dummynet`, it is easily to setup many pipes between sender and receiver peers, and all the packets will be carried in these pipes. Depending on the delay length computed by the previous step, each pipe can be configured with a different delay period.

### 6.3.4 Implementation of Forcing Packet Loss Module for FPLS and HPLS

The implementation of forcing packet loss module includes two main steps:

- *Computation of packet loss rate*: the packet loss rate for each IP address is computed according to Eqs. (6.2) and (6.9) for FPLS and HPLS respectively in the proposed method section.

- *Forcing packet loss*: `Dummynet` was used to setup pipes from sender to receiver peers. According to the packet loss rate computed by previous step, each packet can be forced to drop at the given probability.

### 6.3.5 Implementation of Bandwidth Limitation Module for FBLS and HBLS

The implementation of bandwidth limitation module includes two main steps:

- *Computation of limited bandwidth*: the limited bandwidth for each IP address is computed according to Eqs. (6.3) and (6.10) for FBLS and HBLS respectively in the proposed method section.

- *Bandwidth limitation*: `Dummynet` was used to setup pipes from sender to receiver peers. Each pipe can be configured with a different bandwidth value computed from the previous step.

For the fixed-length schemes, the value of delay length, packet loss rate, or limited bandwidth are constants for all foreign peers. The implementation is therefore very simple as shown in algorithm 1. For every new peer coming into the router, I simply check its country information and apply delay insertion, forcing packet loss, or bandwidth limitation if the peer does not come from Japan.

For the hierarchical schemes, the delay length, the packet loss, and the limited bandwidth are computed from the logical distance. As described above, the logical distance depends on the numbers of peers in the same AS, ISP, and country as the querying peer. Since these numbers may change when a new peer comes, the delay lengths for all connected peers should be recomputed again in such cases. This causes a very high load on the CPU of the router. In addition, the delay insertion might not be effective if we change the configuration too often. My solution, therefore, employs to compute the logical distance for the new peer in real time and to update the logical distances for all the connected peers every one minute. This avoids the high load on the router's CPU, and ensures a regular updating of the delay lengths for all peers.

Algorithms 2 and 3 show pseudo codes of HDIS, HPLS, and HBLS for a new peer and for a list of connected peers, respectively.

**Algorithm 1:** Fixed-length degrading network performance schemes: configure the delay length, packet loss rate, or bandwidth for a new peer

**Data**: New packet, List of connected IP addresses: *ip_list*, The method: FDIS, FPLS, or FBLS

**Result**: Depending on the method, configure the delay length, packet loss rate, or bandwidth for a candidate peer

```
while TRUE do
    packet ⇐ read_new_packet();
    ip ⇐ check_header(packet);
    if ip is new then
        country_code ⇐ resolve_location(ip);
        if country_code != "JP" then
            if method = FDIS then
                delay ⇐ C1;
                call_dummynet_for_delay_insertion(ip, delay);
            if method = FPLS then
                plr ⇐ C2;
                call_dummynet_for_packet_loss(ip, plr);
            if method = FBLS then
                bw ⇐ C3;
                call_dummynet_for_limiting_bandwith(ip, bw);
        else
            do_nothing;
        ip_list ⇐ add_new_ip_to_list(ip);
    else
        do_nothing;
```

**Algorithm 2:** Hierarchical degrading network performance schemes: configure the delay length, packet loss rate, or bandwidth for a new peer

**Data**: New packet, The method: DIS, PLS, or BLS

**Result**: Depending on the method, configure the delay length, packet loss rate, or bandwidth for a new peer

**while** *TRUE* **do**
- *packet* ⇐ **read_new_packet**();
- *ip* ⇐ **check_header**(*packet*);
- **if** *ip is new* **then**
  - $(as, isp, country, lat, lon)$ ⇐ **resolve_location**(*ip*);
  - $(n_1, n_2, n_3)$ ⇐ **update_no_peers_same_area**($as, isp,$ $country$);
  - *logical_distance* ⇐ **compute_logical_distance**($as, isp,$ $country, lat, lon, n_1, n_2, n_3$);
  - **if** $method = HDIS$ **then**
    - *delay* ⇐ **compute_delay_length**(*logical_distance*);
    - **call_dummynet_for_delay_insertion**(ip, delay);
  - **if** $method = HPLS$ **then**
    - *plr* ⇐ **compute_packet_loss_rate**(*logical_distance*);
    - **call_dummynet_for_packet_loss**(ip, plr);
  - **if** $method = HBLS$ **then**
    - *bw* ⇐ **compute_limited_bandwidth**(*logical_distance*);
    - **call_dummynet_for_bandwidth_limitation**(ip, bw);
  - *ip_list* ⇐ **add_new_ip_to_list**(*ip*);
- **else**
  - **do_nothing**;

**Algorithm 3:** Hierarchical degrading network performance schemes: reconfigure the delay length, packet loss rate, or bandwidth for all connected peers

**Data**: List of connected IP addresses: *ip_list*

**Result**: Depending on the method, reconfigure the delay length, packet loss rate, or bandwidth for all connected peers

**call_dummynet_for_flushing_all_old_configurations**();<br>
$(n_1, n_2, n_3) \Leftarrow$ **count_no_peers_same_area**(*ip_list*);<br>
**for** $i = 1$ **to count**(*ip_list*) **do**<br>
    $(as, isp, country, lat, lon) \Leftarrow$ **get_location**(*ip_list*[$i$]);<br>
    *logical_distance* $\Leftarrow$ **compute_logical_distance**($as, isp,$ $country, lat, lon, n_1, n_2, n_3$);<br>
    **if** $method = HDIS$ **then**<br>
        *delay* $\Leftarrow$ **compute_delay_length**(*logical_distance*);<br>
        **call_dummynet_for_delay_insertion**(ip, delay);<br>
    **if** $method = HPLS$ **then**<br>
        *plr* $\Leftarrow$ **compute_packet_loss_rate**(*logical_distance*);<br>
        **call_dummynet_for_packet_loss**(ip, plr);<br>
    **if** $method = HBLS$ **then**<br>
        *bw* $\Leftarrow$ **compute_limited_bandwidth**(*logical_distance*);<br>
        **call_dummynet_for_bandwidth_limitation**(ip, bw);

## 6.4 Experimental Results

### 6.4.1 Experimental Setting

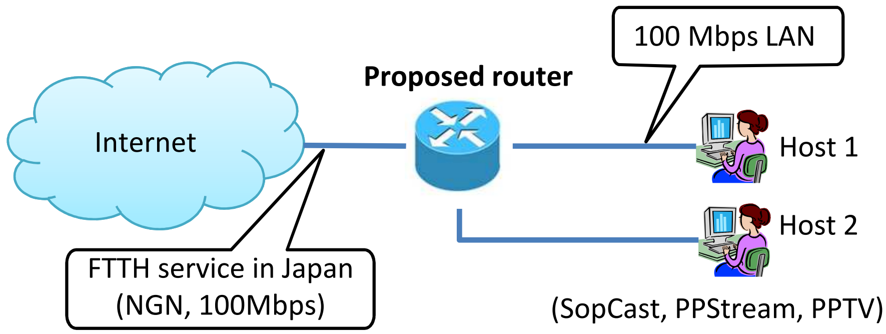



Figure 6.6: The network environment setting.

In this setup, the proposed router is placed as a subnet gateway router as shown in Fig. 6.6. The experiments were performed on P2PTV applications because of their popularity. Two types of P2PTV applications are selected: SopCast for performing video live streaming; and PPStream and PPTV for performing video-on-demand service. These applications did not consider peer locality, as reported in several previous studies [19, 23, 29, 46]. I set each application to run one-by-one on the measurement hosts. On SopCast, a live Chinese channel, CCTV-2, was selected to play. An on-demand drama popular in Japan and a Chinese drama were selected for the experiment on PPStream and PPTV, respectively. The average bit rates of these three video streams were 800 kbps, 705 kbps, and 1000 kbps, respectively.

All the experiments were conducted in my laboratory. The location information of measurement hosts in detail is as follows:

- AS number: AS4713
- ISP: NTT Communications Corp.
- Country: Japan.

To skip the implementation of the traffic classification module, as described in the previous section, only a P2P application and `Wireshark` are permitted to run on the measurement hosts.

The values of the parameters in the equations shown in previous section were chosen as follows: $\varepsilon = 0.1$, $\theta_1 = \theta_2 = 1000$, $\theta_3 = 2000$, $\varepsilon_1 = 1$, $T = 2000$, $B = 2000000$, $C_1 = 500ms$ and $1000ms$, $C_2 = 10\%$ and $20\%$, and $C_3 = 800kbps$ and $1000kbps$.

### 6.4.2 Criteria of Evaluation

The proposed schemes should be evaluated and compared with other schemes from two viewpoints: the traffic locality and the QoS. From the former viewpoint, I ran the each video for 300 seconds and measured the amount of downloaded data and the number of neighbor peers, and report their ratios by regions as the evaluation indexes in the following section. Each experiment on each P2P application was performed three times, with 300 seconds each time. The means of evaluation indexes were calculated as final results. Since I also wanted to check the possibility that a measurement host downloading the video data from the very neighbor peer inside our laboratory, I always run each P2P application on two measurement hosts simultaneously, as host 1 and host 2.

From the latter viewpoint, I measured the waiting time of users. After clicking on the play button, users have to wait a short time for the application to buffer enough data for starting playing. Therefore, the waiting time reflects the download speed, and thus can be used as a metric for measuring the performance of the applications. The waiting time was examined five times for each P2PTV application, and the average waiting time will be reported as the evaluation index in the following section.

### 6.4.3 Results of Delay Insertion Method

To evaluate the effectiveness of the delay insertion method, I compared the results of three different schemes:

- (1) random and/or RTT-based peer-selection scheme, i.e., the original behavior of P2PTV applications.
- (2) FDIS, fixed-length delay insertion scheme, in which all the overseas traffic will be constantly inserted additional delay, e.g., 500 ms and 1000 ms.
- (3) HDIS, hierarchical delay insertion scheme.

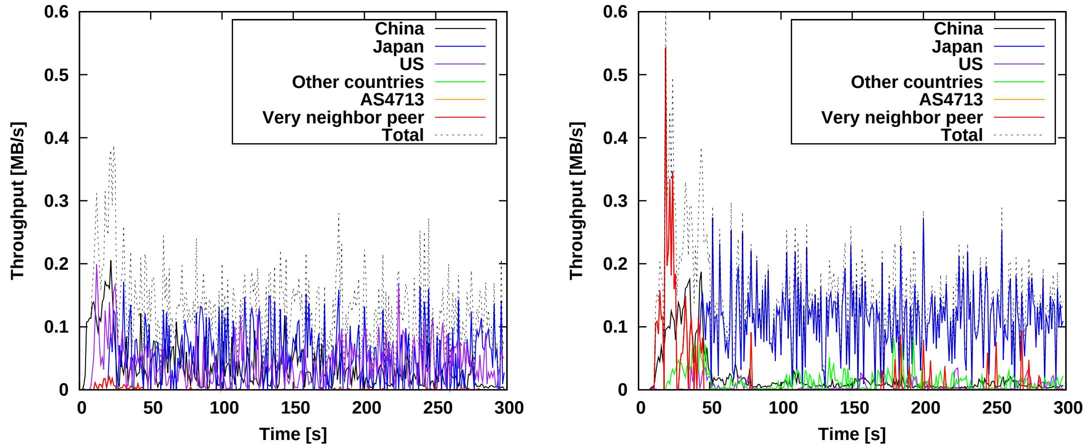
### (1) Results with SopCast

Table 6.1: An example of inserted additional delay length for SopCast. The last digits of IP addresses are anonymized.

| **Country** | Japan | | | China | Canada | United States |
|---|---|---|---|---|---|---|
| **ISP name** | NTT Commun. Corp. | NEC BIGLOBE, Ltd. | Softbank BB Corp. | China Telecom (Group) | Hurricane Electric, Inc. | Cogent/PSI |
| **AS number** | AS4713 | AS2518 | AS17676 | AS4812 | AS6939 | AS174 |
| **Peers' IP addresses** | **192.168.12.36** | 122.134.170.*** | 126.130.252.*** | 211.152.36.*** | 135.0.160.*** | 38.121.64.***, 38.121.64.*** |
| **Additional delay length [ms]** | **0** | **313** | **470** | **609** | **1279** | **1254** |

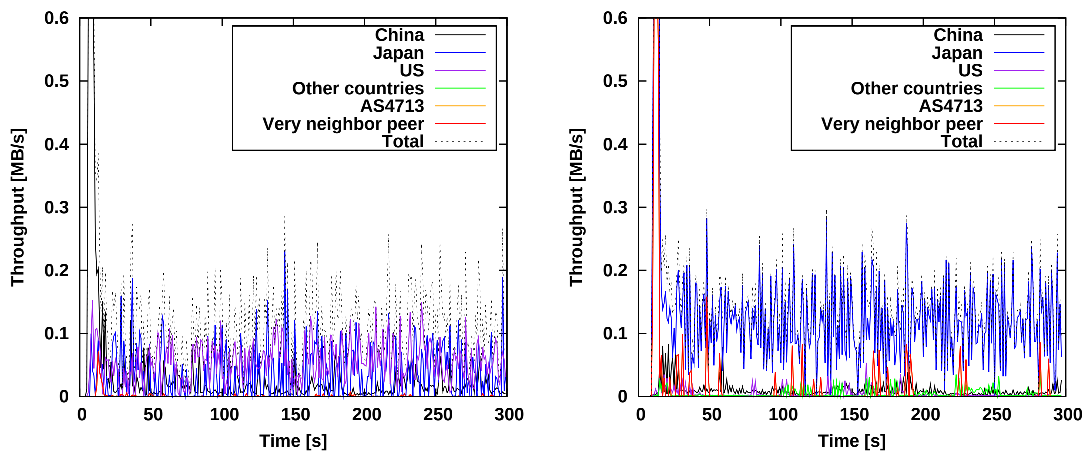
First, I present the results of HDIS obtained with SopCast. Table 6.1 shows an example of delay length assigned to several peers in different regions. We can see that the delay lengths are changed depending on the locations of the packet destinations.

Figures 6.7 and 6.8 show examples of temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying the FDIS on the measurement host 2. We can see from 6.7 (a) and 6.8 (a) that the original behavior of SopCast tends to download video data pieces from many countries including China, Japan, the United State and the others. However, host 1 did not recognize and download video data from the very neighbor peer, host 2. In particular, the traffic coming from host 2 is almost zero. In case of applying fixed-length 500 ms and 1000 ms delay insertion, most traffic measured on host 2 comes from Japan because SopCast tends to remove the connection paths with foreign peers that have longer RTTs than Japanese peers. However, the traffic coming from the very neighbor peer, host 1, is still




(a) Keep original behavior of SopCast, host 1. (b) Apply fixed-length 500 ms delay insertion on host 2.

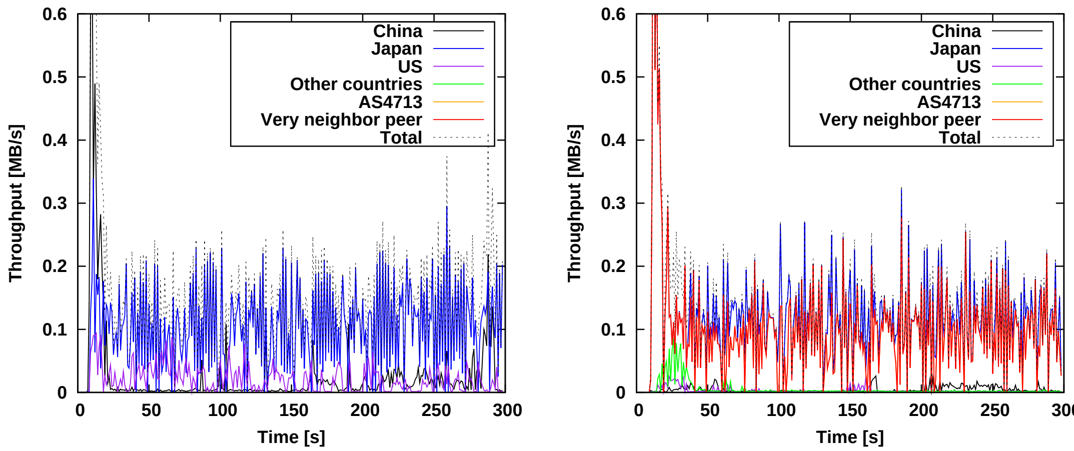
Figure 6.7: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying the fixed-length 500 ms delay insertion scheme on the measurement host 2.




(a) Keep original behavior of SopCast, host 1. (b) Apply fixed-length 1000 ms delay insertion on host 2.

Figure 6.8: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying the fixed-length 1000 ms delay insertion scheme on the measurement host 2.

very small. This is because the FDIS does not distinguish the very neighbor peer from the other Japanese peers, SopCast can download video data pieces from the very neighbor peer at one time, and from other Japanese peers at other times when the new peers are better.




(a) Keep original behavior of SopCast on host 1.(b) Apply hierarchical delay insertion on host 2.

Figure 6.9: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying HDIS on the measurement host 2.

Figure 6.9 shows an example of temporal changes of throughput when keeping original behavior of SopCast on the measurement host 1 and applying HDIS on the measurement host 2. With the hierarchical scheme, almost all the traffic measured on host 2 is downloaded from the very neighbor peer, host 1, as shown in Fig. 6.9 (b). In contrast, at the same time host 1 could not download any video data from its very neighbor peer, host 2. This is because our hierarchical scheme has degraded network performance of inter-AS connections by inserting additional delay. Therefore, SopCast tends to preferably download video data pieces from the neighbor peer in the same AS that usually has better performance than other peers, e.g., shorter RTT. The feature of HDIS that forces a P2P streaming application to download the data from a peer in the same LAN is very significant. To the best of my knowledge, no other research shows the same result.

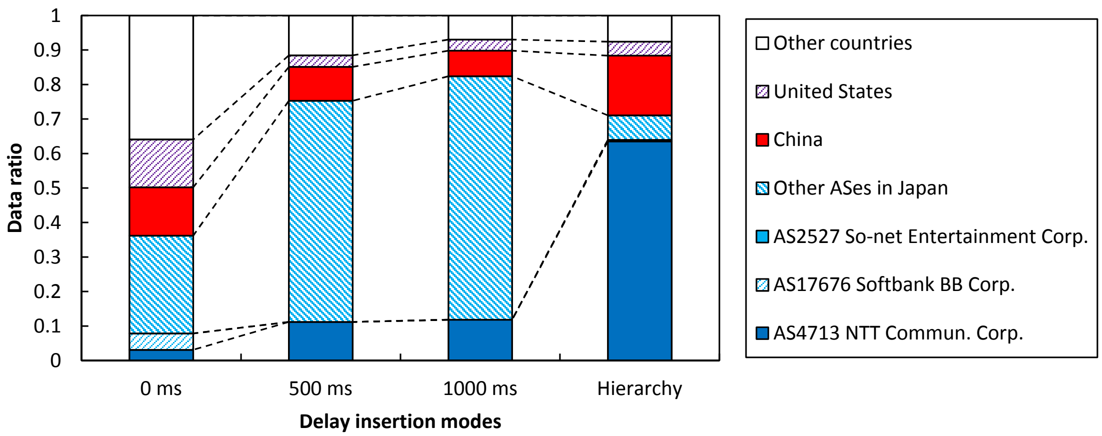



Figure 6.10: Downloaded data distributions for SopCast in four modes of delay insertion.

Figure 6.10 presents the average downloaded data distributions in four delay insertion modes of P2P-HDISTO. The vertical axis represents the region-by-region ratios for the downloaded traffic on the measurement host. I showed the ratios by countries for the traffic from the outside of Japan, and by ASes/ISPs for the traffic inside Japan. I grouped the information of AS and ISP together in the results because I had not found any traffic coming from different AS in the same ISP in the experiments. It can be seen that HDIS significantly increases the traffic inside the same AS as the measurement host, AS4713 NTT Communications Corp., i.e., it significantly reduces the cross-AS/ISP traffic. The traffic from the outside of AS4713 NTT Communications Corp. is marked as the cross-AS/ISP traffic. Without delay insertion, i.e., with the original behavior of SopCast, the cross traffic accounts for approximately 98% of the total traffic. This is a very high percent of inter-domain traffic. Such the cross-domain traffic accounts for 93%, 89%, and 36% of the total traffic in using the delay insertion with 500 ms, 1000 ms, and the hierarchy, respectively. The total traffic inside Japan of HDIS is lower than that of the 1000 ms delay insertion mode. However, this is reasonable because the delay lengths inserted to some Asian peers were much less than 1000 ms as shown in Table 6.1.

Figure 6.11 illustrates the neighbor peer distributions by three schemes, where the vertical axis represents the region-by-region ratios of the number of peers

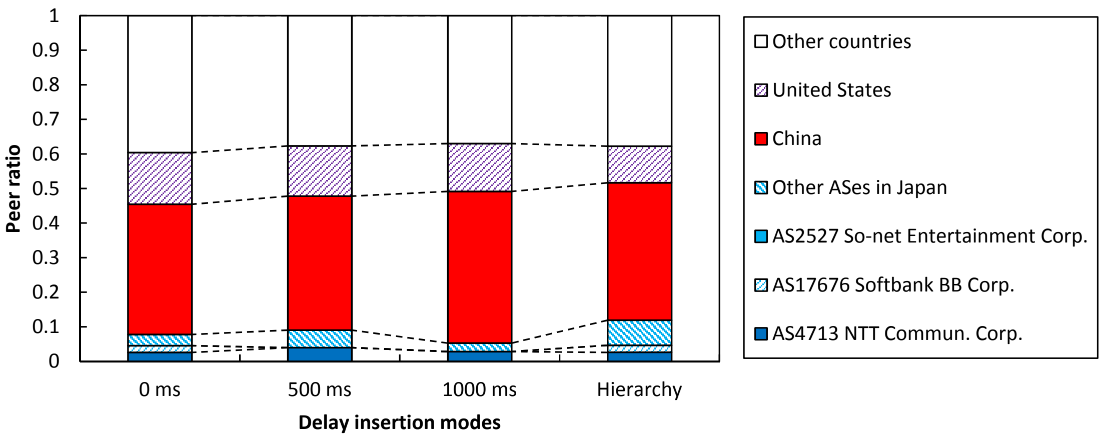



Figure 6.11: Neighbor peer distributions for SopCast in four modes of delay insertion.

that the measurement host communicated with. Figure 6.11 indicate that the neighbor peer distributions do not vary much and almost independent of the delay insertion. This can be explained as follows: SopCast first contacts with some peer list servers to obtain a list of available online peers, and then forms an overlay network for exchanging video data pieces with a subset of those peers. HDIS, however, cannot intervene at the step of obtaining the peer list. The neighbor peer distributions are therefore pretty stable. Nevertheless, the application preferably selects closer peers to download data pieces even when very few candidates exist. HDIS thus successfully realizes traffic localization on SopCast.

Table 6.2: Average waiting time of SopCast.

| **Delay insertion modes** | **Average waiting time [s]** |
|---|---:|
| Without delay insertion | 12.30 |
| 500 ms | 22.48 |
| 1000 ms | 33.46 |
| **Hierarchy** | **13.10** |

It is an important point to prove that HDIS will be better than fixed-length delay insertion from the viewpoint of the quality performance of P2P applications. Since the fixed-length scheme does not consider the number of peers in the same

AS, ISP, and country as the measurement peer, it always inserts delay into foreign peers even when there is no Japanese peer for localizing. This will cause the quality degradation of P2P applications. Table 6.2 highlights the average waiting time for SopCast in four delay insertion modes. Waiting time is the time a user has to wait after clicking the play button till starting to watch the video. It is clearly evident that the fixed-length delay insertion method has made degradation on SopCast performance. In comparison with the original behavior, the waiting time is almost doubled in case of 500 ms and tripled in case of 1000 ms delay insertion. In contrast, the waiting time of the proposed hierarchical method is almost the same as that of the original behavior.

From the statistics above, I conclude that HDIS successfully suppresses the cross-AS/ISP/country traffic and also maintains the quality performance of SopCast.

**(2) Results with PPStream**

Table 6.3: An example of inserted additional delay length for PPStream. The last digits of IP addresses are anonymized.

| **Country** | Japan | | | | United States | |
|---|---|---|---|---|---|---|
| **ISP name** | NTT Commun. Corp. | Softbank BB Corp. | KDDI Corp. | SAKURA Internet Inc. | AT&T Services Inc. | Cablevision Systems Corp. |
| **AS number** | AS4713 | AS17676 | AS2516 | AS9371 | AS7018 | AS6128 |
| **Peers' IP addresses** | 153.183.143.***, 153.183.64.***, 180.0.59.***, ... | 60.71.179.***, 60.71.209.***, 60.71.155.***, ... | 59.134.247.***, 59.137.143.*** | 153.120.216.*** | 12.205.168.*** | 32.160.18.*** |
| **Additional delay length [ms]** | **0** | **397** | **237** | **200** | **1393** | **1557** |

Secondly, I present the results obtained with PPStream in a similar manner to SopCast case. Table 6.3 shows an example of delay lengths assigned to peers located in different areas. Figures 6.12 and 6.13 present the downloaded data distributions and the neighbor peer distributions in four delay insertion modes, respectively. For the foreign traffic, I showed the data ratios of China and the United State, and bundled the other countries in a group. The amount of data received from the same AS as the measurement host, AS4713 NTT Communications Corp., increases significantly in applying HDIS. The cross-AS/ISP traffic

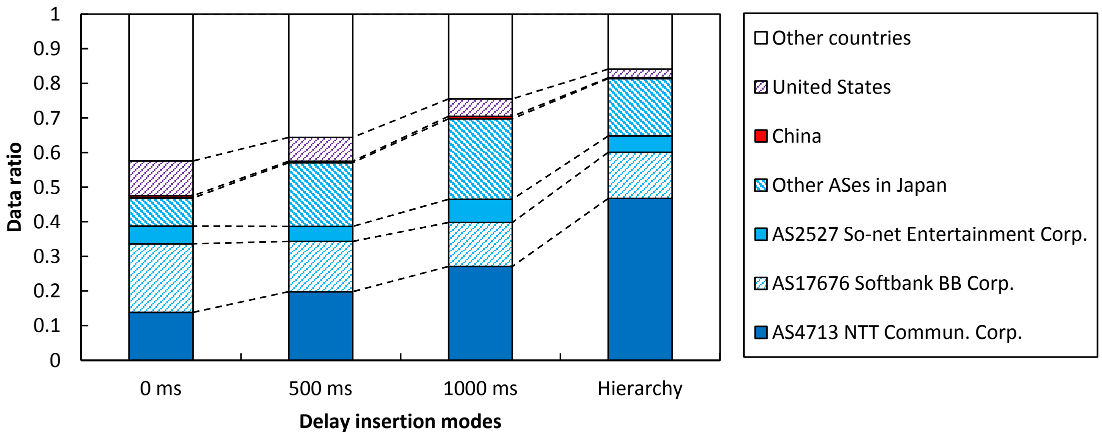



Figure 6.12: Downloaded data distributions for PPStream in four modes of delay insertion.

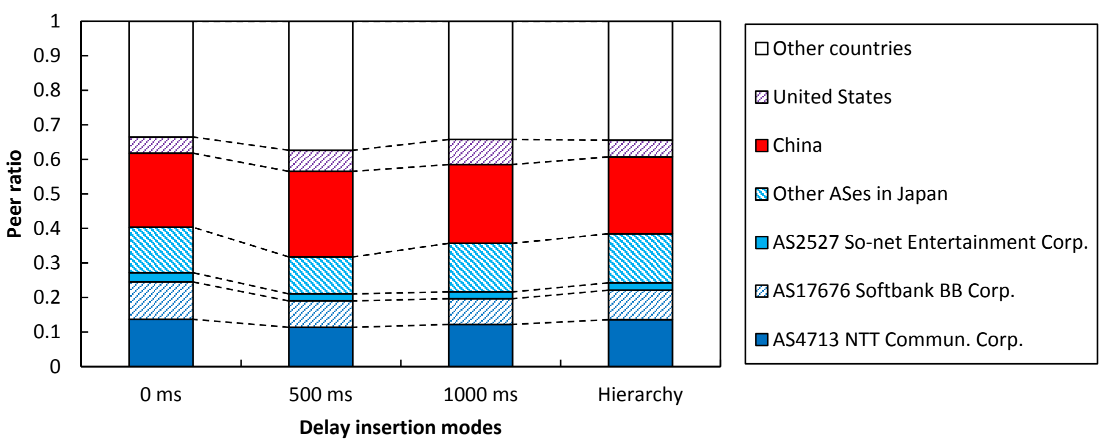



Figure 6.13: Neighbor peer distributions for PPStream in four modes of delay insertion.

accounts for approximately 86%, 80%, and 72% of the total traffic in case of 0 ms, 500 ms, and 1000 ms delay insertion modes, respectively. On the other hand, such traffic accounts for only 53% of the total traffic in the proposed hierarchical method. This statistic proves that HDIS significantly suppresses the cross-domain traffic. HDIS thus realizes traffic localization on PPStream.

The results in Fig. 6.13 also indicate that the neighbor peer distributions of PPStream are pretty stable. Table 6.4 shows the average waiting time in four modes of delay insertion. The results show that the waiting time was almost the same in four modes. This can be understood because many Japanese peers appeared in this experiment of PPStream as shown in Fig. 6.13.

Table 6.4: Average waiting time of PPStream.

| **Delay insertion modes** | **Average waiting time [s]** |
|---|---:|
| Without delay insertion | 20.51 |
| 500 ms | 21.06 |
| 1000 ms | 20.38 |
| **Proposed method** | **20.94** |

### (3) Results with PPTV

Table 6.5: An example of inserted additional delay length for PPTV. The last digits of IP addresses are anonymized.

| **Country** | Japan | | | | China | | United States |
|---|---|---|---|---|---|---|---|
| **ISP name** | NTT Commun. Corp. | Softbank BB Corp. | Asahi Net | eMobile Ltd. | Chinanet | CNCGROUP China169 Backbone | Metropolitan Telecomm |
| **AS number** | AS4713 | AS17676 | AS4685 | AS37903 | AS4134 | AS4837 | AS16524 |
| **Peers' IP addresses** | 180.47.102.***, 114.161.189.***, 222.145.100.***, ... | 221.84.103.***, 126.108.203.***, 126.215.177.*** | 183.77.250.*** | 117.55.68.*** | 121.10.44.***, 121.10.20.***, 219.130.193.***, ... | 124.160.184.***, 124.160.184.***, 124.160.184.***, ... | 72.11.221.*** |
| **Additional delay length [ms]** | **0** | **224** | **182** | **179** | **643** | **548** | **1395** |

Finally, I show the results obtained with PPTV, another popular video-on-demand application. Table 6.5 demonstrates an example of delay length applied to peers in different regions. Figures 6.14 and 6.15 present the downloaded data distributions and the neighbor peer distributions in four delay insertion modes. The results are similar to those of SopCast and PPStream, which leads to the fact

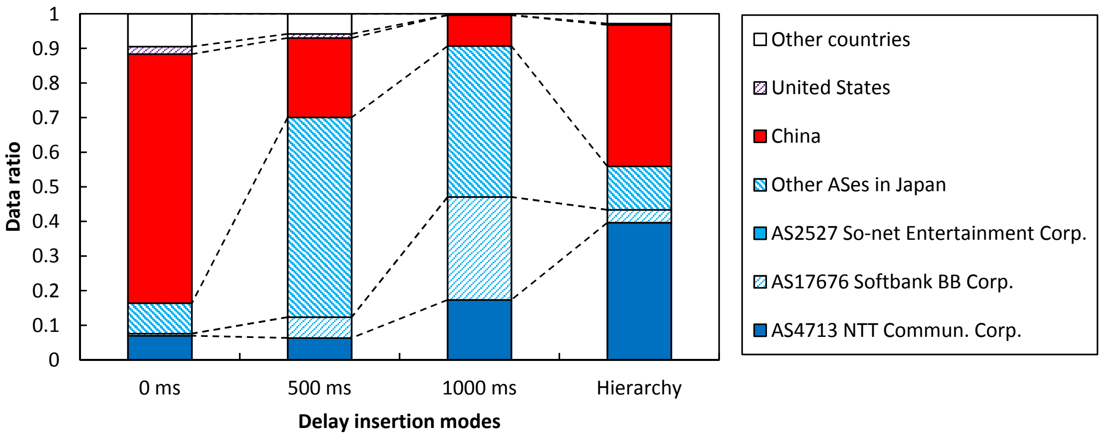



Figure 6.14: Downloaded data distributions for PPTV in four modes of delay insertion.

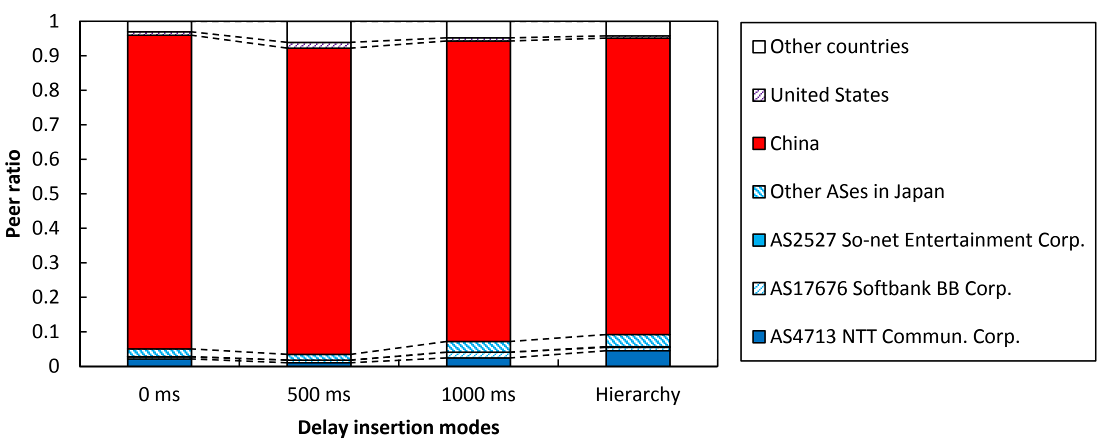



Figure 6.15: Neighbor peer distributions for PPTV in four modes of delay insertion.

that the communication protocols of three applications are probably very similar. From Fig. 6.14, the cross-AS/ISP traffic accounts for 92%, 94%, and 82% in case of 0 ms, 500 ms, and 1000 ms delay insertion modes, respectively. On the other hand, the proposed method substantially reduces such kind of traffic down to 60%.

The results in Fig. 6.15 highlight that the neighbor peer distributions are also stable, and that many peers exist in China. This may cause a degradation of PPTV performance in case of FDIS. As expected, the results shown in Table 6.6 indicate that the average waiting time increases in increasing additional delay length, whereas the waiting time remains low with the proposed method. HDIS thus realizes traffic localization and maintains quality performance of PPTV.

Table 6.6: Average waiting time of PPTV.

| **Delay insertion modes** | **Average waiting time [s]** |
|---|---:|
| Without delay insertion | 17.64 |
| 500ms | 21.76 |
| 1000ms | 25.50 |
| **Proposed method** | **17.42** |

### 6.4.4 Results of Forcing Packet Loss Method

To evaluate the effectiveness of the forcing packet loss method, I compared the results of three different schemes:

- (1) random and/or RTT-based peer-selection scheme, i.e., the original behavior of P2PTV applications.
- (2) FPLS, fixed-length packet loss scheme, in which every packet exchanging with foreign peers will be forced to lost with a given probability, e.g., 10% and 20%.
- (3) HPLS, hierarchical packet loss scheme.

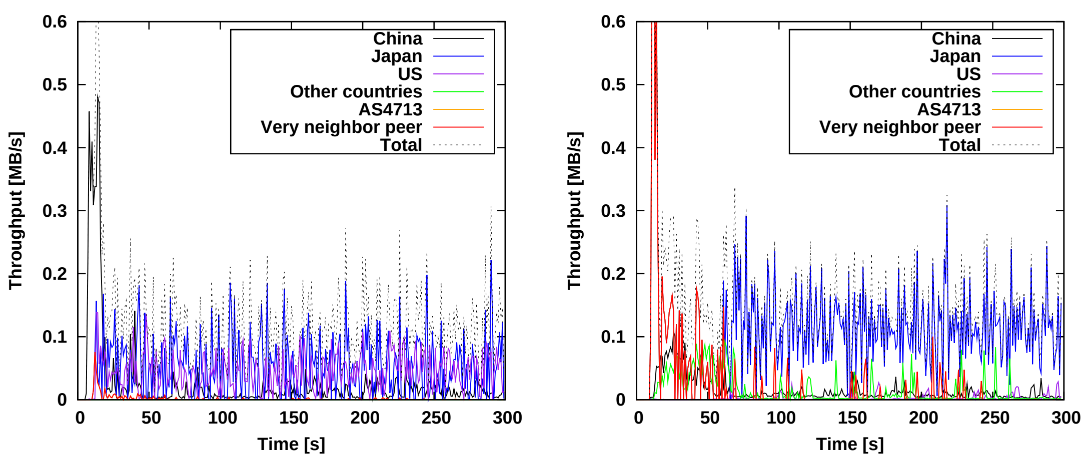
Scenarios for all experiments of HPLS are the same as those of HDIS. Since the communication performance in P2P networks would be measured in terms of not only latency but also packet loss probability, it is not surprising that the experimental results of HPLS are very similar to those of HDIS.

### (1) Results with SopCast

Table 6.7: An example of packet loss rate assigned to peers in SopCast. The last digits of IP addresses are anonymized.

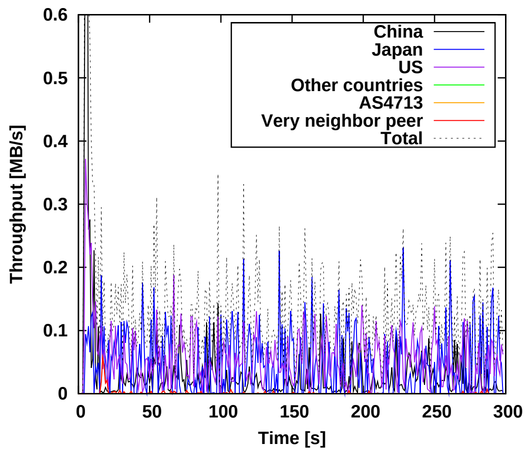
| **Country** | Japan | | | China | United States |
|---|---|---|---|---|---|
| **ISP name** | NTT Commun. Corp. | KDDI Corp. | Softbank BB Corp. | China Net | Cox Commun. Inc. |
| **AS number** | AS4713 | AS2516 | AS17676 | AS4134 | AS22773 |
| **Peers' IP addresses** | **27.114.109.*** ** | 121.105.201.*** | 126.108.203.*** | 119.145.196.*** | 68.5.224.*** |
| **Packet loss rate** | **0** | **0.11** | **0.16** | **0.2** | **0.45** |

Table 6.7 shows an example of packet loss rate assigned to several peers in different regions.

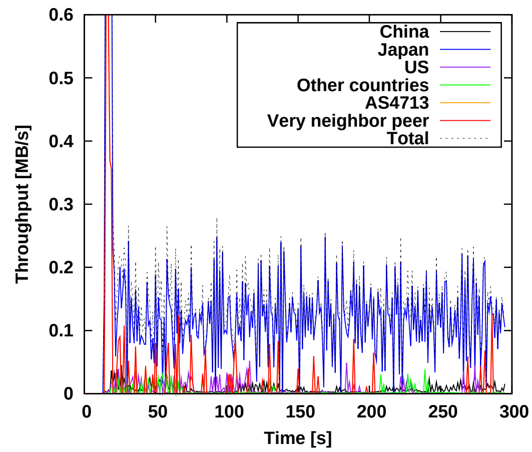



(a) Keep original behavior of SopCast on host 1. (b) Apply fixed-length 10% packet loss on host 2.

Figure 6.16: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying fixed-length 10% packet loss on the measurement host 2.

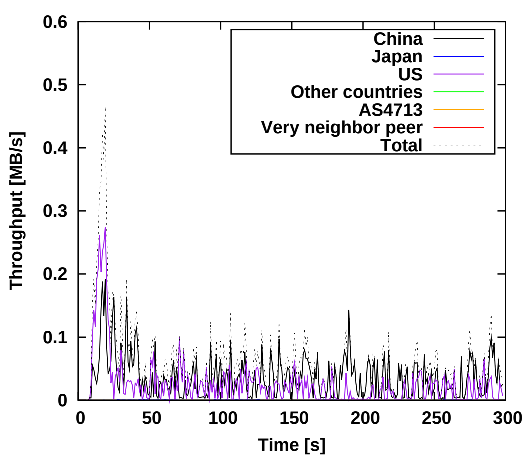



(a) Keep original behavior of SopCast on host 1.

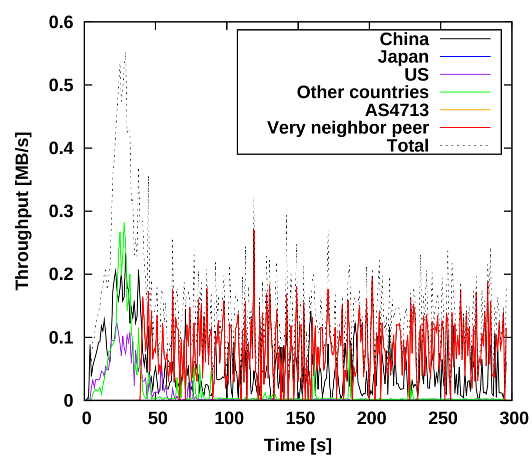



(b) Apply fixed-length 20% packet loss on host 2.

Figure 6.17: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying fixed-length 20% packet loss on the measurement host 2.




(a) Keep original behavior of SopCast on host 1.




(b) Apply hierarchical packet loss on host 2.

Figure 6.18: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying hierarchical packet loss scheme on the measurement host 2.

Figures 6.16 and 6.17 show examples of temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying the FPLS on the measurement host 2. In case of applying fixed-length 10% and 20% packet loss, most traffic measured on host 2 comes from Japan. The increasing of traffic inside Japan proves that the fixed-length packet loss scheme successfully realizes traffic localization on SopCast. However, the traffic coming from the very neighbor peer, host 1, is very small.

Figure 6.18 shows an example of temporal changes of throughput when keeping original behavior of SopCast on the measurement host 1 and applying HPLS on the measurement host 2. Similar to that shown in Fig. 6.9, almost all the traffic measured on host 2 is downloaded from the very neighbor peer, host 1, when applying HPLS. In contrast, at the same time host 1 could not download any video data from its very neighbor peer, host 2. This is because HPLS has degraded network performance of inter-AS connections by forcing packet loss. Therefore, SopCast tends to preferably download video data pieces from the neighbor peer in the same AS that usually has better performance than other peers.

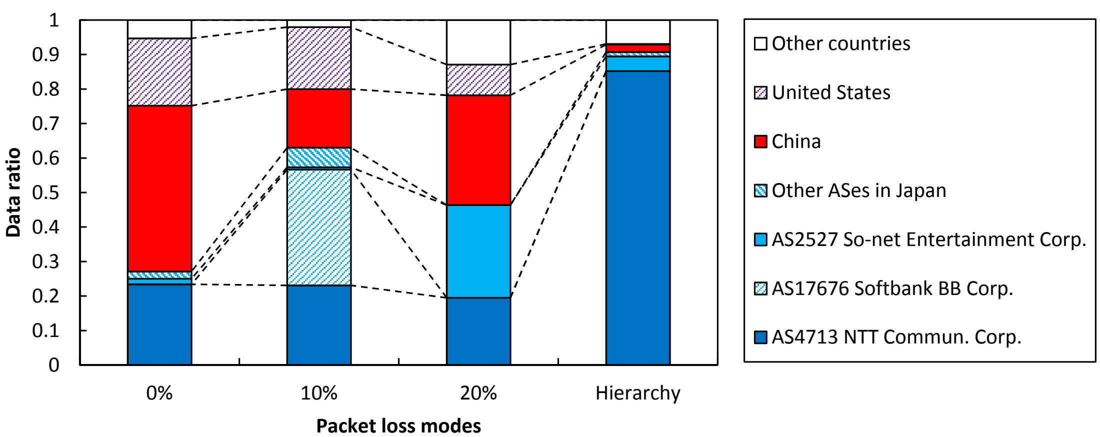



Figure 6.19: Downloaded data distributions for SopCast in four modes of forcing packet loss.

Figures 6.19 and 6.20 present the downloaded data distributions and the neighbor peer distributions in four packet loss modes including 0%, i.e., keeping the original behavior of SopCast, 10% of packet loss, 20% of packet loss, and the hierarchical packet loss scheme. The results resemble those of HDIS:

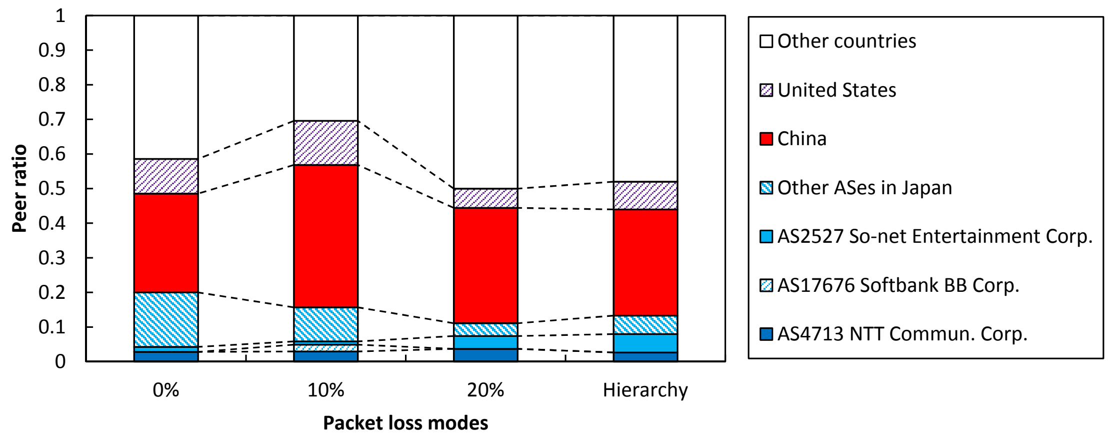



Figure 6.20: Neighbor peer distributions for SopCast in four modes of forcing packet loss.

even though the neighbor peer distributions do not vary and very few peers exist in the same AS as the measurement host, SopCast tends to remove connection paths to inter-domain peers and accordingly achieved the localization of traffic. In particular, the cross-AS/ISP traffic accounts for 76%, 77%, and 80% in case of 0%, 10%, and 20% packet loss rate, respectively. On the other hand, HPLS significantly reduces such the cross-domain traffic down to 15%.

Table 6.8: Average waiting time of SopCast.

| **Forcing packet loss modes** | **Average waiting time [s]** |
|---|---|
| Without packet loss | 12.30 |
| 10% | 24.25 |
| 20% | 34.20 |
| **Hierarchy** | **14.15** |

Table 6.8 highlights the average waiting time for SopCast in four packet loss modes. It is easy to see that the FPLS degrades the performance of SopCast.

From the statistics above, I conclude that HPLS successfully reduces the cross-AS/ISP/country traffic and also maintains the quality performance of SopCast.

**(2) Results with PPStream**

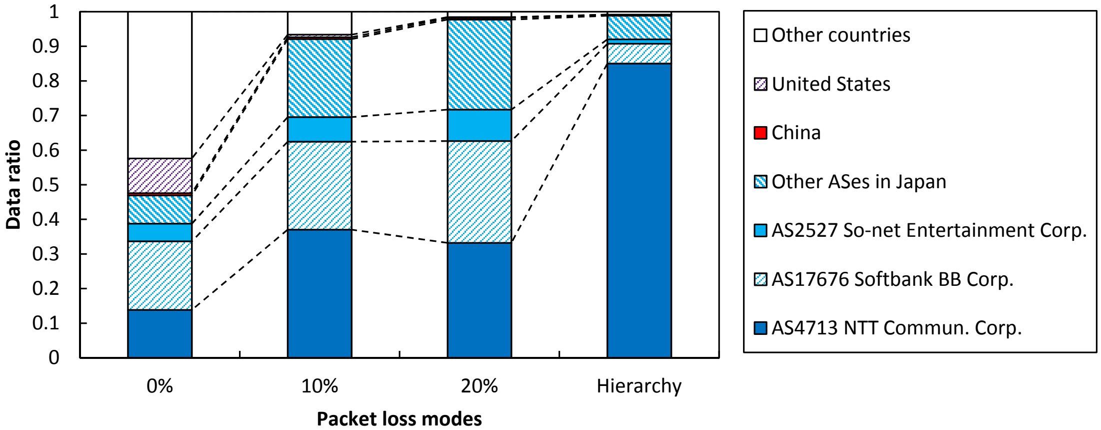



Figure 6.21: Downloaded data distributions for PPStream in four modes of forcing packet loss.

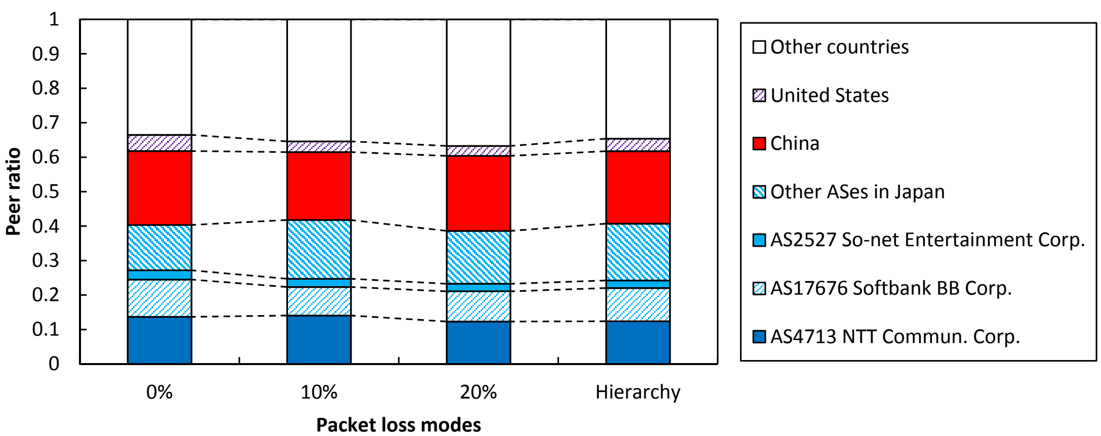



Figure 6.22: Neighbor peer distributions for PPStream in four modes of forcing packet loss.

Figures 6.21 and 6.22 present the downloaded data distributions and the neighbor peer distributions in four packet loss modes, respectively. The interpretation of the results in Figures 6.21 and 6.21 is simple: the amount of data received from the AS as the measurement host, AS4713 increases significantly when applying hierarchical packet loss scheme, whereas the neighbor peer distributions do not vary. In particular, the cross-AS/ISP traffic accounts for approximately 86%, 63%, and 67% of the total traffic in case of 0%, 10%, and 20% packet loss rate, respectively. Such the cross-domain traffic accounts for only 15% of the total traffic in the proposed hierarchical method.

Table 6.9: Average waiting time of PPStream.

| **Forcing packet loss mode** | **Average waiting time [s]** |
|---|---:|
| Without packet loss | 22.30 |
| 10% | 24.10 |
| 20% | 25.40 |
| **Proposed method** | **23.96** |

Table 6.9 shows the average waiting time in four modes of forcing packet loss. The results show that the waiting time is just changed a little bit with different packet loss schemes. In particular, the fixed-length packet loss scheme does not degrade the performance of PPStream much. This can be explained as follows: fixed-length packet loss scheme causes PPStream difficult to connects to foreign peers. However, PPStream can easily contact with some Japanese peers at the initial phase since many Japanese peers appeared in the experiment.

### (3) Results with PPTV

Figures 6.23 and 6.24 present the downloaded data distributions and the neighbor peer distributions in four packet loss modes. The results are similar to those of SopCast and PPStream. From Fig. 6.14, the cross-AS/ISP traffic accounts for 99%, 94%, and 90% in case of 0%, 10%, and 20% packet loss rate, respectively. On the other hand, the proposed method substantially reduces such kind of traffic down to 78%.

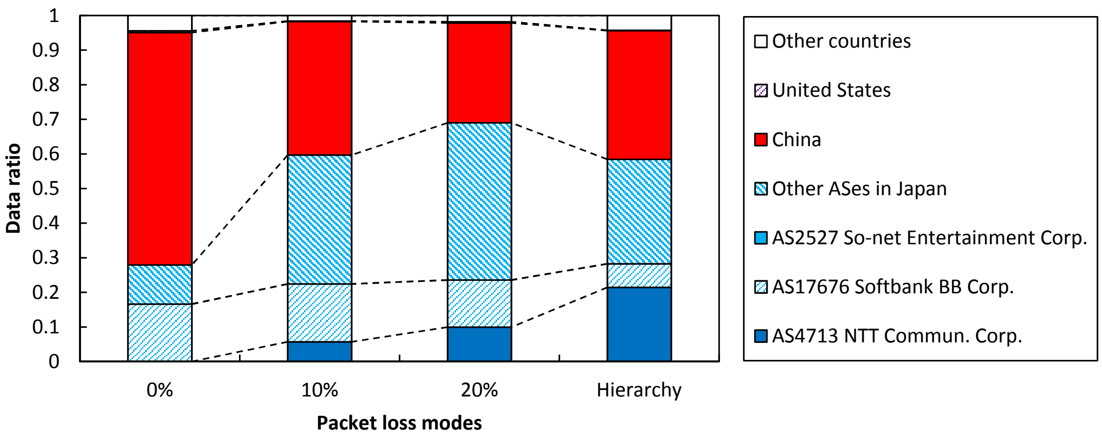



Figure 6.23: Downloaded data distributions for PPTV in four modes of forcing packet loss.

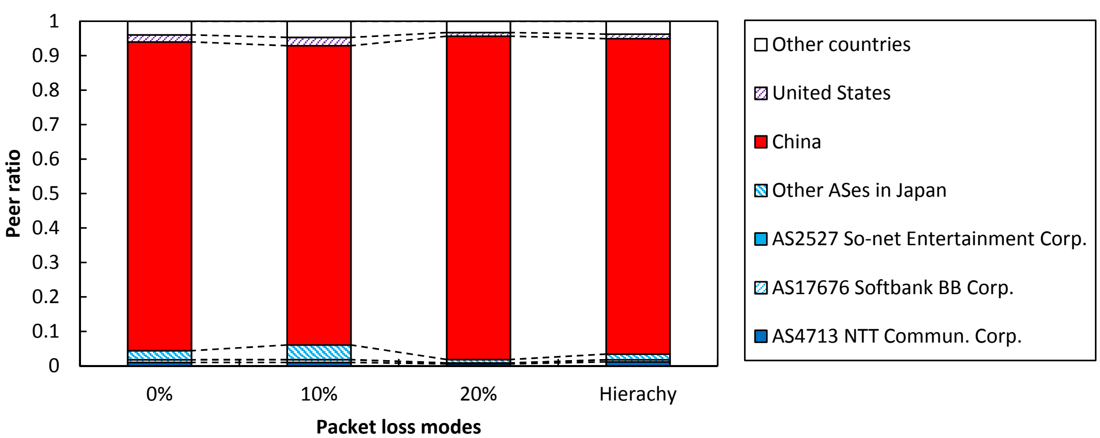



Figure 6.24: Neighbor peer distributions for PPTV in four modes of forcing packet loss.

The results in Fig. 6.24 highlight that the neighbor peer distributions are also steady, and that many peers exist in China. This causes a degradation of PPTV performance in case of fixed-length packet loss scheme as shown in table 6.10.

Table 6.10: Average waiting time of PPTV.

| **Forcing packet loss modes** | **Average waiting time [s]** |
|---|---|
| Without packet loss | 16.50 |
| 10% | 24.60 |
| 20% | 27.50 |
| **Proposed method** | **17.30** |

### 6.4.5 Results of Bandwidth Limitation Method

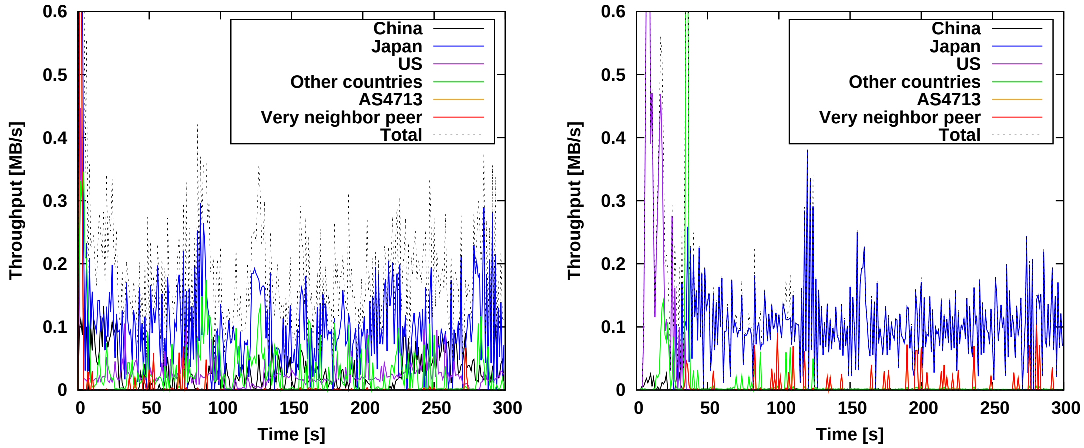
I compared the results of three following schemes:

- (1) random and/or RTT-based peer-selection scheme, i.e., the original behavior of P2PTV applications.
- (2) FBLS, fixed-length bandwidth limitation scheme, in which all the connections to foreign peers were limited by a constant value, e.g., 800kbps and 1000kbps. I selected 800kbps and 1000kbps as limited values because 800kbps and 1000kbps are approximate equal to the average bit rates of the three video streams of SopCast, PPStream, and PPTV.
- (3) HBLS, hierarchical bandwidth limitation scheme.

Scenarios for all experiments of HBLS are the same as those of HDIS and HPLS. Since the results resemble those of HDIS and HPLS, I just present some statistics.

**(1) Results with SopCast**

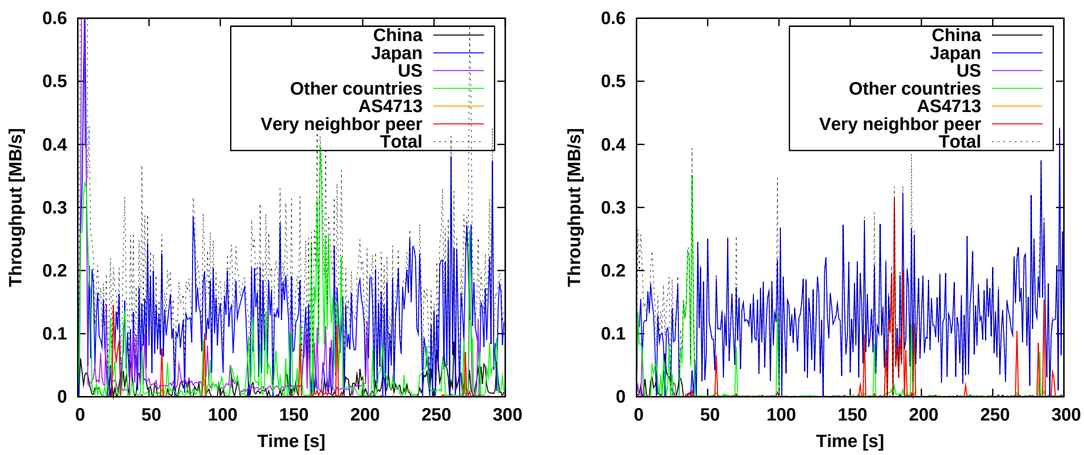
Figures 6.25 and 6.26 show examples of temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying the FBLS on the measurement host 2. In case of applying FBLS,




(a) Keep original behavior of SopCast on host 1.

(b) Apply fixed-length 800 kbps bandwidth limitation on host 2.

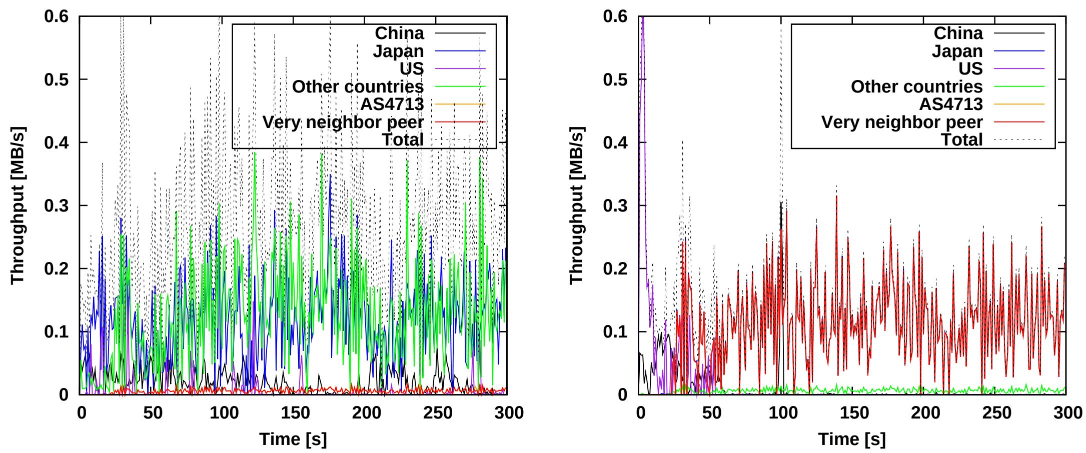
Figure 6.25: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying fixed-length 800 kbps bandwidth limitation on the measurement host 2.




(a) Keep original behavior of SopCast on host 1.

(b) Apply fixed-length 1000 kbps bandwidth limitation on host 2.

Figure 6.26: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying fixed-length 1000 kbps bandwidth limitation on the measurement host 2.




(a) Keep original behavior of SopCast on host 1. (b) Apply hierarchical bandwidth limitation on host 2.

Figure 6.27: Temporal changes of throughput when simultaneously keeping original behavior of SopCast on the measurement host 1 and applying hierarchical bandwidth limitation on the measurement host 2.

most traffic measured on host 2 comes from Japan. This proves that FBLS realizes traffic localization on SopCast. However, the traffic coming from the very neighbor peer, host 1, is very small.

Figure 6.27 shows an example of temporal changes of throughput when keeping original behavior of SopCast on the measurement host 1 and applying HBLS on the measurement host 2. It can be seen that almost all the traffic measured on host 2 is downloaded from the very neighbor peer, host 1, when applying HBLS. In contrast, at the same time host 1 could not download any video data from its very neighbor peer, host 2.

Figures 6.28 and 6.29 present the downloaded data distributions and the neighbor peer distributions in four modes of bandwidth limitation. Even though the neighbor peer distributions do not vary as shown in Fig. 6.29, the amount of data received from the AS4713 NTT Communications Corp. increases significantly in applying HBLS. In particular, the cross-AS/ISP traffic accounts for 95%, 95%, and 94% in case of no limit, 800kbps, and 1000kbps bandwidth limitation, respectively. Such the cross traffic dramatically decreases in case of hier-

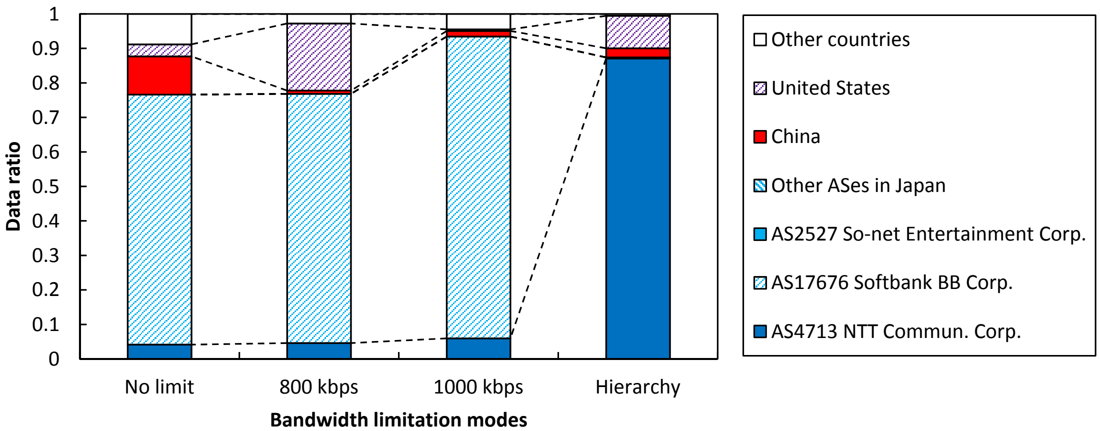



Figure 6.28: Downloaded data distributions for SopCast in four modes of bandwidth limitation.

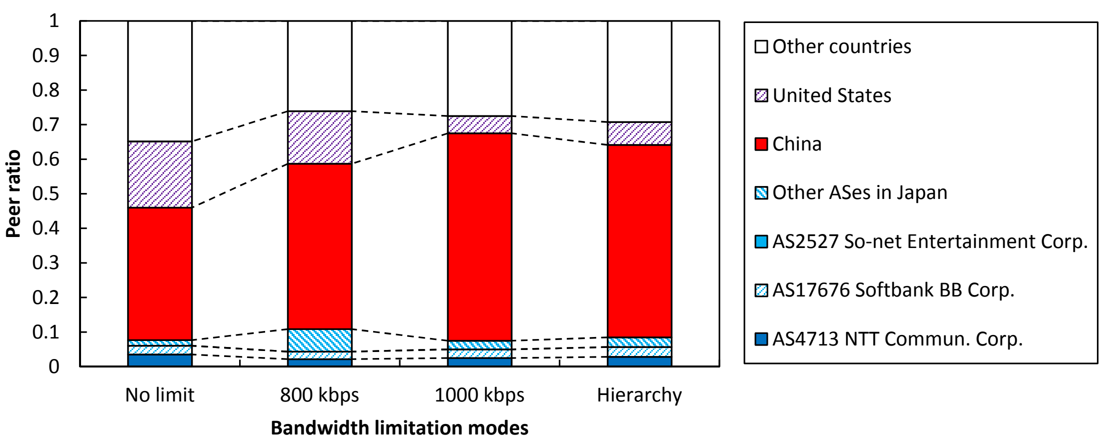



Figure 6.29: Neighbor peer distributions for SopCast in four modes of bandwidth limitation.

archical bandwidth limitation scheme, accounts for only 13% of the total traffic. This statistic indicate that the proposal successfully realizes traffic localization on SopCast.

Table 6.11: Average waiting time of SopCast.

| **Bandwidth limitation modes** | **Average waiting time [s]** |
|---|---:|
| No limit | 13.45 |
| Fixed-length (800kbps) | 42.50 |
| Fixed-length (1000kbps) | 45.20 |
| Hierarchy | 14.00 |

Table 6.11 shows the average waiting time for SopCast by three schemes. It is easy to see that the fixed-length bandwidth limitation scheme degrades the performance of SopCast. In comparison with the original behavior of SopCast, the waiting time is much longer in case of 800kbps and 1000kbps bandwidth limitation. This is because the FBLS always limits the bandwidth of overseas traffic even when very few Japanese peers exist for localizing. In the worst case, no Japanese peer for contact, the waiting time of SopCast will be very long due to bandwidth limitation applied to overseas traffic. In contrast, the waiting time of the hierarchical bandwidth limitation scheme is almost the same as that of the original behavior. These results show the effectiveness of the hierarchical feature. In case of no Japanese peer for contact, no bandwidth limitation is applied as described in previous section. Therefore, I conclude that the hierarchical bandwidth limitation scheme also maintains the quality performance of SopCast.

**(2) Results with PPStream**

Figures 6.30 and 6.31 present the downloaded data distributions and the neighbor peer distributions for PPStream in four modes of bandwidth limitation, respectively. Even though the neighbor peer distributions do not vary as shown in Fig. 6.31, the cross-domain traffic reduces significantly when applying bandwidth limitation. In particular, the cross-AS/ISP traffic accounts for approximately 86%, 70%, and 73% of the total traffic in case of no limit, 800kbps,

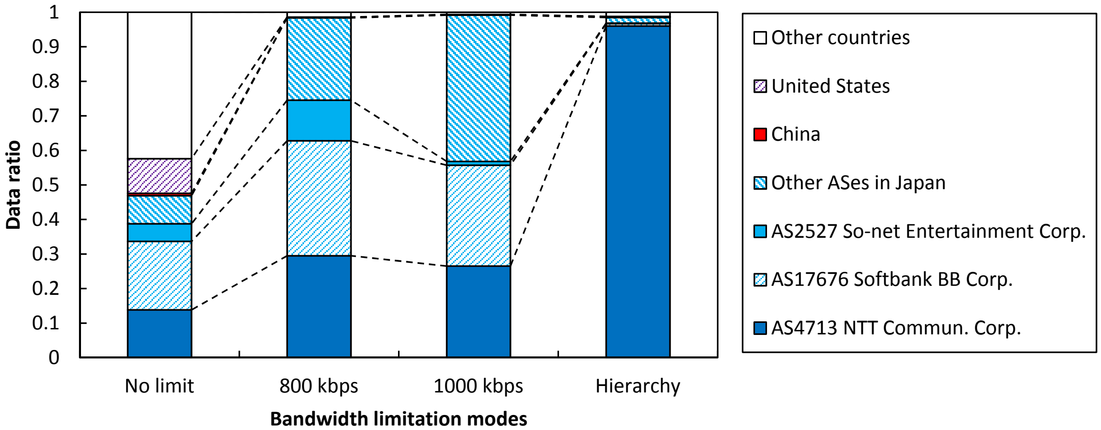



Figure 6.30: Downloaded data distributions for PPStream in four modes of bandwidth limitation.

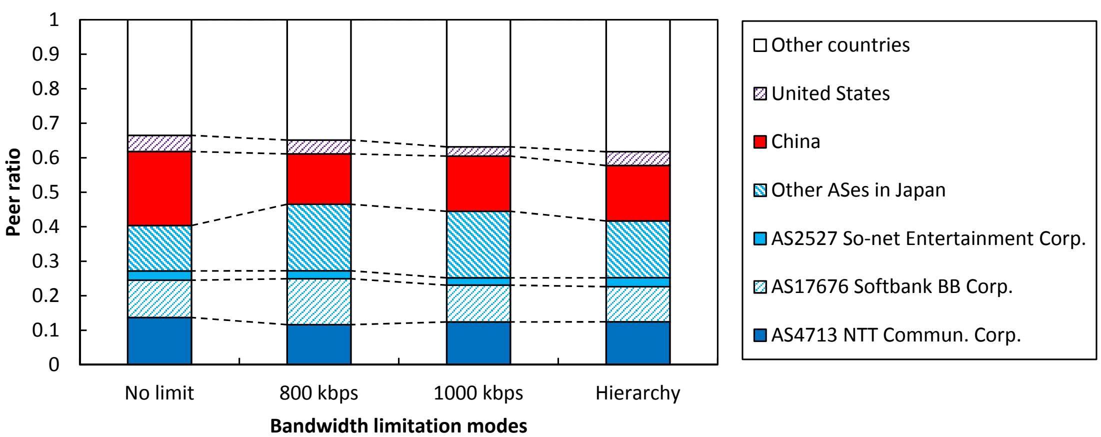



Figure 6.31: Neighbor peer distributions for PPStream in four modes of bandwidth limitation.

and 1000kbps bandwidth limitation, respectively. Such the cross-domain traffic accounts for only 4% of the total traffic in case of hierarchical scheme.

**(3) Results with PPTV**

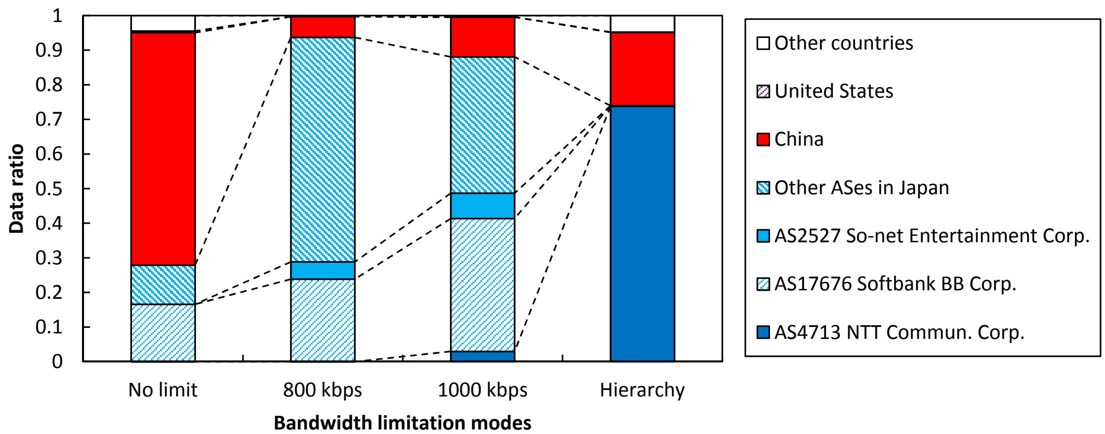



Figure 6.32: Downloaded data distributions for PPTV in four modes of bandwidth limitation.

Figures 6.32 and 6.33 present the downloaded data distributions and the neighbor peer distributions for PPTV in four modes of bandwidth limitation, respectively. The neighbor peer distributions are stable and almost independent of bandwidth limitation scheme as shown in Fig. 6.33. The cross-AS/ISP traffic accounts for approximately 99%, 99%, and 97% of the total traffic in case of no limit, 800kbps, and 1000kbps bandwidth limitation, respectively. Such the cross-domain traffic accounts for only 26% of the total traffic in case of hierarchical scheme.

## 6.5 Impact of the Proposed Router to Network

This section discusses about the impact of the proposed router to the network. Figure 6.34 shows a network situation when only router R1 inside my laboratory is replaced by the proposed router. I assume that the video server is located in ISP 2, different ISP of our laboratory's ISP. It means that some local peers in ISP 1 have to connect to the server or some other peers in ISP 2 to download the video

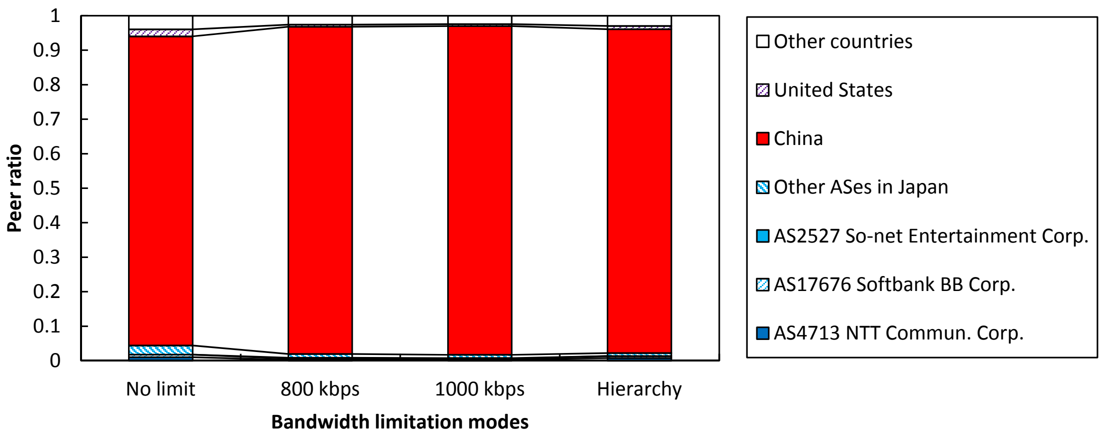



Figure 6.33: Neighbor peer distributions for PPTV in four modes of bandwidth limitation.

data. Since connection paths from peer A to outside of ISP 1 have delay caused by the proposed router, peer A tends to download the video data from local peers in the same ISP, e.g., peer B. Because R2 is a normal router, peer B can download video data from some peers in ISP 2 without the skewed artificial inserted delay, and therefore the probability that peer A can download the desired data from peer B or other local peers is very high. The proposed router can be configured to degrade the network performance of only the downstream. In other words, the probability that other peers can download desired data from peer A might not be affected by the proposed router. From the above analysis, I hypothesize that my proposed router generally has no impact to the network when it works at only one point. One exception remains: if all the peers in ISP 1 that peer A connects to have low upload capacity, the video quality viewed by peer A will be degraded.

Now, I consider the situation when all routers are replaced by the proposed router. In this case, all peers will tend to download video data from neighbor peers which are in the same ISP. This leads to over-localization problem resulting in poor video quality. Furthermore, since the video server is located in ISP 2, many peers in ISP 1 must connect to peers in ISP 2 to download the video data. However, the connections' network performance are degraded because of the proposed router routers, the performance of P2P applications are therefore

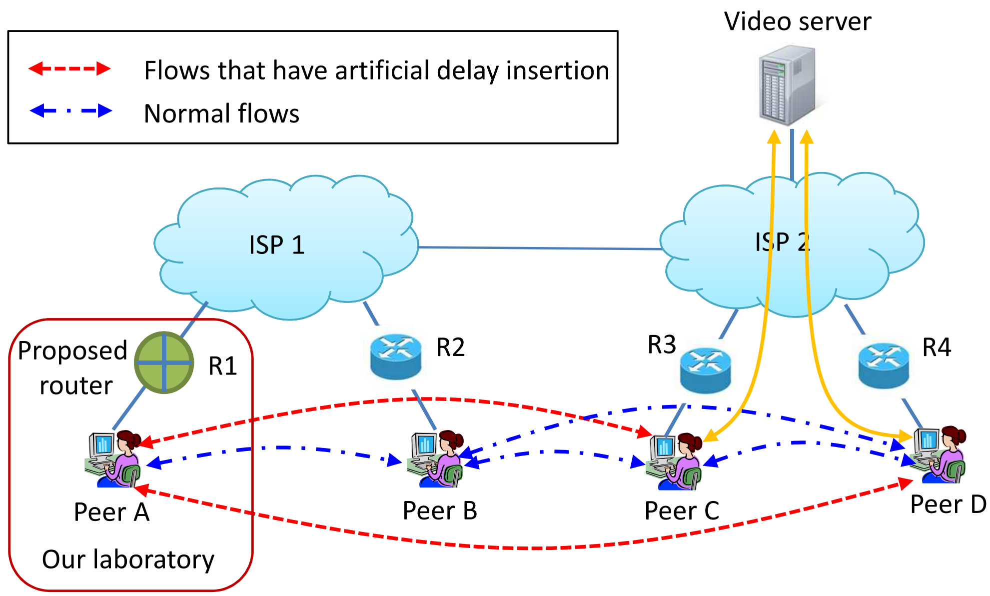



Figure 6.34: A network situation when P2P-HDISTO is deployed into real network.

much degraded. Considering a method to avoid over-localization and to balance the trade-off between localization and QoS is mandatory. This is one of the future works of this dissertation.

## 6.6 Discussion

Based on the experimental results performed with three different P2PTV applications, I conclude that the idea of degrading network performance of inter-domain traffic successfully realizes traffic localization on P2PTV systems. Furthermore, the most important point of the proposal is that the method does not require any modification of existing P2P applications. Therefore, I believe that it can be applied to other types of P2P such as file sharing systems.

The stability of the neighbor peer distributions indicates that three proposed methods (delay insertion, forcing packet loss, and bandwidth limitation) cannot intervene in the step of peer selection of P2P applications. The experimental results also prove that considering the number of peers in the same area as the querying peer helps to maintain the performance of P2P applications. However, this will face some difficulties when some P2P applications are running simultaneously. In this situation, it would be hard to recognize which peer belongs to which application; hence the logical distance computed by Eq. (6.4) may become wrong. Considering the format of peer list packets would be a simple way to filter the packets. Nevertheless, further study on traffic classification is essentially required in the future.

HBLS seems to be the most effective scheme since it reduces sharply the cross-domain traffic of SopCast, PPStream, and PPTV. On the other hand, I believe that the selection of parameters in equations from 1 to 7 affects to the performance of HDIS, HPLS, and HBLS. For instance, with the values of the parameters in equations as described in section 6.4: for HDIS, I have differentiated the traffic between inside and outside the AS/ISP with an additional delay of 100 ms at a maximum. The traffic coming from foreign peers was complemented with an additional delay of 200 ms at a maximum. Note that the delay length assigned to a foreign peer comprises an additional delay due to different AS (100 ms maximum), additional delay due to different ISP (100 ms maximum), complemented delay due to different country (200 ms maximum), and further additional delay by considering the physical distance. Although HDIS works well with these values of the parameters, further study on selection of the parameters must be taken into account to optimize the performance of HDIS, HPLS, and HBLS.

Finally, I discuss the applicability of the proposed method. To introduce proposed routers into the network, this thesis proposed to replace the conventional home gateway routers by the proposed routers as shown in Fig. 6.6. Experimental results indicate that the proposed router works well as a home router, where the number of packets flowing through the router is not so large. In the future, the proposed method might be applied in mobile environment, where users have no gateway router. In that situation, I want to consider the proposed router as a shaping device at the ISP side as shown in Fig. 6.35. The scalability problems in terms of the logical distance computation as well as the memory would occur. Therefore, it is necessary to consider the collaboration among proposed routers in which the delay insertion, forcing packet loss, or bandwidth limitation process is divided to some on-the-path routers. If a router is busy to process, it may relegate the task to the next-hop router.

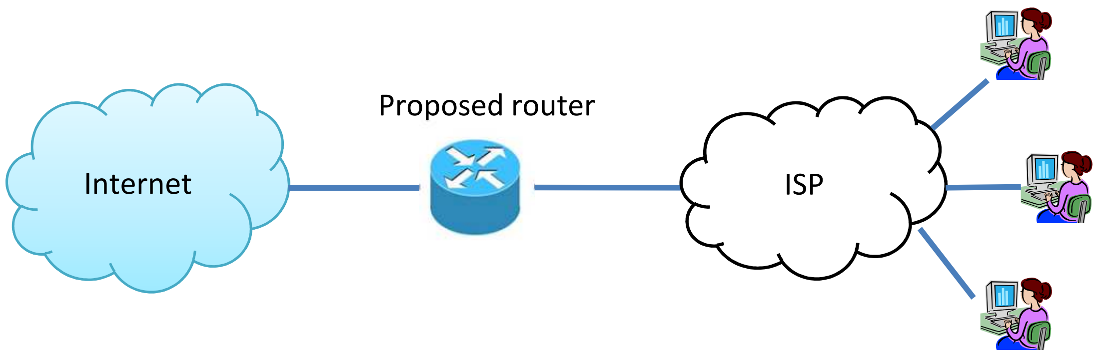



Figure 6.35: A scenario to introduce the proposed router into network in the future.

## 6.7 Conclusion

This chapter proposed a router-aided approach for hierarchical P2P traffic localization by degrading the quality of connection paths of inter-domain traffic. The traffic can be hierarchically localized with multiple levels such as AS level, ISP level, and country level. To achieve the hierarchy of localization, I introduced the logical distance between two peers based on not only the physical distance but also the number of peers exists in the same area as the querying peer. To deploy the idea of degrading network performance of inter-domain traffic, three different schemes were proposed including HDIS, HPLS and HBLS. The implementation in detail of three schemes is also introduced utilizing `libpcap`, `GeoLite`, and `dummynet` on a PC-based router with Linux OS. Experiments on three P2PTV applications demonstrated that the proposed method significantly reduces the cross-domain traffic even when few peers exist in the same area while maintaining the quality performance of the applications.

Several future challenges remain as described in Section 6.6. In the future, I will test the proposal on other types of P2P applications such as BitTorrent. I am also planning to study on P2P traffic classification as well as collaboration among the proposed routers.

# Chapter 7

# Peer List Sharing by Router Collaboration for Traffic Localization

## 7.1 Introduction

Chapter 4 has proposed a peer list modification scheme for P2P traffic localization without any peer reaction. Every packet flowing through a gateway router is first examined. Packets that contain the peer list sent by trackers or peer list servers will be modified for localizing before they arrive at their destination, i.e., the P2P application. Since the application then connects to some peers in the modified peer list, the traffic will therefore be localized without any modification of application software. For instance, to localize the traffic inside Japan, we can replace the foreign peers by Japanese peers in the peer list. The list of Japanese peers is not only collected at the current peer list but also accumulated from all the previous peer lists.

Since the list of local peers is collected at only one router, the peer list modification scheme will not work well if few local peers exist at that router. Let us consider the following example: two peers, A and B are simultaneously watching the same channel, and only peer A could find Japanese peers whereas the other could not find any Japanese peers for requesting video data pieces. In this situation, the peer list modification scheme that trying to localize the traffic inside Japan will not work well on peer B because we have no Japanese peer for replacing with foreign peers in the peer list. This chapter proposes that routers should collaborate with each other for sharing the list of local peers. Considering the above example, the router that peer A connects to will share the list of Japanese peers to the router that peer B connects to. By doing that, the peer list modification scheme can thus work well on peer B. With the router collaboration, the local peers are collected at not only one router but also many routers. Clearly, the peer list modification scheme will become much more effective in combining with router collaboration scheme because we will have more local peers in hand.

In this chapter, I just evaluate the effectiveness when combining the router collaboration and peer list modification, though. It does not mean that the router collaboration can not be combined with other scheme, e.g., video request packet redirection scheme introduced in chapter 5, for P2P traffic localization. I believe that the combination of router collaboration and video request redirection will have the same results as those of router collaboration combined with peer list modification.

## 7.2 Proposed scheme

Figure 7.1 shows the concept of the router collaboration scheme combined with the previous peer list modification scheme. The process is simple as follows: (1) Every packet flowing through each gateway router is examined; (2) If a packet contains the peer list, all the local peers, e.g., Japanese peers are collected and registered to the local peer list table; (3) Each gateway router then sends the local peer list table to the other routers for making a shared list of local peers; (4) Each gateway router modifies the peer list packet by replacing the foreign peers by local peers from the shared peer list table. Since the P2PTV does not recognize that the peer list packet has been already modified, the P2PTV simply contacts with some peers in the localized peer list, i.e., the traffic will be localized.

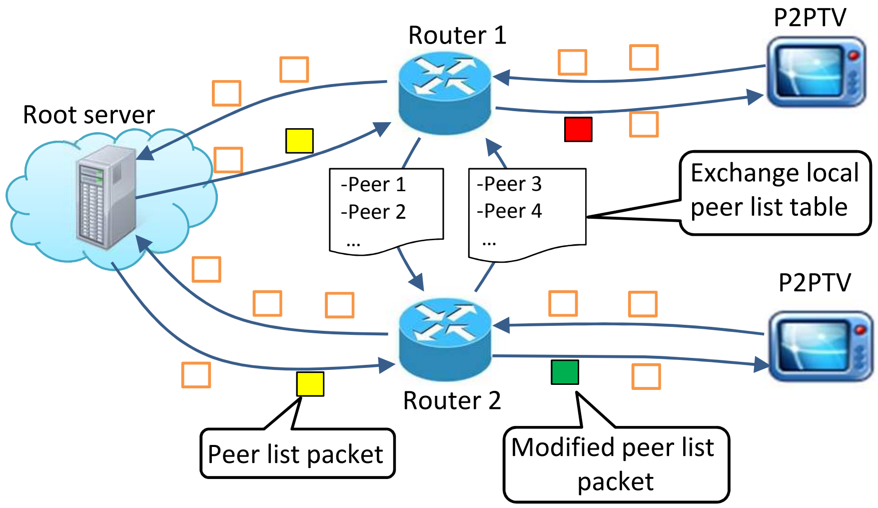



Figure 7.1: The concept of router collaboration scheme.

Figure 7.2 presents the proposed router architecture. Four modules, a packet monitoring module, a location identifier, a peer list modification module, and a communication module, are added into a common router. The packet monitoring module inspects every packet coming into the router to check if it contains the peer list or not. Only peer list packets go to the location identifier whereas the others go directly to the common routing function. The location identifier resolves the location information of all peers in the peer list by using several IP-to-Geographic-location databases. The location information and the peer list packet are then forwarded to the peer list modification module. The communication module communicates with other routers to send or receive a local peer list table. Finally, the peer list modification module modifies the peer list packets by replacing locality-unaware peers by local peers from the shared list.

## 7.3 Implementation

To evaluate the proposed router collaboration scheme, two desktop PCs were set up as two software routers. The hardware configuration of both routers are the same as follows: Intel Core i7-2600 3.4GHz CPU, 12 GB of DDR3 memory, and two 1 Gbps Ethernet network interface cards, operated under Linux Ubuntu

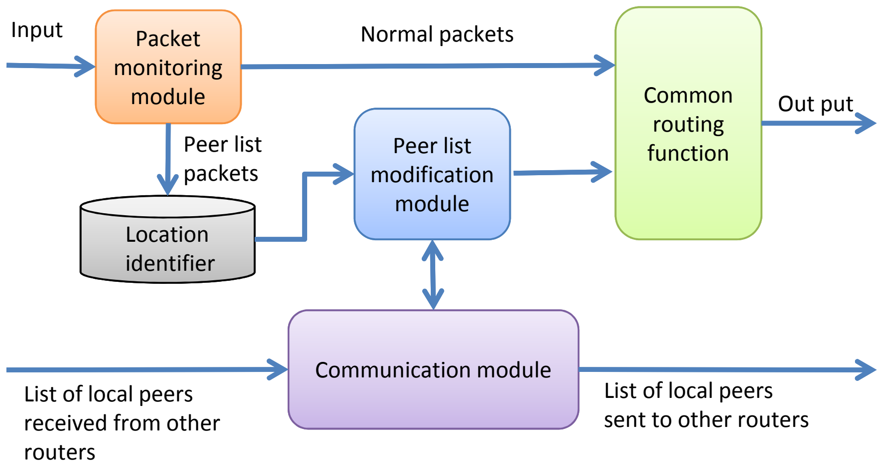



Figure 7.2: The proposed router architecture.

12.04 with 3.2.0-29 generic kernel. The implementation of four modules detail is as follows:

- For the packet monitoring module, all packets are first pushed into QUEUE by using `iptables` interface. `Libipq` is then used to read every packet from the QUEUE and to send it back to user space. As mentioned in chapter 4, the peer list packet can be easily recognized by finding the identifier, e.g., “01 0b” in the packet.

- For the location identifier: This study just checks the countries where peers are located by utilizing GeoLite Country database, a free IP geo-location database created by MaxMind [12].

- For the peer list modification module, `Libipq` is also utilized to modify the packets before they arrive at their destination, i.e., the P2P application. In particular, the data payload of the packet containing the peer list will be modified for localizing.

- For the communication module, TCP/IP is impelemented as a communication protocol to share the local peer list table among routers.

## 7.4 Experimental results

Figure 7.3 shows the experimental network setting. For the internet connection, we subscribed to *FLET'S HIKARI NEXT*, a 100 Mbps optical access service on the next generation network (NGN), and *plala HIKARI Mate* with *FLET'S* as an ISP in Japan. Two gateway routers were setup; each router was connected with a measurement host. On each measurement host, PPStream was installed as a P2PTV application. An on-demand video channel was selected to view on the PPStream. For the statistical measurements `Wireshark` was installed on the measurement hosts. All the measurement were conducted in December 2013.

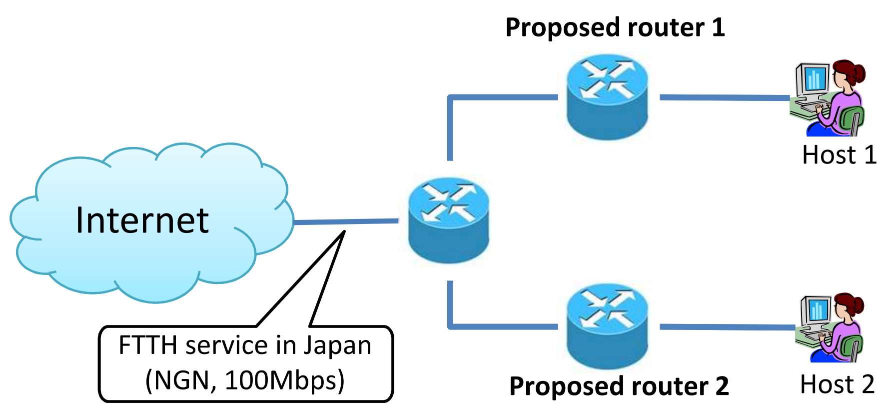


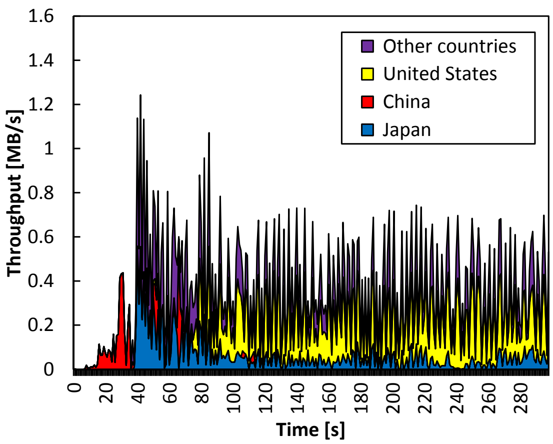

Figure 7.3: Network environment setting.

This chapter primarily focuses on verifying the effectiveness of the router collaboration scheme. For simplicity, I localize the traffic only at a country level. The experiments were conducted with three different scenarios as follows:

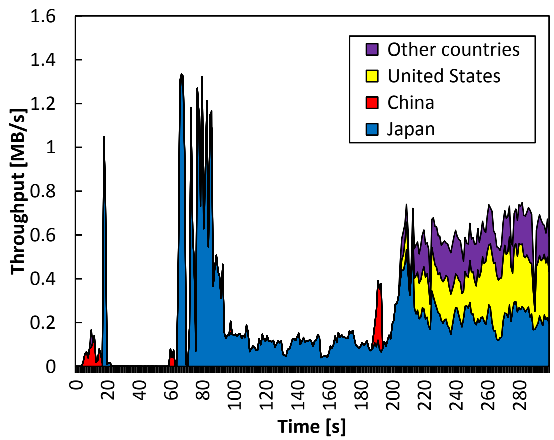
- Scenario 1: Router 1 does nothing to the peer list packets going through the router. Router 2 replaces all foreign peers in the peer list by Japanese peers that are collected at router 2. In this scenario, routers 1 and 2 do not collaborate with each other.
- Scenario 2: Router 1 does nothing to the peer list packets, but collects all Japanese peers in the peer lists and sends those peers to router 2. Router 2 replaces all foreign peers in the peer list by Japanese peers that are received

from router 1. In this scenario, router 1 helps router 2 to find Japanese peers for traffic localization.

- Scenario 3: Router 1 does nothing to the peer list packets, but collects all Japanese peers in the peer lists and sends those peers to router 2. Router 2 replaces all foreign peers in the peer list by Japanese peers. Half of the Japanese peers are collected at router 2, and the remaining peers are received from router 1. In this scenario, Japanese peers are collected at both routers.




(a) Keep original behavior of PPStream on host 1.

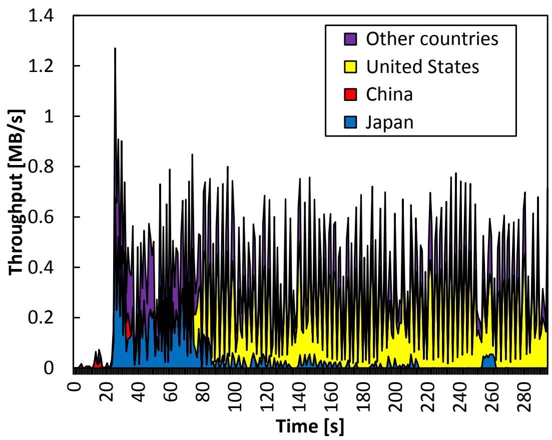



(b) Apply peer list modification method on host 2, replacing all foreign peers in the peer list by Japanese peers collected from host 2.

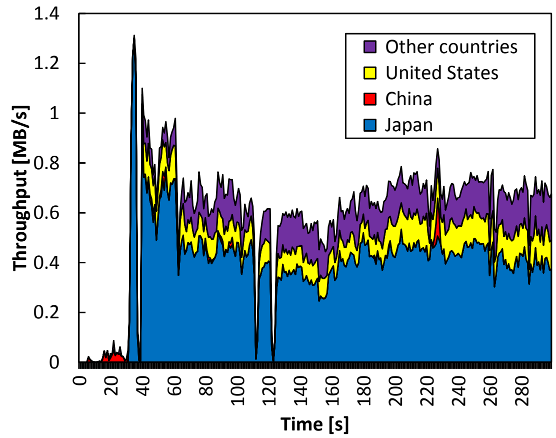
Figure 7.4: Scenario 1: Temporal change of throughput by regions measured on two hosts running PPStream simultaneously without applying the router collaboration scheme.

Table 7.1: Average downloaded data distribution by countries.

| Method | Japan | China | USA | Other countries |
|---|---|---|---|---|
| Original behavior of PPStream | 12% | 2% | 44% | 42% |
| Scenario 1 | 61% | 3% | 20% | 16% |
| Scenario 2 | 67% | 1% | 14% | 18% |
| Scenario 3 | 58% | 3% | 20% | 17% |



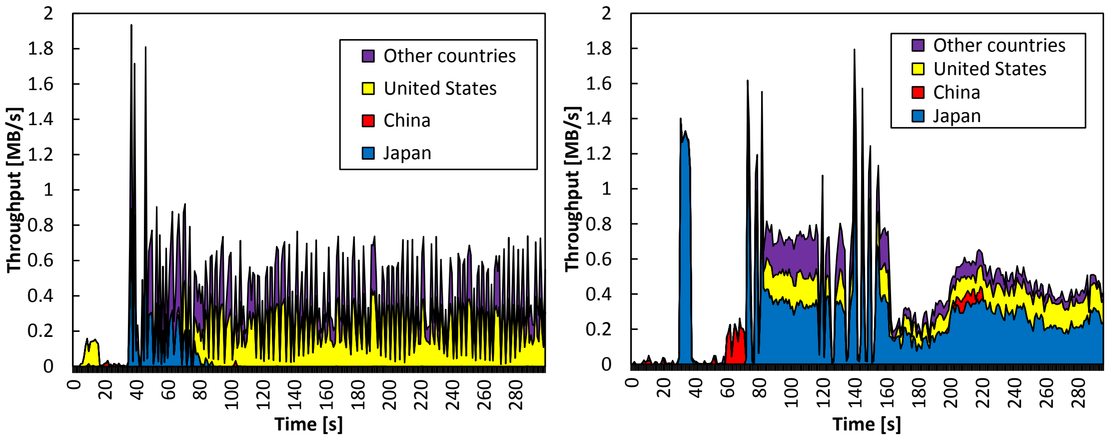

(a) Keep original behavior of PPStream on host 1.




(b) Apply peer list modification method on host 2, replacing all foreign peers in the peer list by Japanese peers received from host 1.

Figure 7.5: Scenario 2: Temporal change of throughput by regions measured on two hosts running PPStream simultaneously when applying the router collaboration scheme. The list of Japanese peers are collected only at host 1.

Figures from 7.4 to 7.6 present examples of temporal change of throughput by regions of three different scenarios. We can see that the original behavior of PPStream tends to download video data pieces from all over the world. In particular, as shown in Figs. 7.4 (a), 7.5 (a) and 7.6 (a), the traffic measured on host 1 comes from many countries including Japan, China, the United States, and other countries, which are bundled into one group. The traffic coming from Japan only accounts for minor portion, about 12% in average. In contrast, by replacing all foreign peers in the peer list by Japanese peers, the traffic coming from Japan measured on host 2 increases significantly. In particular, it accounts for 58%, 67% and 61% of the total traffic for scenarios 1, 2 and 3, respectively. Table 7.1 presents the average downloaded data ratio by different countries. It can be seen that there is no big difference in traffic distribution between scenario 1 and the others, i.e., between peer list modification scheme without router collaboration and peer list modification combining with the router collaboration. This is reasonable because two routers are placed very close together, in the same subnet. However, the scenario 2 has simulated the worst case assuming that router 2 could




(a) Keep original behavior of PPStream on host 1.

(b) Apply peer list modification method on host 2, replacing all foreign peers in the peer list by Japanese peers; half of Japanese peers are collected at host 2, and the remaining peers are received from host 1.

Figure 7.6: Scenario 3: Temporal change of throughput by regions measured on two hosts running PPStream simultaneously when applying the router collaboration scheme. The list of Japanese peers are collected at both hosts.

not find any Japanese peers. By using the list of Japanese peers recommended by router 1, the peer list modification scheme still work very well on router 2. Therefore, I strongly believe that the router collaboration will generally improve the performance of the peer list modification scheme on P2P traffic localization problem.

An interesting thing is that the overseas traffic measured on host 2 is not zero in all scenarios even though all the foreign peers have been already eliminated from the peer list. This proves that PPStream also contacts with some peers that do not appear in the plain-text peer list. However, PPStream still contacts primarily with Japanese peers that exist in the modified peer list. Compare to the results shown in chapter 4, we can found that the protocol of PPStream has been changed a little bit. Nevertheless, the increase of traffic inside Japan indicates that the router collaboration scheme combining with peer list modification scheme successfully realizes traffic localization on PPStream.

## 7.5 Conclusion

This work proposed a router collaboration scheme to combine with the previous work, peer list modification scheme, for improving the performance of P2P traffic localization problem. Routers collaborate with each other for sharing the list of local peers. Therefore, the peer list modification applied at each gateway router becomes much more effective because each router has more local peers in hand. An implementation of the proposed router is also introduced utilizing `iptables`, `libipq`, and GeoLite country database. The experiment on a P2P streaming application, named PPStream, with two PC-based routers indicated that the proposal successfully realizes traffic localization.

This implementation only localizes the traffic at a country level though, is extendable to localize the traffic at ISP and AS levels. In further study, I will improve the proposed method and try to test on a larger network scale, with the collaboration among many routers.

# Chapter 8

# Conclusion and Future Work

## 8.1 Conclusions

This dissertation discussed a router-aided approach for P2P traffic localization. Four methods have been proposed: peer list modification, video request packet redirection, degrading network performance of inter-domain traffic, and router collaboration methods.

In the degrading network performance of inter-domain connections method, I proposed to localize the traffic hierarchically with multiple levels such as AS level, ISP level, and country level. To achieve the hierarchy of localization, I introduced the logical distance between two peers based on not only the physical distance but also the number of peers that exist in the same area as the querying peer. To deploy the idea of degrading network performance of inter-domain traffic, three different schemes were introduced including HDIS, HPLS, and HBLS.

The router architectures and the implementations of all the methods were also described in detail in this dissertation. For the implementations, `libipq`, `libpcap`, `GeoLite` database, and `dummynet` were utilized on a PC-based router operated under Linux OS.

In summary, this study provides the following contributions:

- Compared to random and/or RTT-based peer-selection strategies, the proposed methods significantly reduce cross-ISP/AS/country traffic.

- Compared with existing locality-enhancing approaches focusing on the application layer, the proposed method requires neither dedicated servers, nor collaboration between ISPs and P2P users, nor any modification of P2P application software. Therefore, the proposed methods can be easily extended to all types of P2P applications and also easily introduced to the current networks.

- This work also provides a hierarchical localization method to balance the tradeoff between traffic localization and service quality of P2P applications. In particular, the proposed hierarchical localization method significantly reduces the cross-domain traffic while maintaining the quality performance of the P2P applications.

- The feature of HDIS, HPLS, and HBLS that force a P2P streaming application to download the data from a peer in the same LAN is very significant. To the best of my knowledge, no other research shows the same result as this study.

## 8.2 Future works

This section briefly outlines the directions of future work.

For the peer list modification method and video request packet redirection method, further study will improve these methods using more information of the peers such as ISP, AS, and quality of connection links to achieve finer-grained results. In addition, the investigation of the degradation of video quality when applying the methods should be also carefully taken into account.

For HDIS, HPLS, and HBLS, considering only the waiting time is not enough for measuring the performance of the applications. In the future, many other metrics should be taken into account such as the amount of downloaded traffic in a certain time (size of cached file by the P2P application within a certain time), the number of disruptions within a certain time, quality of the video, etc.

Furthermore, it is also necessary to consider the QoE when using the proposed routers.

In the future, I am going to evaluate HDIS, HPLS, and HBLS on a large-scale network where different scenarios could be shown and could be used in addition to the current experiments. For instance, due to the delay insertion, packet loss, or bandwidth limitation the application performance would probably be degraded much if many peers are in the same AS, but none has the desired content of the querying peer. Therefore, the proposed logical distance should depend not only on the number of peers in the same AS, ISP, or country but also the number of active peers in the same area (the peers that have the desired content).

Besides, to make the proposed routers more friendly to the ISPs, the proposed logical distance between two peers should be computed based on not only the physical distance of peers but also the relationship among AS/ISP. For example, the AS relationships are classified into four categories: provider-to-customer, customer-to-provider, peer-to-peer, and sibling-to-sibling. This idea can be achieved because we can easily look up the AS/ISP relationship using some databases provided by CAIDA [15].

Finally, since the performance of traffic localization fundamentally depends on the accuracy of P2P traffic classification, study on traffic classification is essentially required to complete the router-aided approach in the future.

# References


[1] Cisco System, Inc. “Cisco visual networking index: forecast and methodology, 2011-2016,” White paper, May 2012.

[2] PPTV. [Online]. Available: http://www.pptv.com/.

[3] PPStream. [Online]. Available: http://www.pps.tv/.

[4] SopCast. [Online]. Available: http://www.sopcast.com/.

[5] Zattoo. [Online]. Available: http://zattoo.com/.

[6] PeerCast. [Online]. Available: http://peercast.sourceforge.net/.

[7] Conviva. [Online]. Available: http://www.conviva.com/.

[8] TVants. [Online]. Available: http://tvants.en.softonic.com/.

[9] Sandvine, Inc. “Global Internet Phenomena Report: Spring 2011”, Technical Report, 2011.

[10] Wireshark. [Online]. Available: http://www.wireshark.org/.

[11] Linux man page. [Online]. Available: http://linux.die.net/man/3/libipq.

[12] MaxMind and GeoIP, IP address location technology. [Online]. Available:http://www.maxmind.com/app/ip-location/.

[13] Tcpdump and libpcap pulbic repository. [Online]. Available: http://tcpdump.org/.

[14] Dummynet. [Online]. Available: http://info.iet.unipi.it/~luigi/dummynet/.

[15] The Cooperative Association for Internet Data Analysis (CAIDA). [Online]. Available: http://www.caida.org/home/.

[16] V. Aggarwal, A. Feldmann, and C. Scheideler. Can isps and p2p users cooperate for improved performance? *ACM SIGCOMM Comput. Commun. Rev.*, **37**(3):29–40, July 2007.

[17] S. Ali, A. Mathur, and H. Zhang. Measurement of commercial peer-to-peer live video streaming. In *ICST Workshop on Recent Advances in Peer-to-Peer Streaming*, 2006.

[18] R. Alimi, R. Penno, and Y. Yang. Alto protocol. In *Internet draft, draft-ietf-alto-protocol-10.txt*, Oct. 2011.

[19] I. Bermudez, M. Mellia, and M. Meo. Passive characteriza-tion of sopcast usage in residential isps. In *IEEE Int. Conf. Peer-to-Peer Comput. (P2P 2011)*, pages 1–9, Sept. 2011.

[20] R. Bindal, P. Cao, W. Chan, J. Medved, G. Suwala, T. Bates, and A. Zhang. Improving traffic locality in bittorrent via biased neighbor selection. In *IEEE Int. Conf. Distributed Comput. Syst. (ICDCS2006)*, July 2006.

[21] R. Birke, E. Leonardi, M. Mellia, A. Bakay, T. Szemethy, C. Kiraly, R. Lo Cigno, F. Mathieu, L. Muscariello, S. Niccolini, J. Seedorf, and G. Tropea. Architecture of a network-aware p2p-tv application: the napa-wine approach. *IEEE Com- mun. Mag.*, **49**(6):154–163, June 2011.

[22] D.R. Choffnes and F.E. Bustamante. Taming the torrent - a practical approach to reducin cross-isp traffic in peer-to-peer systems. In *ACM SIGCOMM2008*, pages 363–374, Aug. 2008.

[23] 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 Trans. Multimedia*, **12**(1):54–63, Jan 2010.

[24] M. Costa, M. Castro, R. Rowstron, and P. Key. Pic: practical internet coordinates for distance estimation. In *24th Int. Conf. Distributed Comput. Syst. (ICDCS'04)*, pages 178–187, March 2004.

[25] R. Dunaytsev, D. Moltchanov, Y. Koucheryavy, O. Strandberg, and H. Flinck. A survey of p2p traffic management approaches: best practices and future directions. *Internet Engineering*, **5**(1):318–330, June 2012.

[26] P. Francis. A global internet host distance estimation service. *IEEE/ACM Trans. Net.*, **9**(5):525–540, Oct. 2001.

[27] X. Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross. Insights into pplive: A mea-surement study of a large-scale p2p iptv system. In *IPTV Workshop, Inter-national World Wide Web Conference*, 2006.

[28] X. Hei, C. Liang, J. Liang, Y. Liu, and K.W. Ross. A measurement study of a large-scale p2p iptv system. *IEEE Transactions on Multimedia*, **9**(8):1672–1687, Dec. 2007.

[29] A. Horvath, M. Telek, D. Rossi, P. Veglia, D. Ciullo, M.A. Garcia, E. Leonardi, and M. Mellia. Dissecting pplive, sopcast, tvants. Technical report, NAPA-WINE project, 2009.

[30] X. Jin and Y.-K. Kwok. Network aware p2p multimedia streaming: capacity or locality? In *IEEE Int. Conf. Peer-to-Peer Comput. (P2P 2011)*, pages 54–63, Sept. 2011.

[31] T. Karagiannis, P. Rodriguez, and K. Papagiannaki. Should internet service providers fear peer-assisted content distribution? In *Internet Management Conf. (IMC 2005)*, pages 63–76, Oct. 2005.

[32] H.-Y. Lee and A. Nakao. A feasibility study of p2p traffic localization through network delay insertion. *IEICE Trans. Commun.*, **E95-B**(11):3464–3471, Nov. 2012.

[33] H.-Y. Lee and A. Nakao. Isp-driven delay insertion for p2p traffic localization. *IEICE Trans. Commun.*, **E96-B**(1):40–47, Jan. 2013.

[34] Bo Li, Susu Xie, Yang Qu, Keung G.Y., Chuang Lin, Jiangchuan Liu, and Xinyan Zhang. Inside the new coolstreaming: Principles, measurements and performance implications. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, April 2008.

[35] B. Liu, Y. Cao, Y. Cui, Y. Lu, and Y. Xue. Locality analysis of bittorrent-like peer-to-peer systems. In *IEEE Consumer Commun. and Networking Conf. (CCNC 2010)*, pages 1–5, Jan. 2010.

[36] B. Liu, Y. Cui, Y. Lu, and Y. Xue. Locality-awareness in bittorrent-like p2p applications. *EEE Trans. Multimedia*, **11**(3):361–371, April 2009.

[37] T. Miyoshi, Y. Shinozaki, and O. Fourmaux. A p2p traffic localization method with additional delay insertion. In *4th Int. Conf. Intelligent Networking and Collaborative Syst. (INCoS2012)*, pages 148–154, Sept. 2012.

[38] K. Mochalski and H. Schulze. "Deep Packet Inspection: Technology, Applications & Net Neutrality," ipoque GmbH, White Paper, 2009.

[39] M. Mushtaq, D. Meddour, and A. Toufik. Open issues in p2p multimedia streaming. In *IEEE ICC 2006 Workshop on Multimedia Communications Workshop (MultiCom)*, June.

[40] T.S.E. Ng and H. Zhang. Predicting internet network distance with coordinates-based approaches. In *IEEE INFOCOM2002*, June 2002.

[41] L. Plissonneau, J. Costeux, and P.Brown. Detailed analysis of edonkey transfers on adsl. In *2nd Conf. Next Generation Internet Design and Engineering (NGI '06)*, pages 256–262, April 2006.

[42] J. Seedorf, S. Kiesel, and M. Stiemerling. Traffic localization for p2p-applications: the alto apprach. In *IEEE Int. Conf. Peer-to-Peer Comput. (P2P2009)*, pages 171–177, Sept. 2009.

[43] A. Sentinelli, M. Gerla G. Marfia, L. Kleinrock, and S. T ewari. Will iptv ride the peer-to-peer stream? *IEEE Com- mun. Mag.*, **45**(6):86–92, June 2007.

[44] L. Sheng, X. Dong, J. Song, and K. Xie. Traffic locality in the emule system. In *Int. Conf. Networking and Distributed Comput. (ICNDC 2010)*, pages 387–391, Oct. 2010.

[45] T. Silverston and O. Fourmaux. Measuring p2p iptv systems. In *NOSSDAV07, International Workshop on Network and Operating Systems Support for Digital Audio and Video*, 2007.

[46] X. Su and L. Chang. A measurement study of ppstream. In *3rd Int. Conf. Commun. and Networking in China (ChinaCom 2008)*, pages 1162–1166, Aug. 2008.

[47] T. Telkamp. Traffic characteristic and network planing. In *NANOG 26 meeting, Eugene, OR*, Oct. 2002.

[48] S. Valenti and D. Rossi. Fine-grained behavioral classification in the core: the issue of flow sampling. In *7th Int. Wireless Commun. and Mobile Comput. Conf. (IWCMC2011)*, pages 1028–1032, July 2011.

[49] E. Veloso, V. Almeida, W. Meira, A. Bestavros, and S. Jin. A hierarchical characterization of a live streaming media workload. In *ACM SIGCOMM IMW*, 2002.

[50] F. Wang, J. Liu, and Y. Xiong. Stable peers: Existence, importance, and application in peer-to-peer live video streaming. In *IEEE Infocom'08, Phoenix, AZ*, Apr.

[51] B. Wong, A. Slivkins, and E. G. Sirer. Meridian: a lightweight network location service without virtual coordinates. In *ACM SIGCOMM2005*, pages 85–96, Aug. 2005.

[52] H. Xie, Y.R. Yang, A. Krishnamurthy, Y. Liu, and A. Silberschatz. P4p: provider portal for applications. In *ACM SIGCOMM 2008*, pages 351–362, Aug. 2008.

# List of Publications

[P.1] H. Hoang-Van, T. Miyoshi, and O. Fourmaux, "A Router-aided P2P Traffic Localization Method with Bandwidth Limitation," *VNU Journal of Science: Computer Science and Communication Engineering (VNU JCSCE)*, Vol. 30, No. 3. pp. 50-63, August 2014.

[P.2] H. Hoang-Van, Y. Shinozaki, T. Miyoshi, and O. Fourmaux, "A Router-aided Hierarchical P2P Traffic Localization Based on Variable Additional Delay Insertion," *IEICE Transactions on Communications*, Vol. E97-B, No. 1 , pp.29-39, January 2014.

[P.3] H. Hoang-Van, K. Mizutani, T. Miyoshi, and O. Fourmaux, "P2P Traffic Localization by Forcing Packet Loss," *International Journal of Networked and Distributed Computing (IJNDC)*, Atlantis press, Vol. 1, No. 4 , pp. 251-259, November 2013.

[P.4] H. Pham-Thi, H. Hoang-Van, T. Miyoshi, and T. Yamazaki. "QoE-driven Bandwidth Allocation Method Based on User Characteristics," to appear in *16th Asia-Pacific Network Operations and Management Symposium (APNOMS 2014)*, Taiwan.

[P.5] H. Hoang-Van, T. Miyoshi, and O. Fourmaux, "Peer list sharing by router collaboration for P2PTV traffic localization," *The Fifth International Conference on Communications and Electronics (ICCE 2014)*, Danang, Vietnam, pp. 623-628, August 2014.

[P.6] H. Hoang-Van, T. Miyoshi, and O. Fourmaux, "A Hierarchical P2P Traffic Localization Method with Bandwidth Limitation," *2014 IEEE International Conference on Communications (ICC2014)*, pp. 3142-3147, Australia, June 2014.

[P.7] H. Hoang-Van, T. Miyoshi, and O. Fourmaux, "A Hierarchical P2P Traffic Localization Method with Bandwidth Limitation," *10th IEEE RIVF International Conference on Computing and Communication Technologies (RIVF2013)*, Hanoi, Vietnam, pp.127-130, November 2013.

[P.8] K. Mizutani, Y. Shinozaki, H. Hoang-Van, and T. Miyoshi, "An Adaptive Packet Discard Method for P2P Traffic Localization," *15th Asia-Pacific Network Operations and Management Symposium (APNOMS2013)*, Hiroshima, Japan, Paper No. I2-2, September 2013.

[P.9] H. Hoang-Van, T. Miyoshi, and O. Fourmaux, "P2PTV Traffic Localization by Deep Packet Inspection," *14th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel / Distributed Computing (SNPD2013)*, Honolulu , Hawaii, USA, pp. 375-380, July 2013.

[P.10] H. Hoang-Van, K. Mizutani, T. Miyoshi, and O. Fourmaux, "P2P Traffic Localization by Forcing Packet Loss," *12th IEEE/ACIS International Conference on Computer and Information Science (ICIS2013)*, Niigata, Japan, pp. 323-328, June 2013.

[P.11] H. Hoang-Van, T. Miyoshi, and O. Fourmaux, "P2P Traffic Localization by Peer List Modification at Network Layer," *7th South East Asian Technical University Consortium (SEATUC) Symposium*, Bandung, Indonesia, March 2013.

[P.12] H. Hoang-Van, K. Mizutani, T. Miyoshi, and O. Fourmaux, "On Balancing the trade-o between localization and performance on P2P applications," 2014 IEICE General Conference, English Session, BS-6-13, September 2014.

[P.13] H. Hoang-Van, T. Miyoshi, and O. Fourmaux, “A Router Collaboration Scheme for P2PTV Traffic Localization,” to appear in *2014 IEICE General Conference*, English Session, BS-1-69, pp. S-129 - S-130, March 2014.

[P.14] H. Hoang-Van, T. Miyoshi, and O. Fourmaux, “A Router-aided Method for Hierarchical P2P Traffic Localization with Variable Additional Delay Insertion,” *2013 Communications Society Conference of IEICE*, English Session, BS-7 -25, pp. S-81 - S-82, September 2013.

[P.15] H. Hoang-Van, T. Miyoshi, and O. Fourmaux, “A Peer List Modification Method for P2PTV Traffic Localization,” *2013 IEICE General Conference*, BS-1-18, pp. S-35 - S-36, March 2013.

[P.16] H. Pham-Thi, H.-T. Nguyen, H. Hoang-Van, and T. Miyoshi, “The Relationship between Quality of Experience and Chunk Loss Ratio on P2PTV,” *IEICE 10th QoS Workshop*, QW10-P-8, November 2012.

[P.17] H. Pham-Thi, H.-T. Nguyen, H. Hoang-Van, and T. Miyoshi, “On the Relationship between Chunk Loss Ratio and Quality of Experience on P2PTV,” *2013 IEICE General Conference*, BS-1-20, pp. S-39 - S-40, March 2013.