國立交通大學

資訊科學與工程研究所

碩 士 論 文

一個基於顯示群播之改良式任播及任意多播服務

An Enhanced Anycast and Manycast Services based on Explicit Multicast

研 究 生：范書豪

指導教授：陳耀宗　教授

中 華 民 國 101 年 11 月

一個基於明確群播之改良式任播及任意多播服務
An Enhanced Anycast and Manycast Service based on Explicit Multicast

研 究 生：范書豪 Student：Shu-Hao Van

指導教授：陳耀宗 Advisor：Yaw-Chung Chen

國 立 交 通 大 學
資 訊 科 學 與 工 程 研 究 所
碩 士 論 文

A Thesis

Submitted to Institute of Computer Science and Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

In

Computer Science

November 2012

Hsinchu, Taiwan, Republic of China

中華民國 101 年 11 月

一個基於明確群播之改良式任播及任意多播服務

學生：范書豪　　　　　　　　　　　　指導教授：陳耀宗 博士

國立交通大學資訊科學與工程研究所

摘要

因為網路技術的日新月異，取得服務的使用者越來越多，要如何在搜尋分散式服務主機時減少整體網路流量並且縮短回應時間，是一個重要議題。選播是在 IPv6 中提出的一種新型網路服務，像單播和群播一樣是網路通訊服務。一個任播封包會在一群擁有相同選播位址的節點去傳遞封包到一個適合的節點。

在本篇論文中，我們一開始介紹 IPv6 下的任播服務。再者我們會提出幾個問題與提供解決方式，我們提出的架構有下列特色:(1)實作任播和任意多播通信服務。(2)減少執行任播和任意多播服務的整體流量。(3)較快找到目標主機或服務。(4) 較精確地找到的目標主機或服務。我們會去比對一般傳統方式和討論可適當減少的負擔。我們提出新的任播和任意多播搭配明確群播技術運作在同儕網路中的網路電視上。

An Enhanced Anycast and Manycast Service based on Explicit Multicast

Student：Shu-Hao Van Advisor：Dr. Yaw-Chung Chen

Institute of Computer Science and Engineering

National Chiao Tung University

Abstract

The population of the Internet has been expanded quickly every year, so does the diversification of Internet services for subscribers. How to search the service host and reduce the network traffic load as well as shorten the response time is an important issue. Anycast is a new IPv6 feature that supports service–oriented address assignments in IPv6 networks, like unicast and multicast IP services. An anycast packet is transmitted to one appropriate node selected out of all the nodes assigned to the same anycast address.

In this thesis, we first review IPv6-based anycast communications, then raise several issues and provide possible solutions to these problems. Our proposed architecture features the following characteristics: It (1) implemented anycast and manycast communications, (2) improved overall network traffic on anycast and manycast protocol, (3) reduced host response time, (4) precisely found host or service. We also investigate and compare the traditional methods and discuss the appropriate techniques for reducing overheads as much as possible. Consequently, we propose new anycast and manycast based on Xcast mechanism, we then applies the proposed method to p2p network IPTV and evaluate their performance improvement.

Key Words

Peer-to-peer computing, peer-to-peer live streaming, IPTV, Anycast, Manycast

誌　謝

在這漫長的求學生涯，一路上很感謝陳耀宗老師在這論文上的指導與鼓勵，每次向請教老師問題時，老師不厭其煩的教導著我，讓我受益良多，此外老師做人處事方面也是我該學習的典範，不時叮嚀著我要趕快把論文做好，尤其是待人親切讓求學在外的我，有股像慈母嚴父一般溫暖的感覺，再次謝謝陳耀宗教授如此對我的用心與關懷。在這也謝謝鍾老師在我過去 2 年的教誨，而鍾老師做事的嚴謹我永遠會記在心中。

接著感謝實驗室的所有學長和同學和學弟妹，尤其在這邊要感謝碩班同學書賢，在論文上的幫助和意見，不僅僅在研究上遇到瓶頸以及論文撰寫的技巧方面，並且在八月的時候和你在實驗室在圖書館一起奮鬥的回憶永遠都不會忘記。感謝碩班同學譽維、俊諺、冠宇學長、阿華學長、小八學姐，一起分享研究心得，一起聊天，以及 Lab 小精靈巧妹的加油打氣。

還有感謝身邊的朋友支持以及願意聽我煩惱的一些室友，致成、國成、漢忠、政緯、騏嘉、鈞隆。

最後，很感謝我的爸媽與書瑋和舒婷，不論遇到什麼挫折他們都會在背後永遠都支持著我，要不是他們我沒辦法無牽掛的專心在論文上。尤其當我每次回新竹時，母親總是在我的行囊中裝了好多吃的東西怕我一人在外餓著，父親總是對我無限包容不僅在每一階段求學遇到困難和受創，很感謝他們的教養與支持讓我能順利的完成這最後一個階段的求學。

Contents

Table List

Figure List

Chapter 1 Introduction

Recently, use of the Internet typified by the WWW (World Wide Web) and e-mail and FTP service have spread extensively. Anycast is a network interface and routing methodology in which datagrams from a single sender are routed to the topologically nearest node in a group of potential member all identified by the same destination address. Anycast, which is defined in the IPv6, is a new networking paradigm supporting service–oriented addresses and an identical address can be assigned to multiple nodes providing a specific service.

1.1 Communication Types

The scale of the Internet has been expanded every year with the steady increasing number of Internet users, as well as diversification of Internet services. The Internet has grown tremendously over the last two decades. The current Internet mainly uses the IPv4 (Internet Protocol version 4) [1] as the major communication protocol. However, IPv4 address cannot function effectively enough to cope with this enormous increasing users and demand for the Internet. For example, all new machines that are being added to the Internet are running short of the assigned IPv4 addresses. With the trigger of the IPv4 address space problem, IPv6 (Internet Protocol version 6) [2] was designed as the next-generation Internet protocol. IPv6 has a wide address space (2^128) that we do not have to worry about the amount of remaining address space even if a unique address were assigned to all Internet users, and that number of problems (e.g., security issues and mobility issues) are raised by IPv4 in the IPv6 specifications. Furthermore, IPv6 includes some new functions that are currently in demand or are predicted to be useful in the future.

Information exchange can broadly be classified as unicast (one-to-one), broadcast (one-to-all) and multicast [3] (one-to-many). One of the biggest advantages of multicasting is the conservation of bandwidth. The multicast server sends out only one packet, then the router generates multiple packets to reach each of the receivers. In this manner the network resources are used efficiently. In unicast routing, the server sends out a packet to each of the receivers. A more recent variation of multicast is anycast. It is a one-to-many distribution. There may be multiple recipients of an anycast message, but the sender sends the message only to the node that is logically or topologically the closest to it. Table 1.1 below is a comparison of unicast, multicast and anycast.

Table 1.1 Comparison for unicast, multicast and anycast.

	Unicast	**Multicast**	**Anycast**
Communication form	Point to point	Point to multipoint	Point to point
Target of address	node	group	service type
Number of membership	single	multiple	multiple
Roles in C/S model	both	client	server

Anycast [4] is the point-to-point flow of packets between a single client and the "nearest" destination server identified by an anycast address [5]. The idea behind anycast is that a client could send packets to any one of several possible servers which provide a particular service or application but does not really care which one. Any number of servers can be assigned a single anycast address within an anycast group. A client sends packets to an anycast server by placing the anycast address in the packet header. Routers then attempt to deliver the packet to a server with the matching anycast address.

1.2 Multicast and Explicit Multicast Technology

Multicast is the term used to describe communication where a piece of information is sent from one or more points to a set of other points. In this case there may be one or more senders, and the information is distributed to a set of receivers. One example of an application which may use multicast is a video server sending out networked TV channels. Simultaneous delivery of high quality video to each of a large number of delivery platforms will exhaust the capability of even a high bandwidth network with a powerful video clip server. This poses a major scalability issue for applications which required sustained high bandwidth. One way to significantly ease scaling to larger groups of clients is to employ multicast services.

In the Host Group Model the packet carries a multicast address as a logical identifier of all group members. In Explicit Multicast (Xcast) [6], the source node keeps track of the destinations in the multicast channel that it wants to send packets to destinations. The source encodes the list of destinations in the Xcast header, and then sends the packet to a router. Each router along the way parses the header, partitions the destinations based on each destination‘s next hop, and forwards a packet with an appropriate Xcast header to each of the next hops. When there is only one destination left, the Xcast packet can be converted into a normal unicast packet, which can be unicasted along the remainder of the route. This is called X2U (Xcast to Unicast).

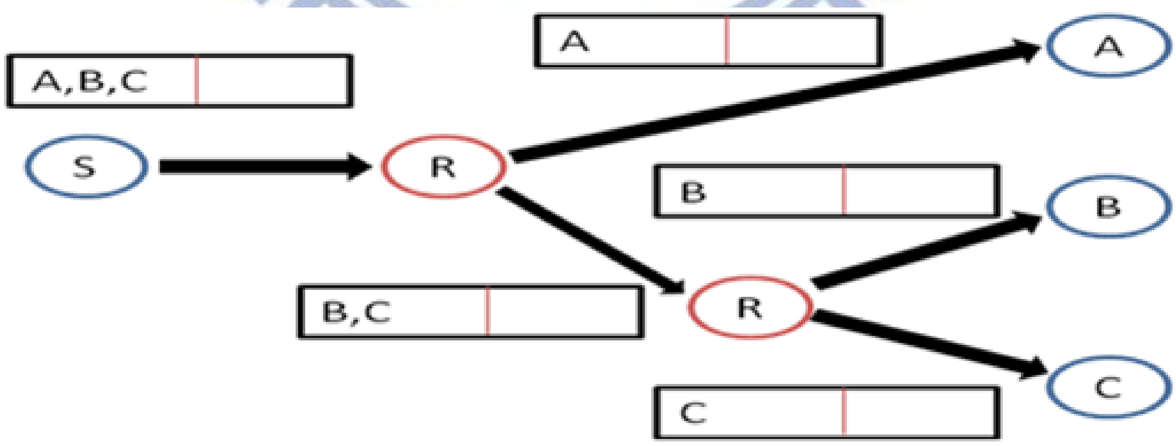


Figure 1.1 Explicit Multicast.

Figure 1.1 shows the source send the data to group members (A,B,C) that have the same packet content. The source encodes the list of destination (A,B,C) in packets header and then sends the packets to a router. The router will decode destination of the packet header that parses the next router. In this case, the destination A and B, C aren't in the same routing that up one level router will duplicate two packets which are sent to appropriate routers.

1.3 Anycast Technology

In recent years, anycast has become increasingly popular for adding redundancy to DNS servers to complement the redundancy that the DNS architecture itself already provides. A packet sent to an anycast address is delivered to one of the interfaces identified by that address (the "nearest" one, according to the routing protocol's measure of distance). An example can be found in [7]. Anycast is the use of routing and addressing policies to affect the most efficient path between a single source and several geographically dispersed targets that "listen" to a service within a receiver group. In Anycast, the same IP address space is used to address each of the listening targets (DNS servers in our case). Layer 3 routing dynamically handles the calculation and transmission of packets from our source (DNS Client) to its most appropriate (DNS Server) target. Configuration methodology that provides redundancy and load sharing to specific types of network services on the Internet. Figure 1.2 shows an example of anycast service. There are five servers associated with the anycast address A_{any} and B_{any}. When two source nodes send a packet, A_{any} and B_{any} will be the destination address, and the packet will be sent to one of five nodes, not to all servers.

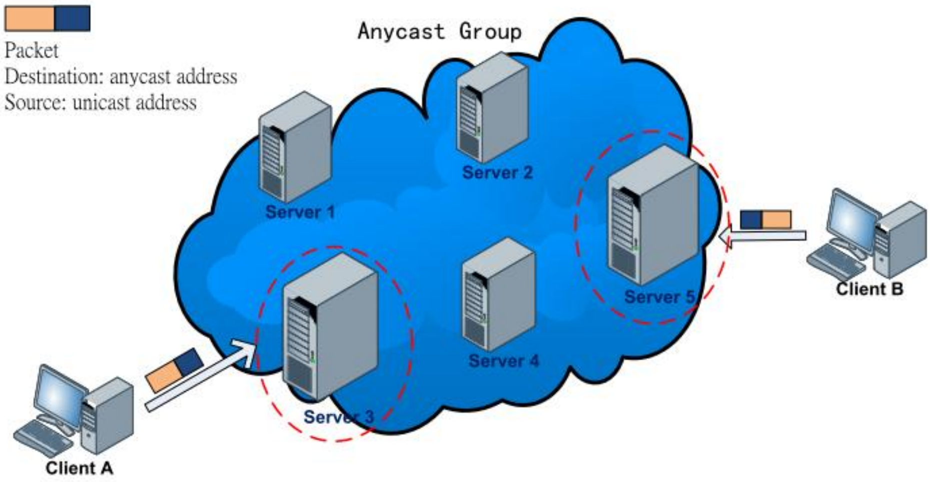


Figure 1.2 Examples of using anycast in Internet.

One of the important features of anycast is that the appropriate responder for anycast packet may change in accordance with the network conditions, location of the anycast initiator, or the availability of anycast responders—even if the same anycast initiator sends multiple anycast packets.

Anycast has some unique and interesting characteristics. The following are some examples of anycast applications.

A. Service discovery

An anycast address enables us to assign a service-oriented address. To achieve a service discovery, we first assign an anycast address to each service, and then give the anycast address to the nodes on which the associated service is running. As a result, we can find the (appropriate) node providing the service simply by specifying the correspondent anycast address. For example, when we assign a well-known anycast address to a DNS server, the DNS query packets destined for the anycast address would be forwarded to an appropriate DNS server. Therefore, we can receive a response from an appropriate DNS server regardless of connected locations.

B. Location-dependent service

It is possible for us to connect to the nearest server by using anycast. That is, we can connect to a location-specific server. For example, we can get the local time via using the same anycast address even if we move between countries. It is similar to the “Emergency call” in the real world.

C. Load-balancing

As the numbers of anycast responders increases the same anycast address, anycast initiators can communicate with the appropriate anycast responders in each case. If anycast responders are globally distributed, the loads of anycast responders achieve balance.

D. Robustness against a breakdown

When an anycast responder fails, another responder with the same anycast address can receive the anycast packet. Therefore, the service for anycast initiators can be provided continuously even after the failure of an anycast responder.

1.4 Manycast Technology

Manycast was firstly proposed in [8]. In the literature, it defined that manycast is a group communication paradigm in which one client communicates simultaneously with k of m equivalent servers in a group. It provides a bidirectional channel for requesting/replying communication between client and servers, not merely one-way dissemination of data. The activity of manycast transaction is shown in Figure. 1.3. Essentially, manycast is a group communication paradigm in which one client communicates simultaneously with some threshold number k of servers from the m members of a group.

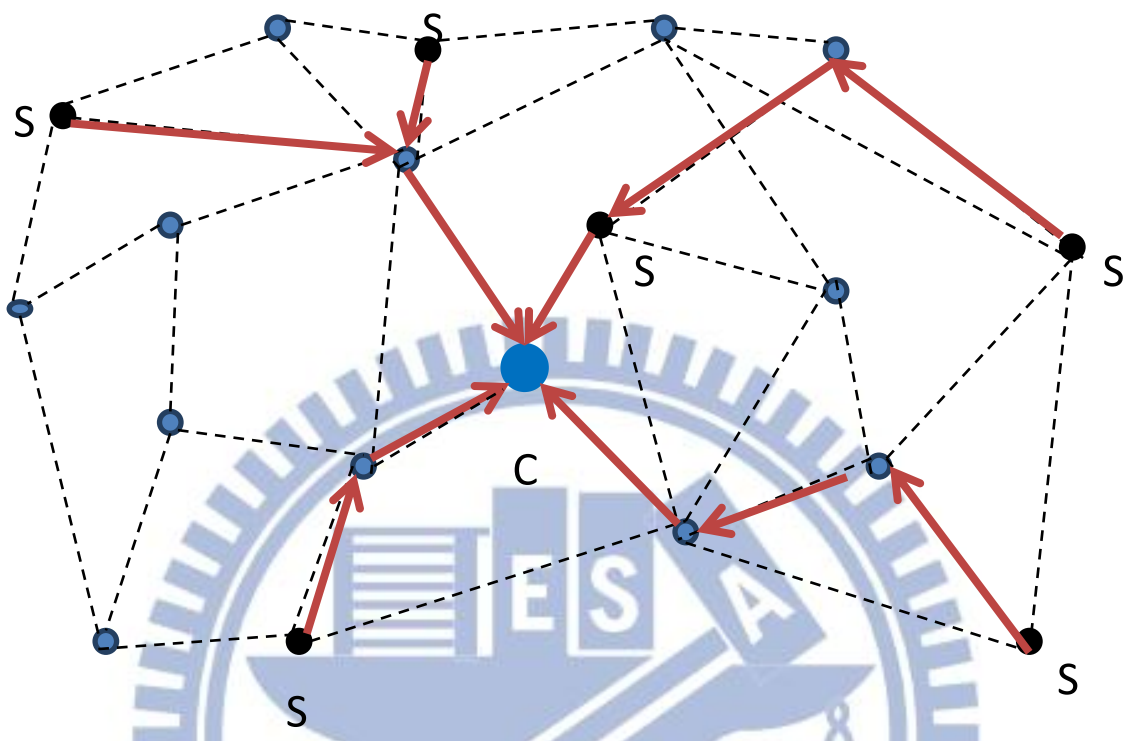


Figure 1.3 Service Request Phase of Manycast.

Other services, localized in nature, require a degree of robustness against failure. They often perform service discovery through a local broadcast-based protocol, such as the Network Time Protocol [9].

In the NTP scheme, a client wishes to locate the three best/nearest servers with which to synchronize its clock. All servers are members of a well-known IP multicast group. Clients locate servers by performing an expanding ring search over the IP multicast tree. Once the server set has been located, clients use a very long-periodic refresh to determine if better/nearer servers have appeared in the network.

1.5 Motivation and Purpose

One important issue is to correctly and quickly find out the location of service providers which is generally called servers. Therefore, the sender host can choose one of many functionally identical hosts. To the best of our knowledge, there are no schemes about how anycast or manycast can be applied in the Internet to accelerate finding the target host. Therefore, we propose a scheme to speed up finding the target host based on anycast and manycast based Xcast. This scheme could expand on different systems, such as P2P IPTV service.

When it comes to P2P IPTV streaming service [10], the first thing for the newly join peer must find out one appropriate tracker among some other ones and obtain the peer list from that selected tracker. In addition, after getting the peer list, the new peer sends queries to those that are already on the list. Moreover, BitTorrent is an application for file sharing, how to get the fast response from partners for user and how to reduce the number of messages from sender side is also to be concerned.

In this thesis, we propose two mechanisms that are anycast by Xcast and manycast by Xcast. Both of them resolve the performance of traditional protocols such as Round Robin, Anycast name resolver and sequential unicast. As a result, load distribution among the hosts that have the same anycast addresses can be achieved if we utilize some appropriate anycast approaches whose requests are evenly distributed to the destination hosts.

In this thesis, we aim to realize that (1) implement the two mechanisms anycast and manycast, both are based on Xcast (2) Quickly find out the service provider (3) Reduce the network traffic on the Internet. (4) Apply Xcast-based anycast and Xcast-based manycast to speed up the search for appropriate tracker and partners in P2P IPTV service.

The rest of this thesis is organized as follows. Chapter 2 presents the background of existing anycast technologies. Chapter 3 describes the schemes we propose in detail. The simulation and numerical results are presented in Chapter 4. Finally, the conclusions are addressed in Chapter 5.

Chapter 2 Background

For a smooth discussion in this thesis, we will describe details of anycast in this section. Firstly, we will introduce some terminologies which are essential in describing our proposed mechanism. Next, we will classify some types of anycast.

2.1 Application Scenarios

In previous work, it showed some suitable applications for anycasting [11]. If there are several hosts that provide the same service in the network, anycast can be used to help clients to find the nearest service. Most of recent implementations of Internet focus on the routing protocols in the network layer. When anycast is realized in the network layer, the anycast functions can be added to existing applications without editing the source codes. Another type is the application layer anycast, where the application implements the anycast mechanism. In this case, we should edit or make new source codes for the applications. This subsection reviews what kinds of applications are suitable to anycast communications. One important example is server location [12], through which the sender host can choose one of many functionally identical hosts.

As a result, load distribution among anycast hosts can be achieved if we utilize some appropriate anycast routing method, where anycast requests are evenly distributed to hosts. Figure 2.1 shows the client communicated with same anycast address of the group.

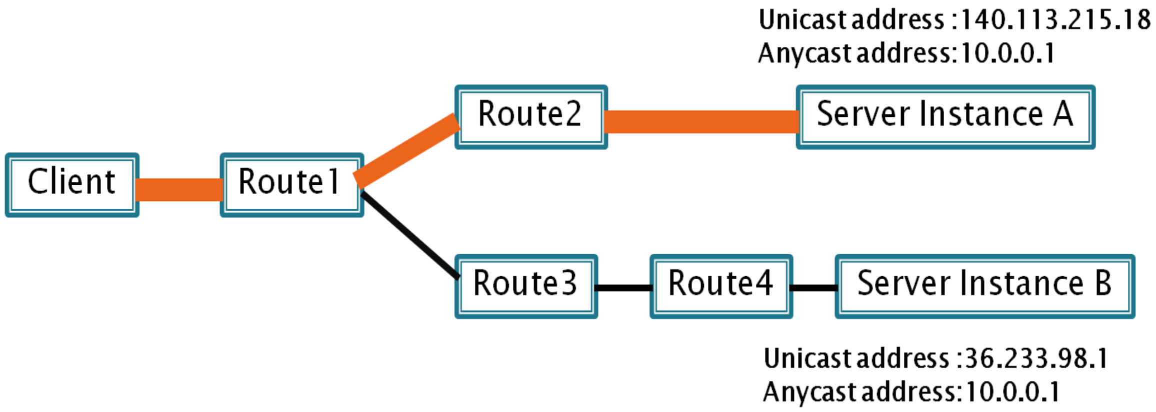


Figure 2.1 Anycast communication (Router).

While this kind of service can be obtained through application–layer anycast, the node can communicate automatically with an appropriate (e.g., nearest) server through network–layer anycasting. Moreover, the additional anycast examples listed below clearly demonstrate this scenario.

Anycast in P2P IPTV System

The gate to overlay network is an example of an anycast application. A P2P (Peer-to-Peer) service server constructs a logical network topology among nodes participating in the service. However, the peer needs to know the address to connect the logical network prior to the service. Each peer only specifies the anycast address in order to participate in the logical network and one of the participating peers becomes the gate of the logical network for the new node, which should be determined by the anycast routing protocol. All nodes have the same service in this logical network.

So, even when the connected peer leaves the logical network, it is possible to continue participating via another peer, which is automatically changed by the anycast routing protocol. Figure 2.2 shows the Gate to Overlay Network in the Internet.

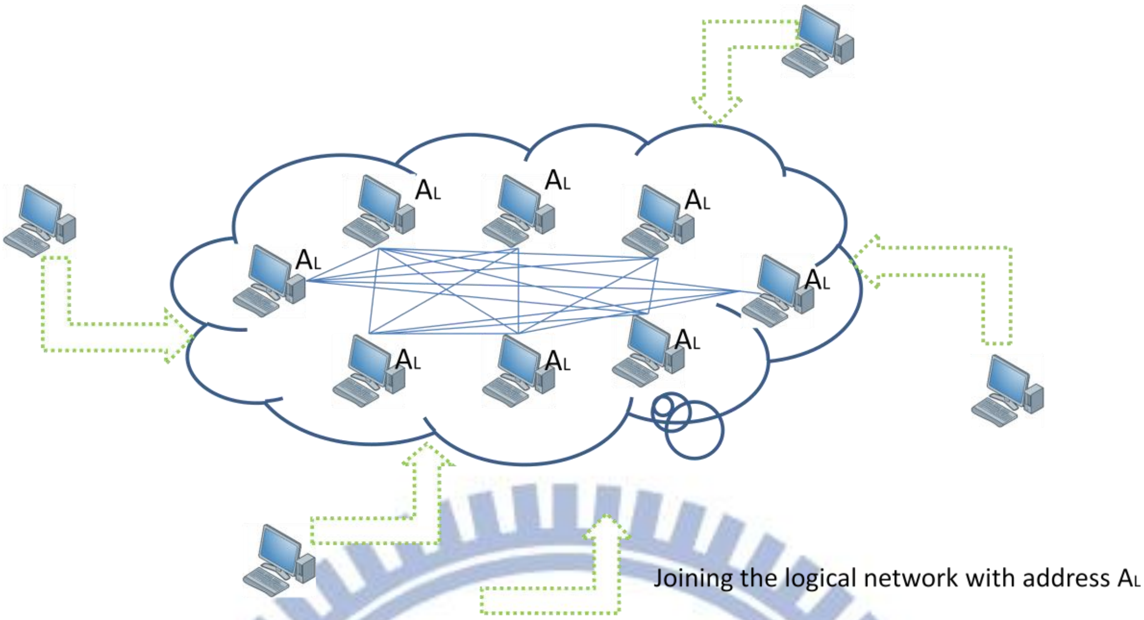


Figure 2.2 A Gate to Overlay Network.

DNS /WWW Redundancy

IP anycast can also support host auto-configuration by assigning an anycast address to the DNS service; after moving to a new network, a host can continue to contact the DNS anycast address instead of being re-configured with the new local DNS server. In this case, a host that is moved to a new network need not be reconfigured with the local DNS address. The host can use the global anycast address to access the local DNS server anywhere. There's the overall redundancy that each name server instance achieves by distributing DNS service for the same IP address across multiple data centers. A global distribution of DNS name servers significantly reduces the latency for users to reach / obtain a DNS response when connecting to a site because of the distributed nature of the network.

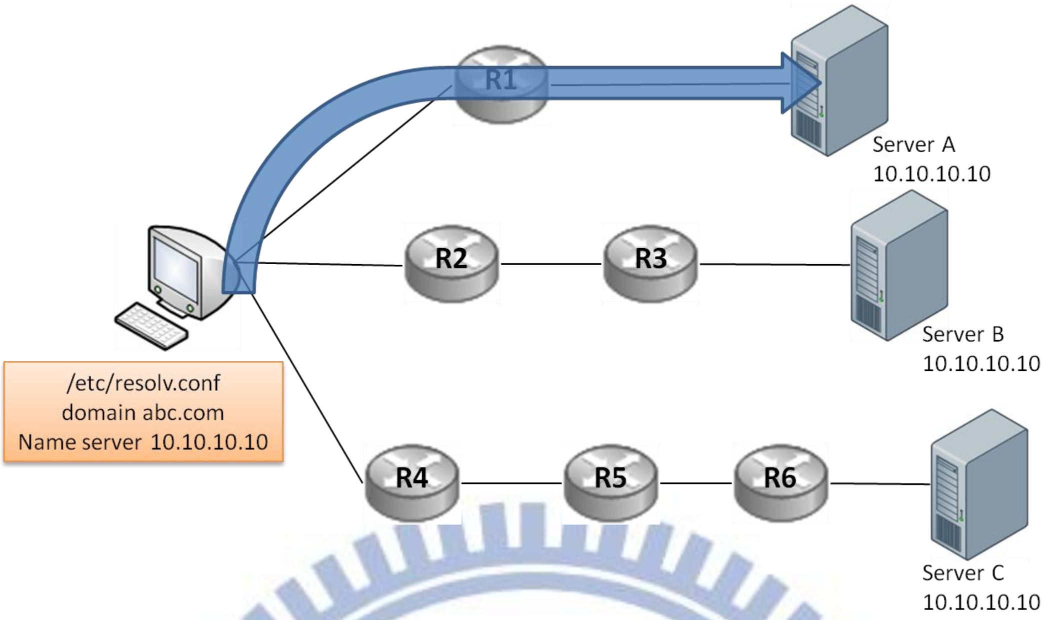


Figure 2.3 Anycast DNS.

Figure 2.3 shows an example of Anycast DNS. A single DNS client workstation, configured with the anycast DNS IP address of 10.10.10.10, is shown performing DNS resolution against its "closest" of three DNS name servers deployed using the same anycast IP address.

SIP Proxy Server Discovery

The session initiation protocol (SIP) is increasingly becoming the de-facto standard for VoIP deployments in fixed and wireless networks. Proxy server is an optional SIP component that handles requests or forward requests to other servers. Every such SIP proxy server SHOULD configure this anycast address on the IP interface that it uses to communicate with SIP clients. In a network there can be multiple SIP proxy servers that are configured to act as outbound proxy servers and all of these servers SHOULD configure this anycast address on the appropriate IP interface. Figure 2.4 shows the SIP Proxy Server.

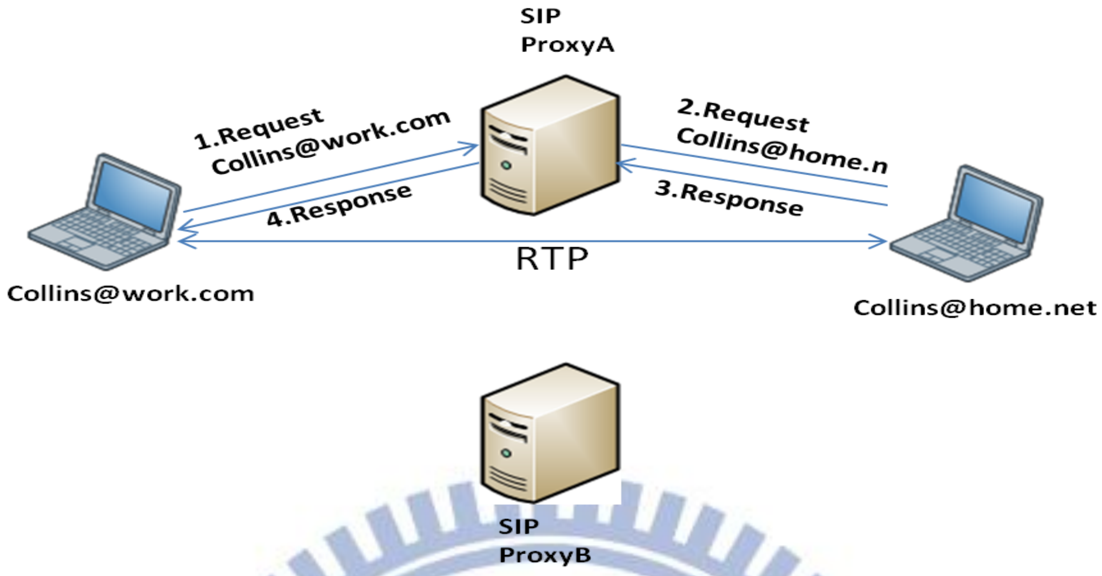


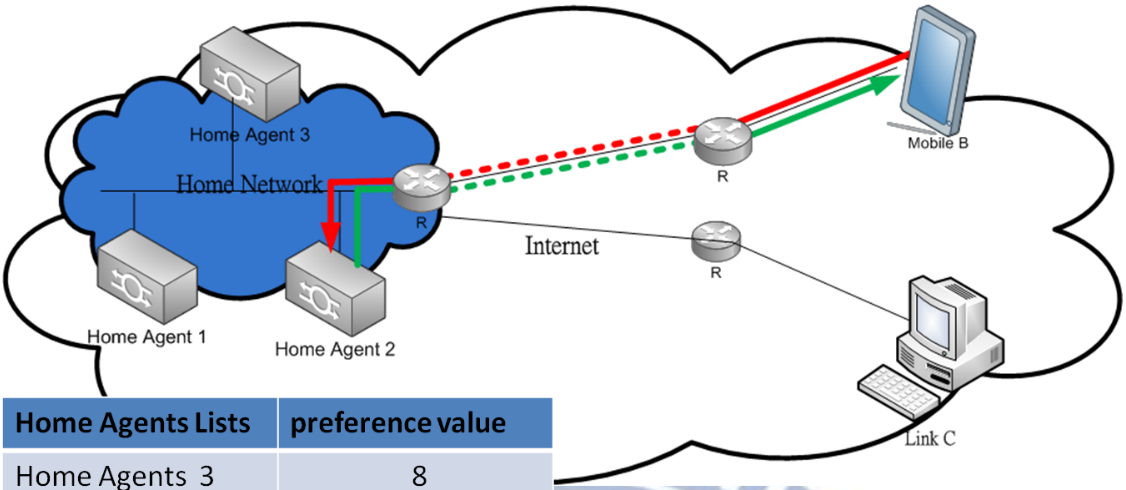
Figure 2.4 SIP Response Through a Proxy Server.

FTP Server Discovery

IP anycast can provide automatic service discovery in the Internet. By assigning the same anycast address to replicated FTP servers, users can download from the closest server without manually choosing from a list of mirrors. Users can get access through the best shortcut to any server that provides the same service. The user can use the anycast address to directly download the file from the nearest replica.

Mobile IPv6 Home Agent Discovery

The Home Agent (HA) [13] concept from having a single entity on the home link towards several geographically distributed Home Agents. The home agent list is sent to a mobile node when the mobile node requests the latest list of home agents. Every home agent has a special anycast address, called a home agent anycast address. A problem that has to be solved within this context is locating the closest HA to the Mobile Node (MN) or Mobile Router (MR), for which the Anycast based Dynamic Home Agent Address Discovery mechanism. The semantics of Anycast are suitable for this task as data is routed to the “nearest” destination, with distance being defined by the routing metrics.



Home Agents Lists	preference value
Home Agents 3	8
Home Agents 1	4
Home Agents 2	1

Figure 2.5 Replying to a Dynamic Home Agent Address Discovery Reply message.

Figure 2.5 show the Replying to a Dynamic Home Agent Address Discovery Reply message. A Dynamic Home Agent Address Discovery request message is delivered to one of the home agents in a home network thanks to the anycast address mechanism. The home agent that receives the message will reply to the mobile node with a Dynamic Home Agent Address Discovery reply message containing all of the home agent addresses which the home agent currently knows. The address list is ordered by the preference value of each home agent. If a mobile node does not receive a reply message, the node will resend a request message.

2.2 Anycast Address to Unicast Address

Anycast technology may be used for connecting to a member of anycast group, but changing network configuration may cause the connection directed to another server. Figure 2.6 show that changing the network configuration causes consecutive packets to be sent to another server.

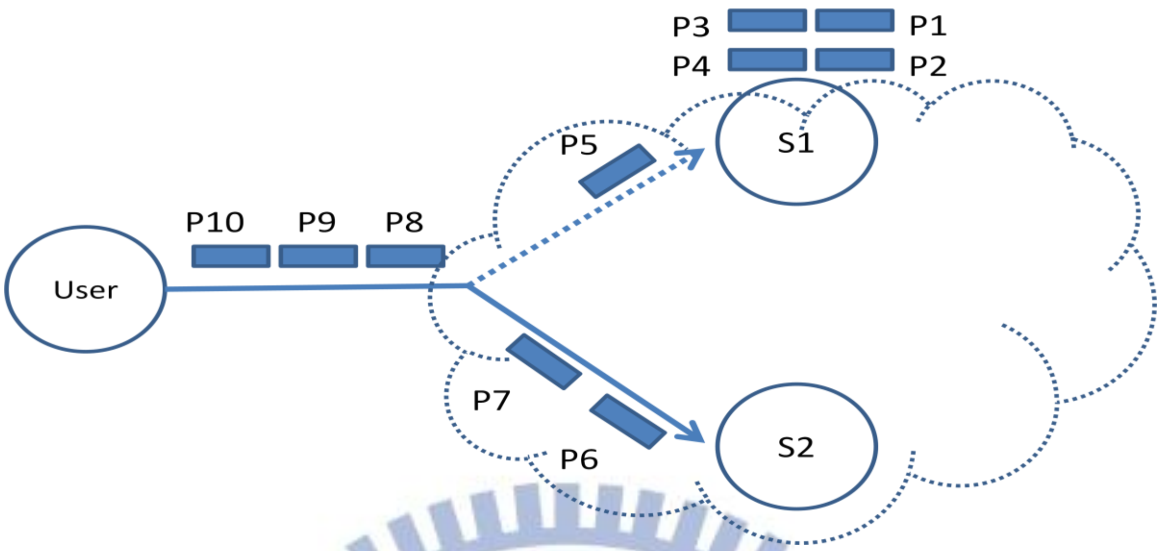


Figure 2.6 Network configuration changed.

A user sends some data to server 1 that receives consecutive packets, but server 1 happened to crash in the anycast group, so server 1 only receives the no.1~no.5 packets, but no.6~no.10 packets have not been sent to the same host. Because server 1 and server 2 did not get all packets, the server can't recognize the user's related message. A routing path between certain nodes may vary with changes in the network configuration. Therefore, packets using an anycast address may not be sent to the same node when changes on the routing path occur. A solution for these issues has been proposed in [14].

An anycast address should be used only at the beginning of communication, so that influences of any changes on the routing path can be prevented. Sending a packet with an anycast address can be considered as a way to find out the end point of the point-to-point communication. After the end point is found, the unicast address which is assigned to the end point should be used for actual protocol processing. To provide this function, the mechanism makes a mapping from an anycast address to its corresponding unicast address is required. To achieve this mapping, two mechanisms are proposed:

Source Identification Option

A client sends a SYN [1] packet to an anycast address M. A server with the anycast address M receives the SYN packet. Instead of using anycast address M as the destination address, the server replies to the client by sending the SYN+ACK packet with its own unicast address B as a source address and attaches "Source Identification Option" as destination option. When the requesting client receives the SYN+ACK packet, it extracts "Source Identification Option" in the packet header and confirms if the packet is replied to the SYN packet. Then, it changes the server address from M to B. The client opens the connection by sending the ACK packet to the unicast address B (Figure 2.7).

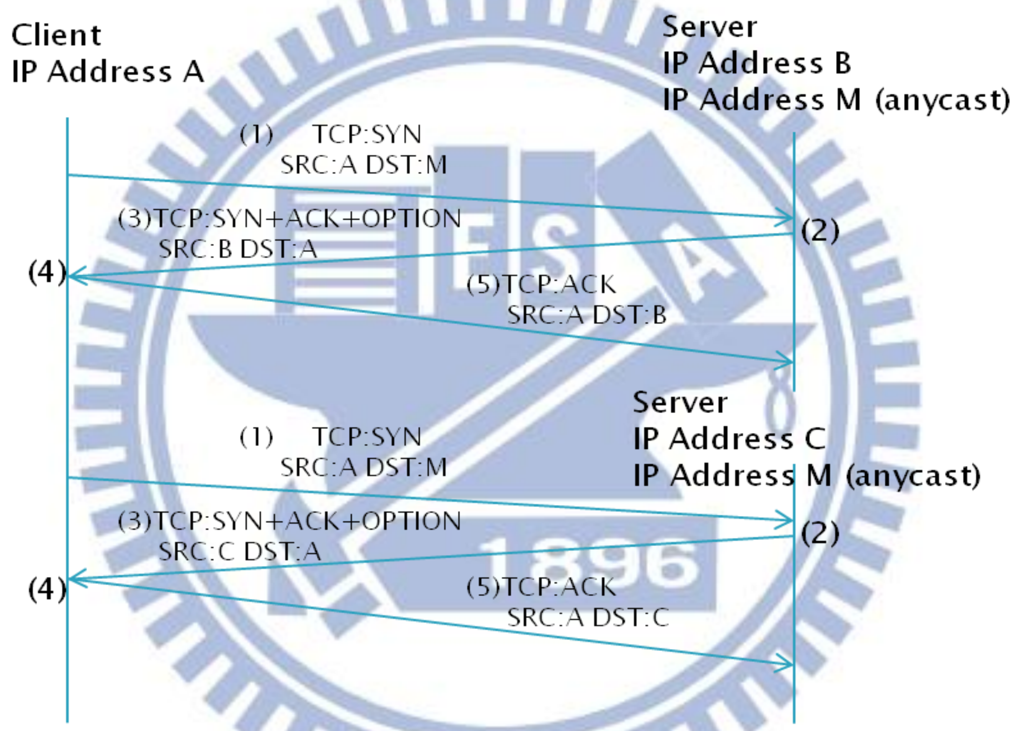


Figure 2.7 Source Identification Option Protocol.

Anycast Address Mapper

Figure 2.8 shows the Anycast Address Mapper Protocol using the ICMP [15] method to find unicast address of anycast group members. If it wants to find out the unicast address corresponding to the anycast address, then using ICMP ECHO request/reply. Before establishing a connection to a server with an anycast address, clients can find out the unicast address by using "Anycast Address Mapper". Clients can use this unicast address to make a connection with the server.

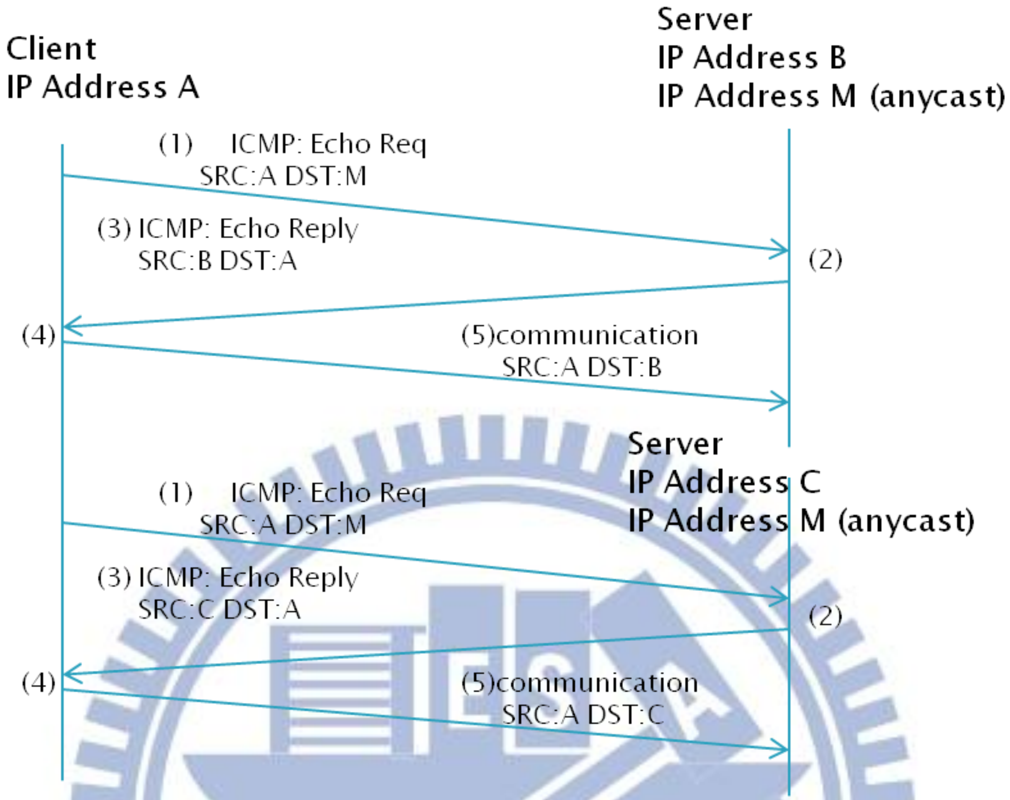


Figure 2.8 Anycast Address Mapper Protocol

"Anycast Address Mapper" determines the unicast address by using ICMP ECHO request/reply and maintains the mapping of anycast and unicast address. The “Anycast Address Mapper” is implemented as a daemon in the user memory space. While transport and IP layers are not changed, we have to modify the application in order to use "Anycast Address Mapper".

2.3 Related Works

I. Round-Robin Domain Name System

The case uses Round-Robin protocol at domain name system in application layer. It doesn’t care about which server supports service that anycast [16].

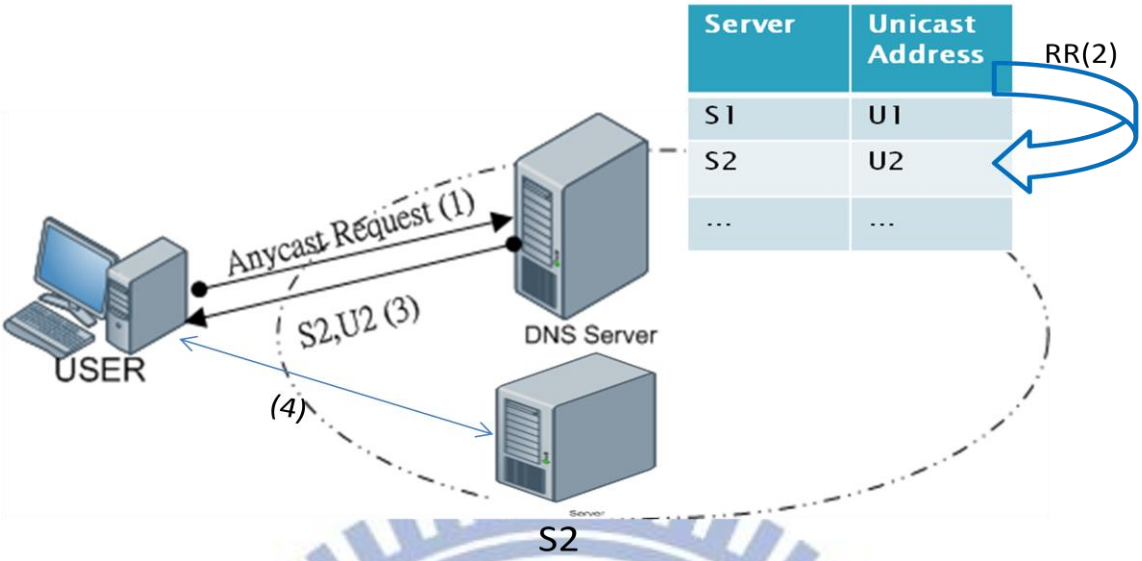


Figure 2.9 Round-Robin Domain Name System

Figure 2.9 shows the Round-Robin Domain Name System operation to find a server in the network. If a user wants to find a server through anycast, user can send an anycast request to DNS server. Round-Robin Domain Name System is responsible for distributing user requests across the cluster. Domain Name System sends uncast address of S2 to user based on Round-Robin method. User then accesses the server S2 to which packets from U2 will be sent via unicast.

II. Anycast name resolvers

The anycast name resolvers [17] are like Domain Name server and work in application layer, the main function is anycast name resolving that translate anycast to unicast IP address. Before a client wants to use the service, it sends an anycast query message to anycast name resolvers.

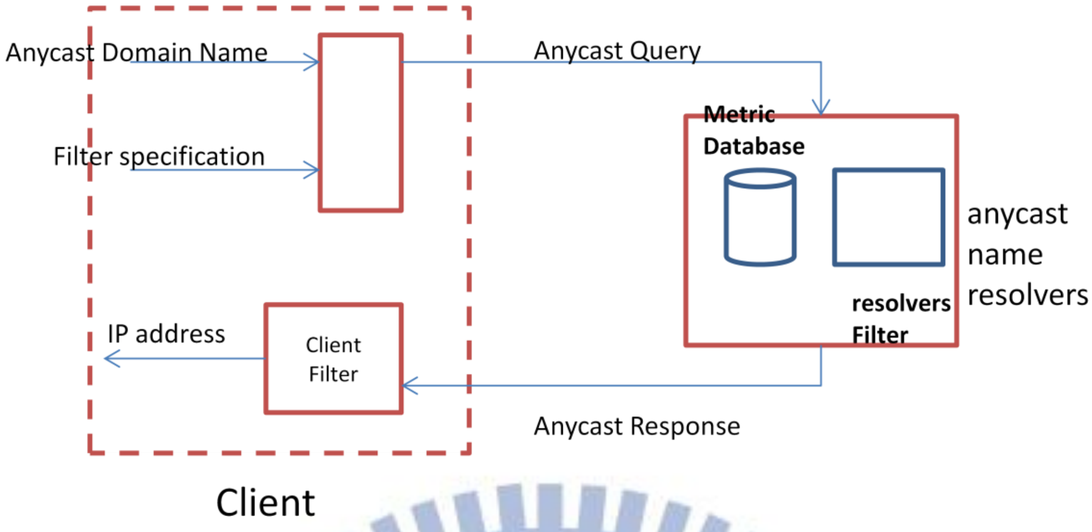


Figure 2.10 Anycast name resolvers.

In anycast name resolver architecture, a client connects with an anycast group via a query–response protocol, as illustrated in Figure. 2.10. The resolver receives the anycast query and applies a filter to choose anycast group members. A resolver filter operates on a set of anycast group members. A resolver filter can returns a (possibly empty) subset. A client filter may be applied to the client. Two Filters may be content-independent (e.g., select any member at random), or based on performance metrics or policy information. The metric database records anycast group members information and uses those information to choose service server. So anycast name resolvers need some information in this method: (1). Anycast group members address. (2).All member information: distance, response time, load of server. Those members' information needs sending test packets to all members at short intervals.

Chapter 3 The Proposed Scheme

The design principles we made in our anycast and manycast services are based on Explicit Multicast are as follows. In this chapter, we present a system model and describe our scheme in detail to illustrate how the system works.

3.1 System Architecture

We propose a method to translate an anycast address to a unicast address for anycast service. The first step is sending the anycast address to the DNS server. After translating the anycast address, the DNS will send back a reply message to the requesting source. Therefore, the source can obtain a set of unicast addresses by extracting the reply message of the DNS server. After that, the source will put the unicast addresses into Xcast Ping header and send it to the host addressed by unicast addresses. Figure 3.1 shows the system flow of our architecture.

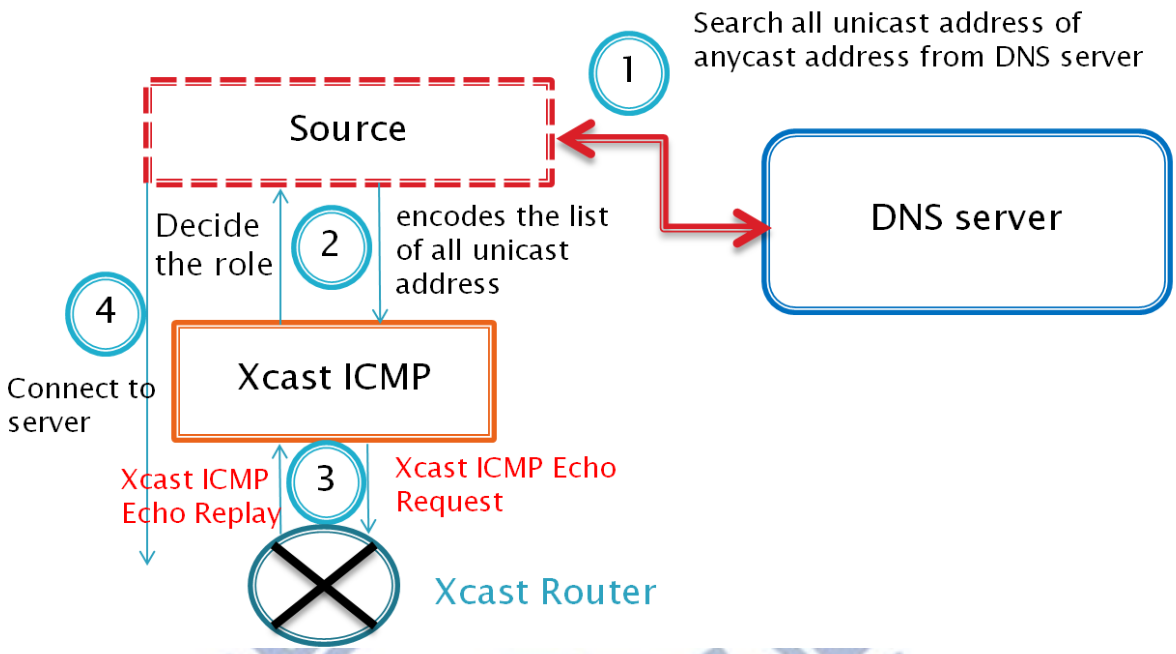


Figure 3.1 Our system architecture.

In the following, we describe every step in Figure 3.1 in more detail.

Step1. If an application wants to apply the application of an anycast service, in our system it will get all unicast addresses of anycast group from the DNS server. In DNS server, there is one component (module) for translating the address called the Anycast To Unicast (ATU) resolver. The job of ATU resolver is to obtain all unicast addresses from the anycast address and reply the individual unicast addresses back to the source sender.

Step2. In Step 2, after receiving the unicast addresses, the source begins sending an Xcast ICMP echo request to all unicast addresses via putting all unicast addresses in the extension header for Xcast.

Step3. The first reply from any of the destinations is considered as the server with highest priority of connection. It reduces the time to search for the service provider compared with the traditional scheme which will be iteratively sending individual unicast address to each server. Step 3 aims at topologically locating the closest server or the server with the quickest response time.

Step4. The source will establish the connection to the dedicated service.

3.2 DNS Message Header Format

In order to obtain all the IP addresses from ATU resolver, the source must sends DNS query message to DNS server. The DNS server sends DNS response message of uncast addresses which specify the number of IP addresses in Question section[1].

The ATU resolver provides a mapping table between the anycast address and unicast address. Let's look at the way in which the DNS server carries out the mapping process. The corresponding address procedure is illustrated in Figure 3.2. The ATU resolver translates one anycast address into many unicast addresses.

Step1. The source sends an anycast address to DNS server.

Step2. The ATU resolver performs lookup in its mapping table to find the corresponding unicast address.

Step3. The DNS server sends all associated unicast addresses to the source.

[1] The question section of DNS query message is used to store the IP address.

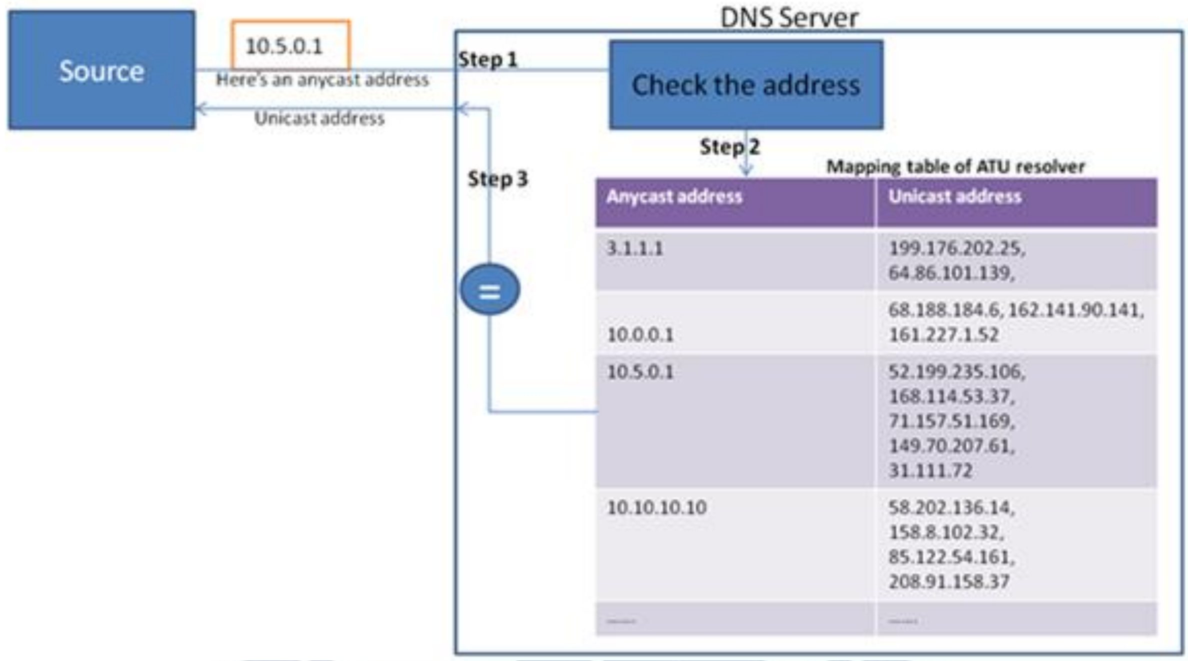


Figure 3.2 Anycast to unicast address mapping table.

In our DNS, it can consult the ATU resolver to find out the mapping between unicast address and anycast address. The mappings between anycast address and unicast address are stored in Question Section shown in Figure 3.3. The source generates the DNS Query message with anycast address and sends it to the DNS server. The response messages from DNS server contain the unicast addresses. Therefore, the source can encapsulate the unicast addresses into one field of the extension header for Xcast header and forward it to the Xcast router[2] further searching for the closest service provider.

[2] Xcast router lookups all the destinations in its unicast routing table, duplicates and forwards the packet to appropriate network interfaces.

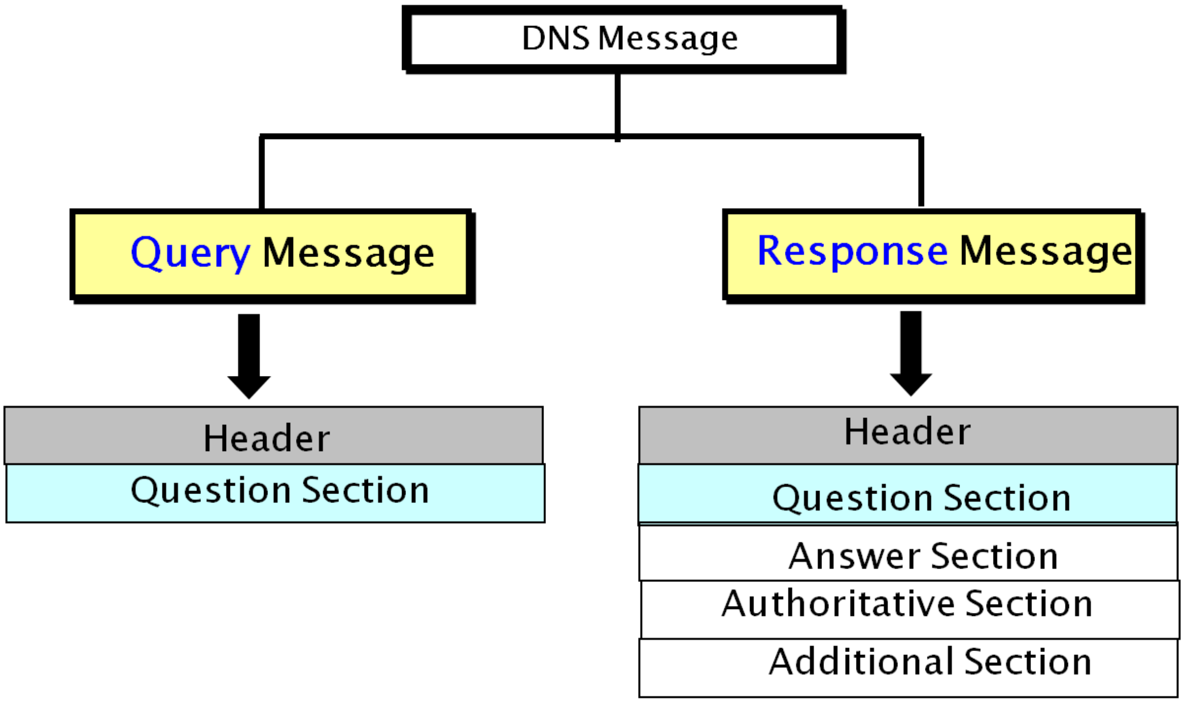


Figure 3.3 DNS Message Header Format.

3.3 Extended ICMP Services Using Explicit Multicast

Explicit multicast (Xcast) is a complementary protocol that can solve the scalability problems of IP multicast by providing a stateless design that encapsulates multicast information in an IP packet header. The structure of a packet is composed of IP header, Xcast addresses of destinations, ICMP header, and effective payload. Figure 3.4 shows the Xcast ICMP protocol structure.

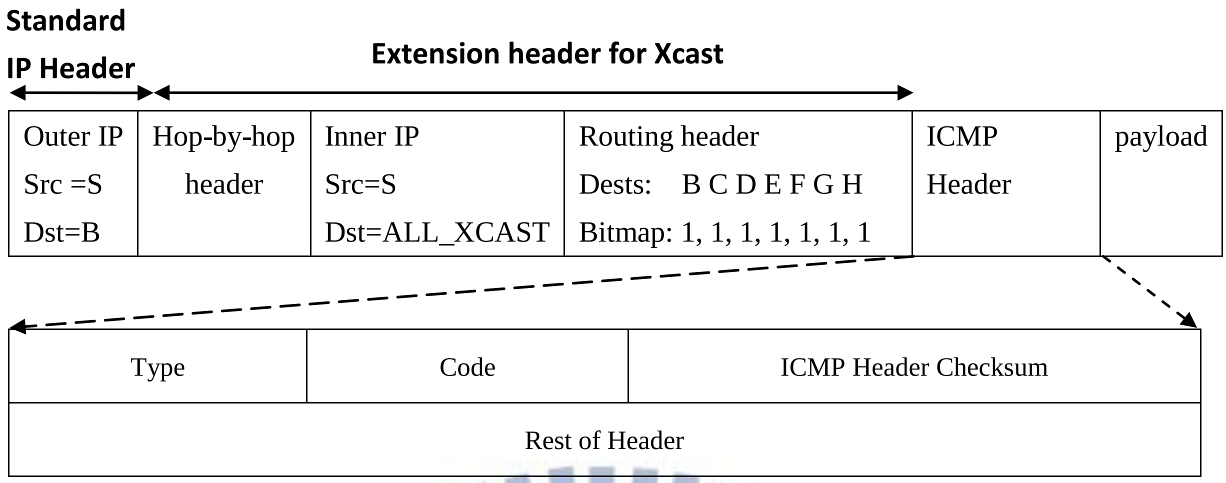


Figure 3.4 Extended ICMP packet format

In summary, Xcast has a list of IP destination addresses and a list of bitmaps as shown in Figure 3.4. Bitmap = 1 is used to mark end-hosts which have not received the packet yet. For every received Xcast packet, Xcast-aware router lookups all the destinations in its unicast routing table then duplicates and forwards the packet to appropriate network interfaces. Figure 3.5 illustrates how the bitmap of Xcast routing header works in our extended ICMP service.

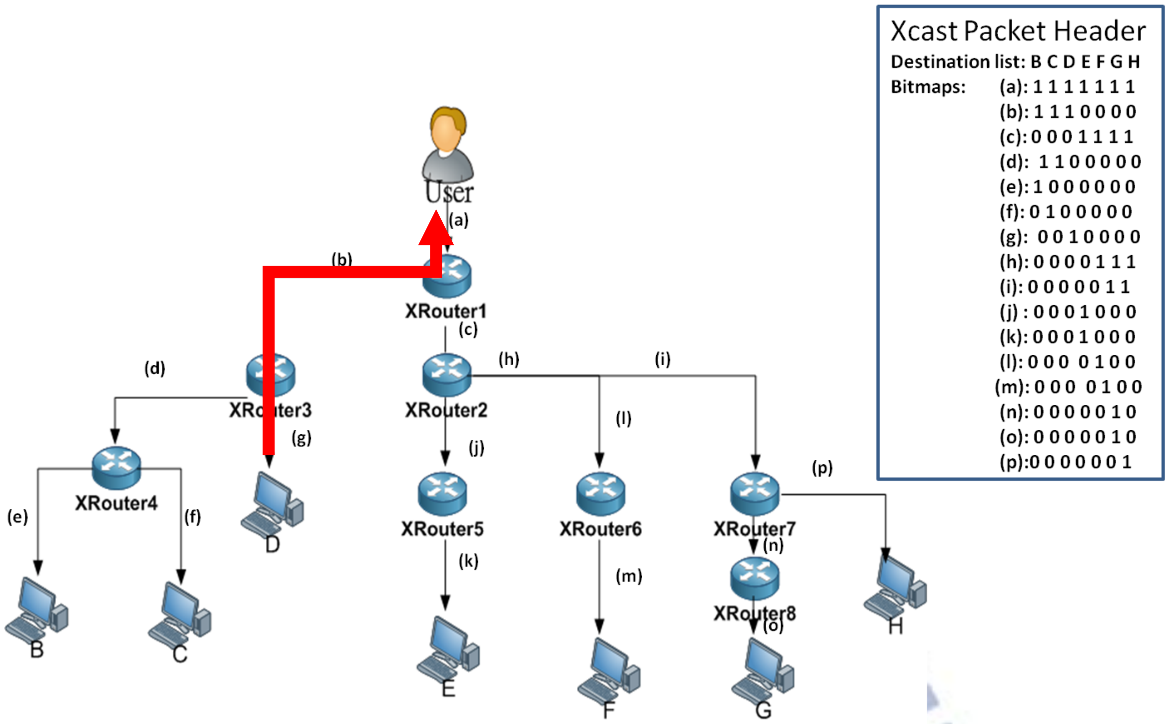


Figure 3.5 Bitmaps of Xcast header.

The source S sends packets with the "outer IP" [src = S, dest = B], the list of destinations and the bitmap are shown in Figure 3.5。XRouter1 receives an Xcast ICMP packet, looks up all the destinations in its unicast routing table. Then, XRouter1 duplicates, changes the bitmap and forwards one packet to XRouter3 and the other packet to a group of (E, F, G, H). According to the tree, XRouter2 has three branches, then it forwards one packet to a group of (G, H). Similarly, B sends another packet to a group of (F) and another packet to a group of (E). Eventually all the destinations (B, C, D, E, F, G, H) get message from A and send the packet which changes ICMP type to A. Because the Round-Trip Time (RTT) between A and destination D is shorter, A will go to destination D.

Table 3.1 Two new types in Xcast ICMP Header and other types in ICMP Header.

Type	Description
0	Echo reply.
1	Xcast Echo request .
2	Xcast Echo reply.
3	Destination unreachable.
4	Source quench.
5	Redirect.
6	Alternate host address.
8	Echo request.

In the original ICMP header, the type 0 is echo reply and the type 8 is echo request. Table 3.1 shows the 2 new types we created in Xcast ICMP Header. The *Type 1* is Xcast Echo request and The *Type 2* is Xcast Echo reply. The Xcast ICMP messages we can use are Xcast echo request and Xcast echo reply.

Our approach allows us to obtain the Round Trip Time (RTT) from source to all anycast and manycast destinations by using Xcast ICMP protocol for each destination. Additionally, we use Xcast trace route to reduce the time and the traffic volume on Internet. Moreover, we decide the minimum end-to-end delay that refers to the time taken for a packet to be transmitted across a network from source to destination. Our proposed Extended ICMP service for measuring network delay based on observing general application packets provides the most accurate representation of performance.

Our proposed Extended ICMP service can be utilized in the application of P2P IPTV system, especially in dealing with the issue of partner selection. Because our proposed approach can speed up finding out the closest partners watching the same video contents. Therefore, we can improve not only the start-up delay of P2P IPTV service but also the channel switch delay as long as we can quickly find out content providers.

3.4 The Approaches for Improvement of Anycast and Manycast

In this section, we describe how our approaches can improve the traditional anycast and manycast operation by the call flow. The subsections are the improvement of both system operations for anycast and manycast.

3.4.1 The Improvement of Anycast System Operation

The handover procedures in the Anycast improved by our proposed Extended ICMP service are described in Figure 3.6 as follows. One source communicates simultaneously with 2 servers providing the same service in an anycast group. The IP address “72.30.38.140” is the first one that responds to our source.

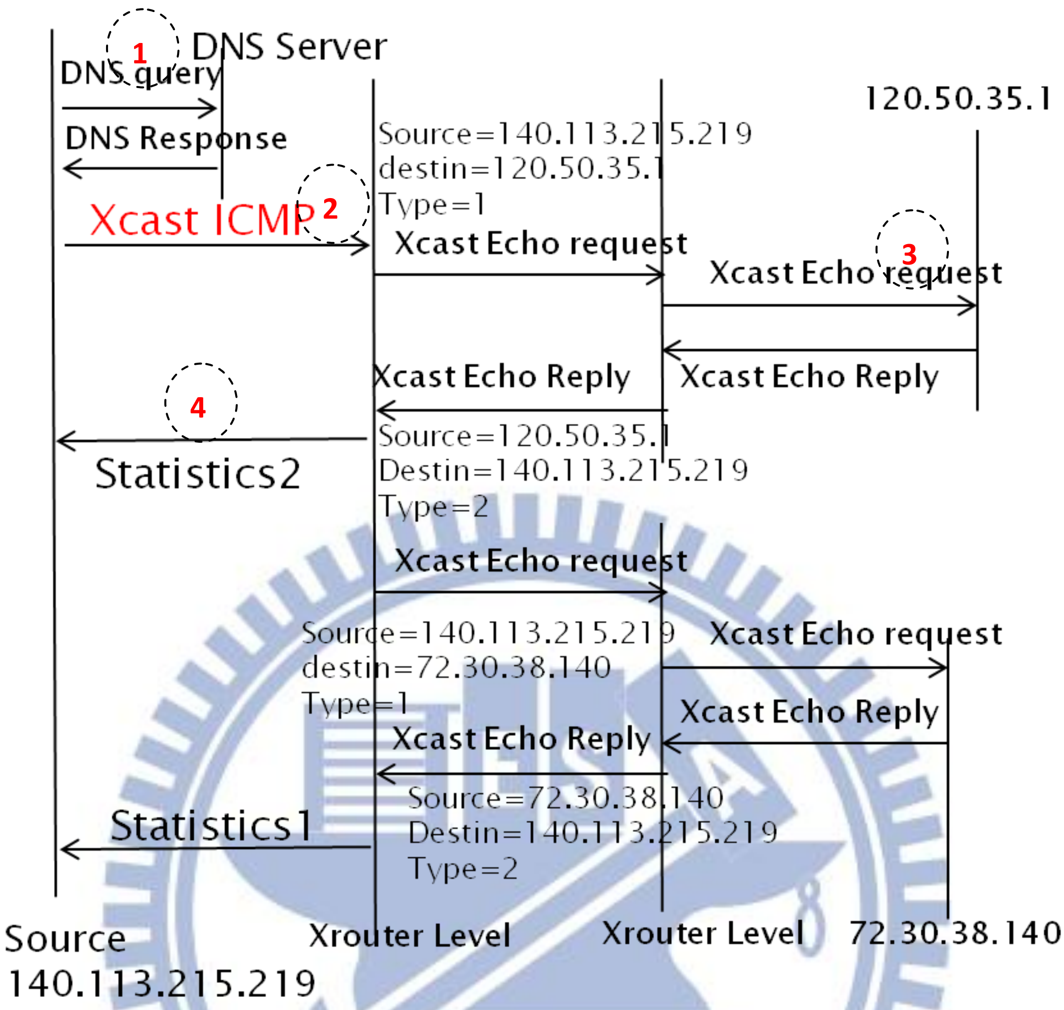


Figure 3.6 Call flows of Anycast Operation using Xcast ICMP.

1. At the beginning, the source “140.113.215.219” will send DNS query message to the DNS server. After ATU resolver finishes translating the anycast address to individual unicast addresses, DNS server will put them into the field of question section[3] and reply the message containing the unicast addresses back to the source.
2. In step 2, Xcast ICMP service sends an ICMP Echo command to the server addressed by unicast addresses.
3. The ICMP Echo message is routed through the network until it reaches the destination Server (120.50.35.1, 72.30.38.140…etc).

[3] The default message length limits the DNS response size to 512 bytes. DNS message header is 12 bytes. It will include 125 addresses in Question Section of DNS response message.

4. Xcast ICMP service uses the ICMP Echo and Echo Reply handshake message for this purpose. Every server sends ICMP messages (Reply) with the type 2 (Echo Request) back to the target source.

When we receive the first echo message from a server, we are able to connect to it. Clients can select one of the replicated servers which are logically "closer" to them according to the response time from the serving server. To find the first responsive server is a network addressing and routing methodology in which datagrams from a single sender are routed to the topologically nearest node in a group of potential receivers all identified by the same destination address.

If the serving server fails, our method will repeatedly send Xcast ping to find another server and obtain the same service for source. Anycast is normally highly reliable, as it can provide automatic failure recovery and save network traffic. If source iteratively applies traditional unicast to find server, it would cause heavy traffic load on the Internet. Furthermore, compared with the traditional scheme, our method can save network bandwidth by the approach of extended ICMP service.

3.4.2 The Improvement of Manycast System Operation

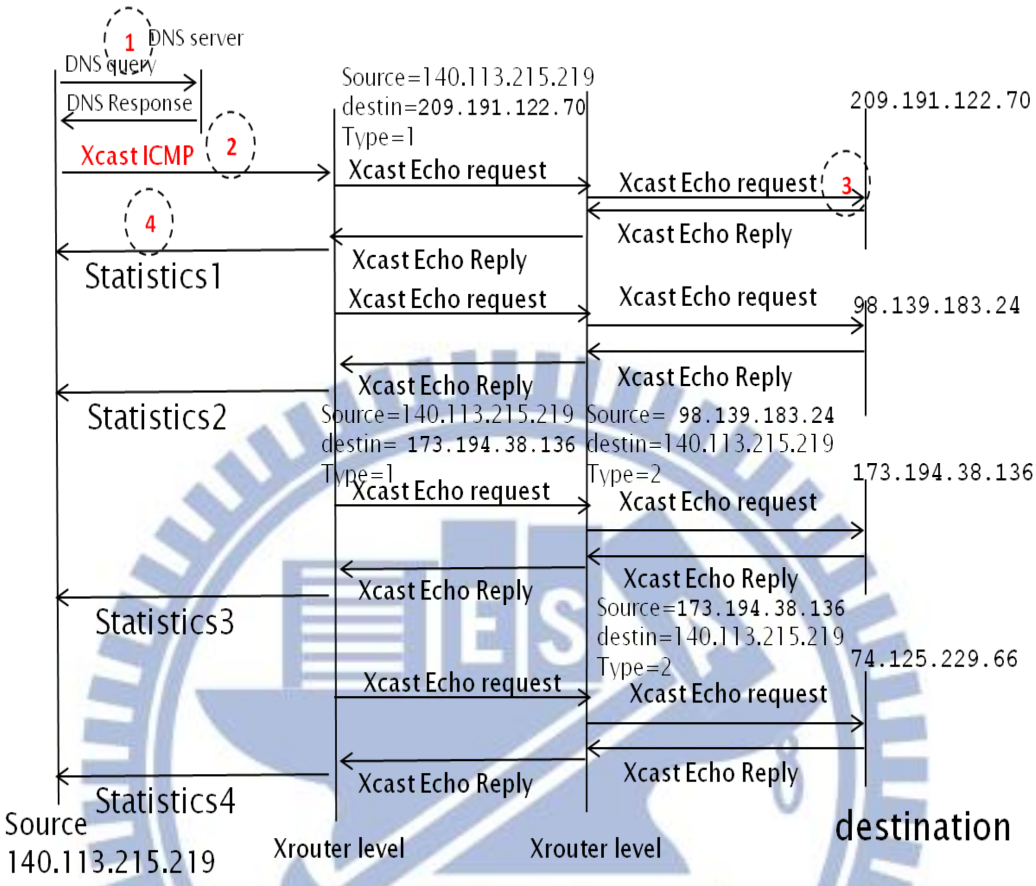


Figure 3.7 Call flows of Manycast Operation by Xcast ICMP

The handover procedures in the Manycast by Xcast ICMP Call flows mode are described as the illustration of Figure 3.7. One source communicates simultaneously with 2 of 4 equivalent servers in a group. Now, “209.191.122.70” and “173.194.38.136” are the fastest two servers that respond to the source.

1. Firstly, source will send DNS query to DNS server which will find out all the unicast of manycast address from DNS table. And DNS server gets the individual unicast addresses putting into the question section. Then, DNS server sends DNS responses to source.
2. The ping application of source sends an Xcast ICMP Echo command to the servers

of unicast addresses.

3. The ICMP Echo message is routing through the network until it reaches the destinations. All the servers that receive ICMP message will send an Xcast Echo Reply to the source.
4. The source will receive k replies from k servers. The responses from the fastest m servers will be the m candidates of the connecting destinations.

The service selection allows clients to choose one of the replicated servers which are "closer" to them and thereby minimize the response time from the service provider.

Our scheme is based on the fastest m echo responses that we need to connect to. When some service providers fail based on the type of response message or no response before timer expires. Our approach will send Xcast ICMP again and search for alternative servers to obtain the same service for source.

As the mechanism of automatic failure recovery manycast possesses, the protocol is highly reliable. It provides a bidirectional channel for enabling request/reply communication between client and servers, not merely one-way dissemination of data. The source that needs a long-term or stateful interaction is best served by using a brief manycast transaction to discover m servers.

3.5 A Fast Partner Selection Scheme in P2P IPTV System

In this section, we apply anycast and manycast by Extended ICMP Service using Xcast to speed up the selection of partners in P2P IPTV system [18].

I. When selecting a tracker, we apply anycast of extended ICMP service using Xcast to obtain the fast response from service provider.
II. When selecting partners, we apply manycast of extended ICMP service using Xcast to

obtain the fast responses from several partners.

3.5.1 Extended Xcast ICMP Service in P2P IPTV System

With the development of Peer-to-Peer technique in recent year, P2P IPTV has aroused wide spread population. We can watch a lot of movies or dramas on the Internet and share the media contents with other people. Therefore, the cost of both the hardware and software can be saved due to the technique of Peer-to-Peer system. The application of P2P IPTV such as PPLive, PPStream, and UUSee are popular not only in China but also expanding all over the world. In P2P IPTV, users obtain the media data from their partners that watch the same media contents. In addition, tracker is responsible for providing the peer list to the newly joined peers or whenever they want to alter the media content. However, the burst increase of users in the system would cause heavy loading on the tracker because everyone in the system has the direct communication with tracker. In order to improve the availability and efficient load distribution of tracker, the replication of multiple trackers is adopted by most of the P2P systems.

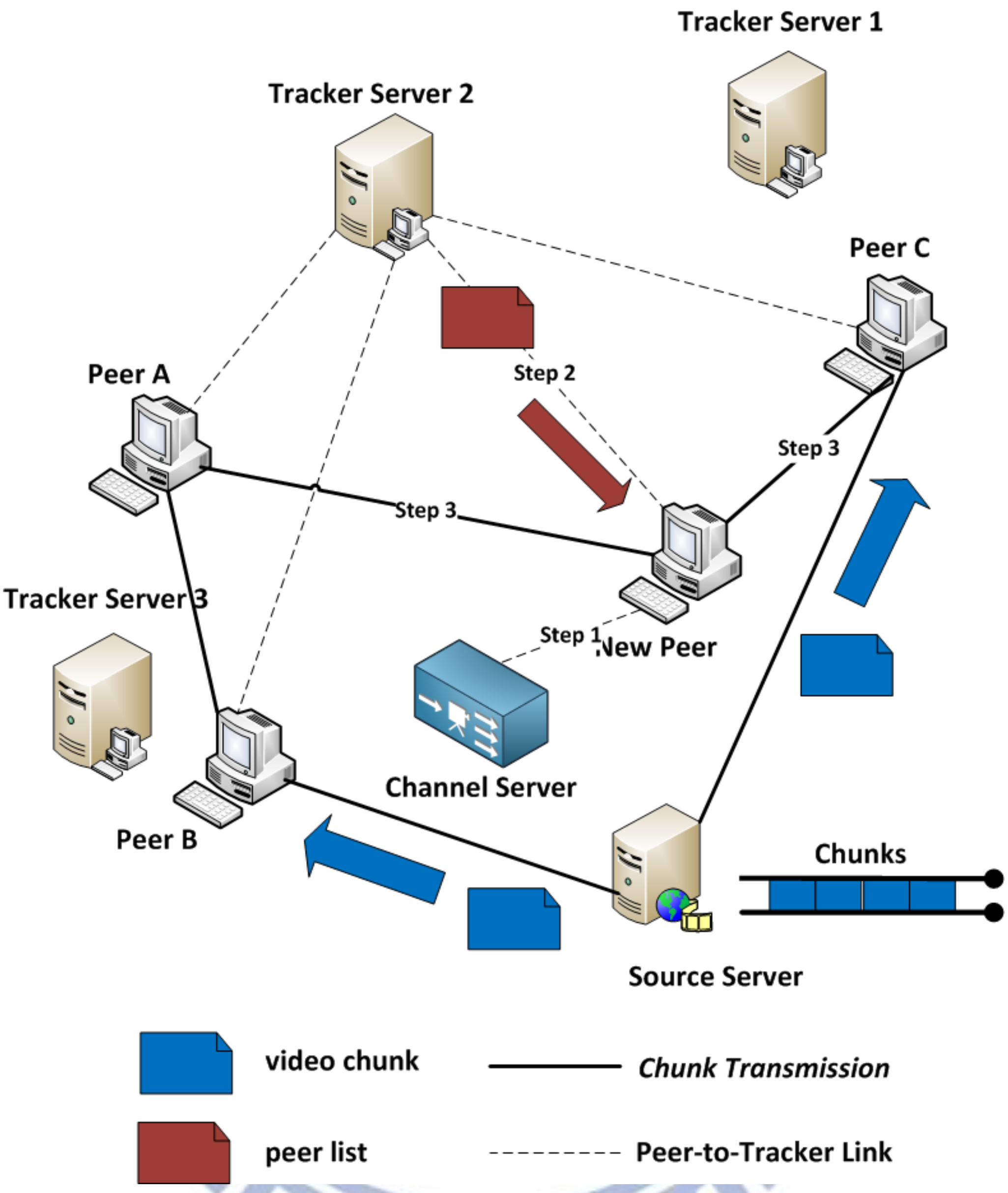


Figure 3.8 P2P IPTV streaming system flow.

As shown in Figure 3.8, mesh-pull P2P architectures have the following characteristics. A video stream is divided into the fixed-sized unit called chunk and is made available from an origin server for broadcast. All the video information is accessible for users at the channel server. And newly joined peer arbitrarily selects one tracker server to obtain information of peer list.

Step 1. A host, interested in viewing the video, requests the available channel from the channel server.

Step 2. We apply the anycast by Xcast ICMP service to find out the nearest tracker server among the three of them. After a user selects his interested channel, it then retrieves a list of active peers currently watching the same channel from tracker.

Step 3. The host establishes the partner relationships with a subset of hosts on the list. We apply our approach of manycast by Xcast ICMP service to find out some nearest partners in the peer list and establish the TCP/UDP connections with them. These peers help each other and deliver video traffic cooperatively.

3.5.2 Tracker Selection by Anycast of Extended Xcast ICMP Service

We explore the issue in which user selects a channel through channel server, from which we get the nearest tracker while reducing start-up delay. Our proposed scheme utilizes the anycast by Xcast ICMP service to obtain the nearest tracker which responds in the first place.

The Xcast ICMP packet format is shown in Figure 3.9, the host encodes the list of tracker addresses (T1, T2, T3, T4.etc) in the Xcast header, and then sends the packet to a router. In comparison with traditional scheme, the sequential unicast approach of getting a faster response from one tracker among all tracker servers, our proposed approach reduces the number of time the same path of packets' routing and improve network traffic to speed up the response time from tracker, thus results in reducing delay.

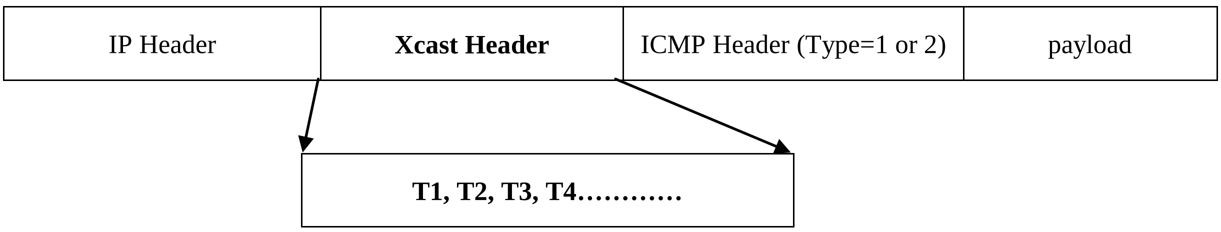


Figure 3.9 Xcast ICMP Reply packets.

1. User selects a channel that he likes and sends message to channel server, then

channel server send addresses of all tracker servers to user. The user explicitly specifies trackers' destination addresses as a list of unicast addresses embedded in the Xcast packet header. For every received Xcast ICMP packet, Xcast-aware router looks up all the destinations in its unicast routing table, duplicates and forwards the packet to appropriate network interfaces.

2. The tracker server that responds in the first place is considered as the suitable one from which we can obtain the peer list.

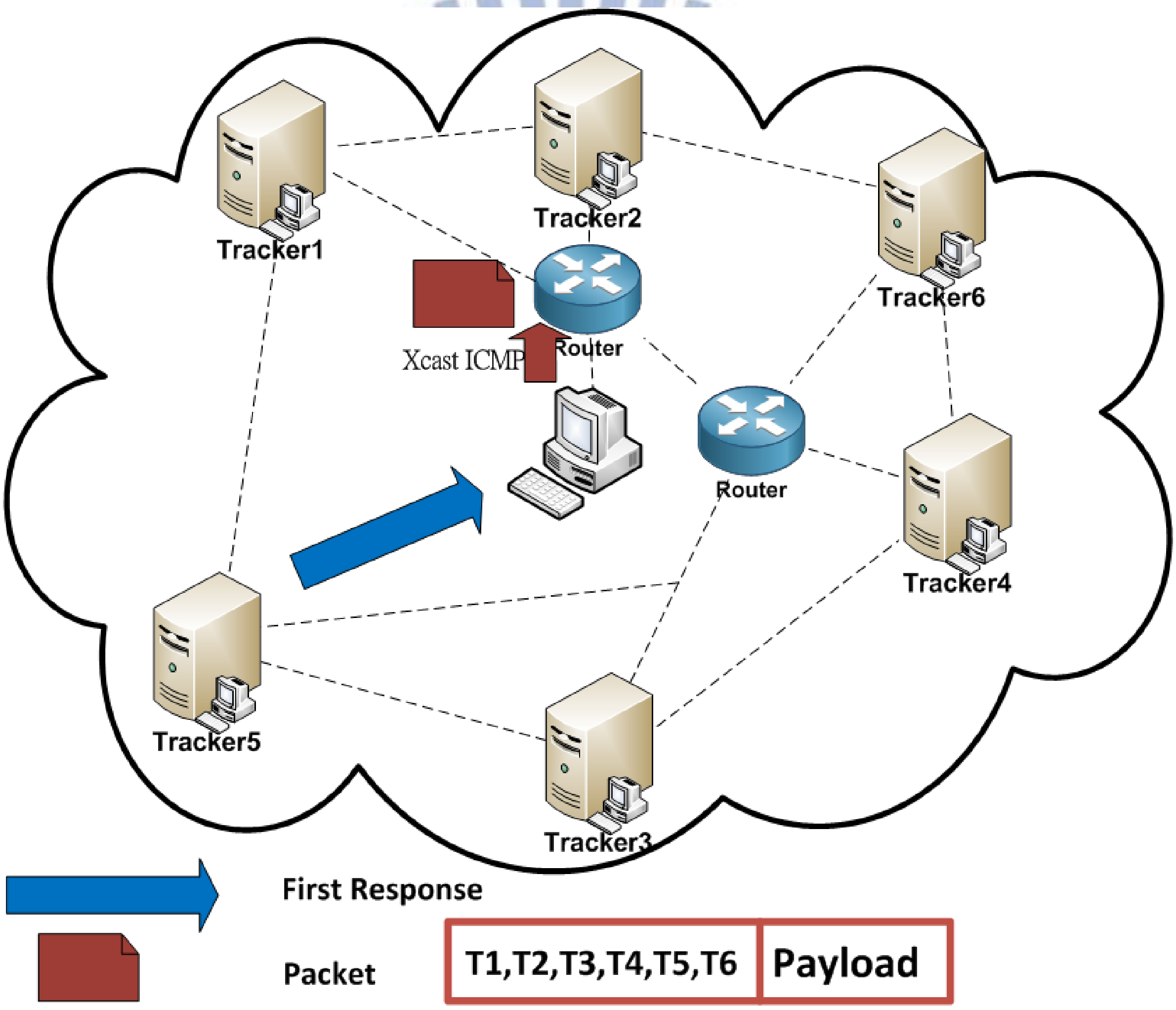


Figure 3.10 Tracker Selection by Xcast ICMP Service.

It is a Xcast ICMP application which is responsible for obtaining the appropriate tracker for requesting the peer list. In Figure 3.10, the user wants to find out the tracker server that responds in the first place. User obtains six addresses of tracker server by the means of translating one anycast address to six unicast addresses. A host sends an Xcast ICMP message with the list of tracker's destinations in the Xcast header. When router receives this packet, it needs to process the Xcast header and send to the next router. A router receiving one of these Xcast packets act as follows: it performs a route table lookup to determine the next hop for each of the tracker destinations listed in the packet, partition the set of tracker destinations and replicate the packet. That is, it modifies the list of trackers' destinations for a given next hop.

In the procedure of packet transmission, tracker1 is the first one that replies the echo message back to the source after it determines whether the type in the ICMP header equals 1. Therefore, tracker 1 is the best candidate for being requested peer list from the user.

3.5.3 Partners Selection by Manycast of Extended Xcast ICMP Service

We explore the issue when peer selects some partners to establish the connection in peer list. Although peer obtains all partners' addresses, how do we get the subset of peer list as peer's partners. We propose that the protocol of manycast by Xcast ICMP service can be applied to obtain the partners according to some fast responses in the peer list in P2P IPTV system. Our approach may bring the benefit that help peers download video chunks much easily, know what partners are nearest to us, improve users' quality of experience, and quickly find out the alternative partners in case that some of the original partners have left the P2P network or even switched to other program channel.

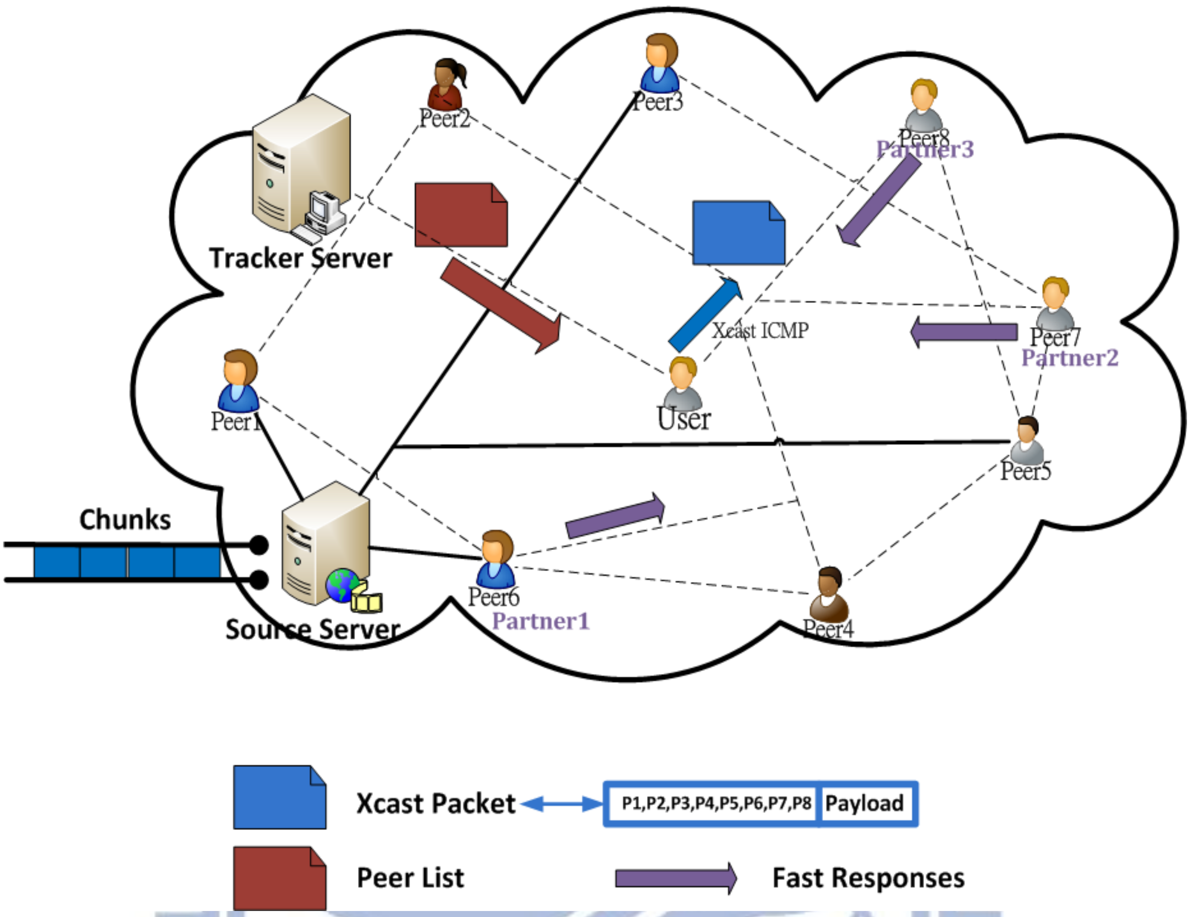


Figure 3.11 Partners Selection by Xcast ICMP Service.

1. User obtains peer list from a tracker server that responds in the first place.
2. User encodes the list of peer addresses (P1, P2, P3, P4. etc) in the Xcast header, and then sends the packet to a router.
3. User connects to the partners by several responses from members in the subset of peer list, then user can download streaming video from those partners.

Figure 3.11 shows the three steps regarding how the user establishes partner relationships by the fast responses from several members in its partner list of 8 peers watching the same channel.

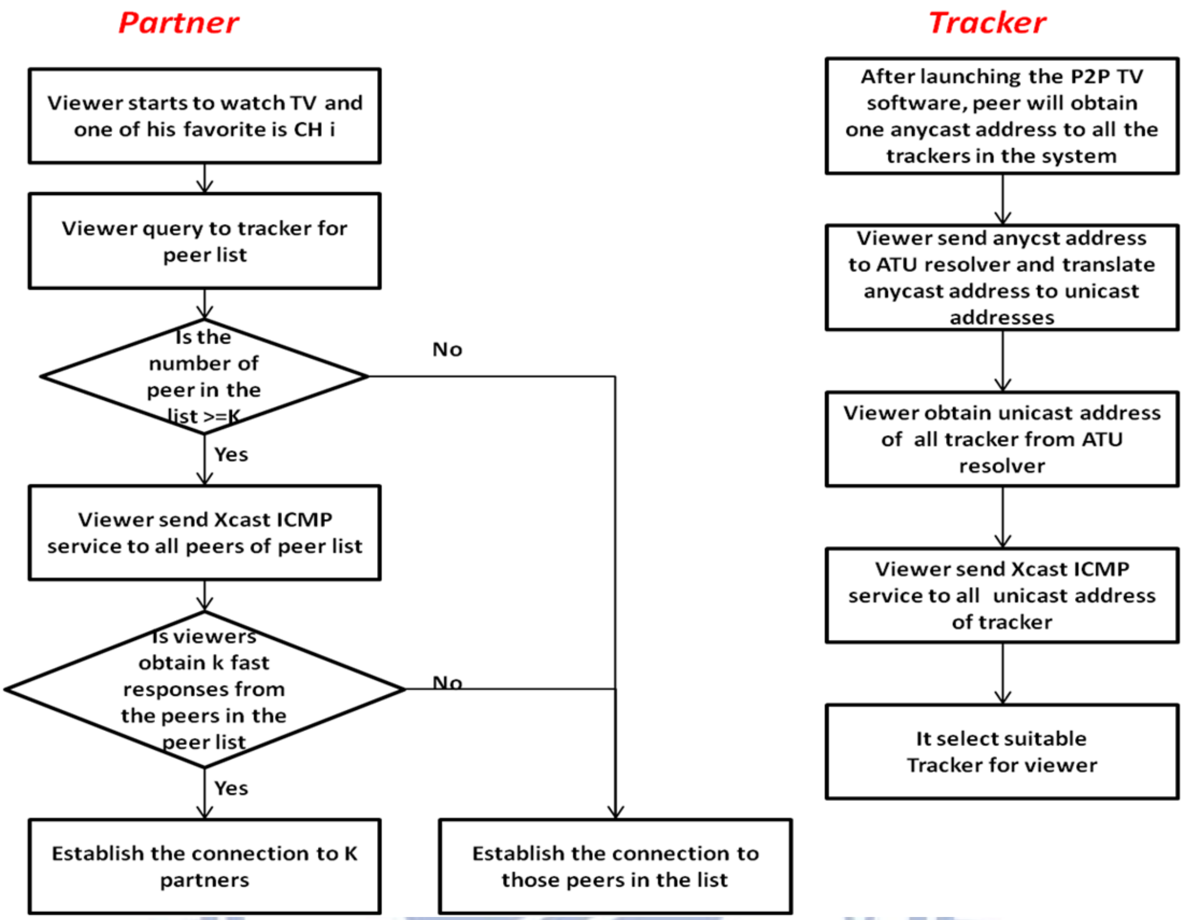


Figure 3.12 Flow Chart of selection Tracker and Partners

The procedure of selecting the tracker and partners via our approaches in P2P IPTV is illustrated in Figure 3.12.The viewer starts to watch TV and query to tracker for peer list. Then the P2P system selects some members in the peer list as its partners. Firstly, in order to find out the nearest tracker, The P2P software will obtain one anycast address containing all of trackers' IP addresses. It then sends anycast address to ATU resolver for address translation. After the reply containing the unicast addresses comes back to the viewer, the P2P software starts Xcast ICMP service and sends ICMP Echo command to the server addressed by unicast address. Through Xcast ICMP Service, the P2P system of the sender side can realize what members are available in the peer list and sends requests to those for establishing connection. The responses from the fastest m peers will be the candidates of the connection destinations.

Chapter 4 Simulation and Results

This chapter introduces the simulation environment first, and then we present the simulation results in conventional scheme and proposed scheme. In each simulation scenario, we compare the response time and network traffic of the two schemes. In the last section, we will illustrate the average response time of partners selection in the application of P2P IPTV.

4.1 Simulation Environment

We design an in-house simulator to evaluate the response time and traffic volume. We use 20,000 real routes of 20,000 different IPs (traced by window tracert) to construct the topology. We design several experiment with C language to evaluate the performance of our propose scheme. We collect the end-to-end delay and traffic volume on the Internet. We could use Xcast trace route to reduce both the time and the traffic volume, since the Xcast packet is also compatible with ICMP protocol.

In addition, in order to evaluate the response time when selecting the partners in P2P IPTV we use OMNeT++ simulator to observe the result. In OMNeT++ simulation, tracker is responsible for channel watching information of all peers. Tracker is also responsible for collecting the status of all the peers in the system and providing the peer list to the requesting peer. The number of peers in the peer list contains about 1~20 peers. The network topology is shown in Figure 4.1.

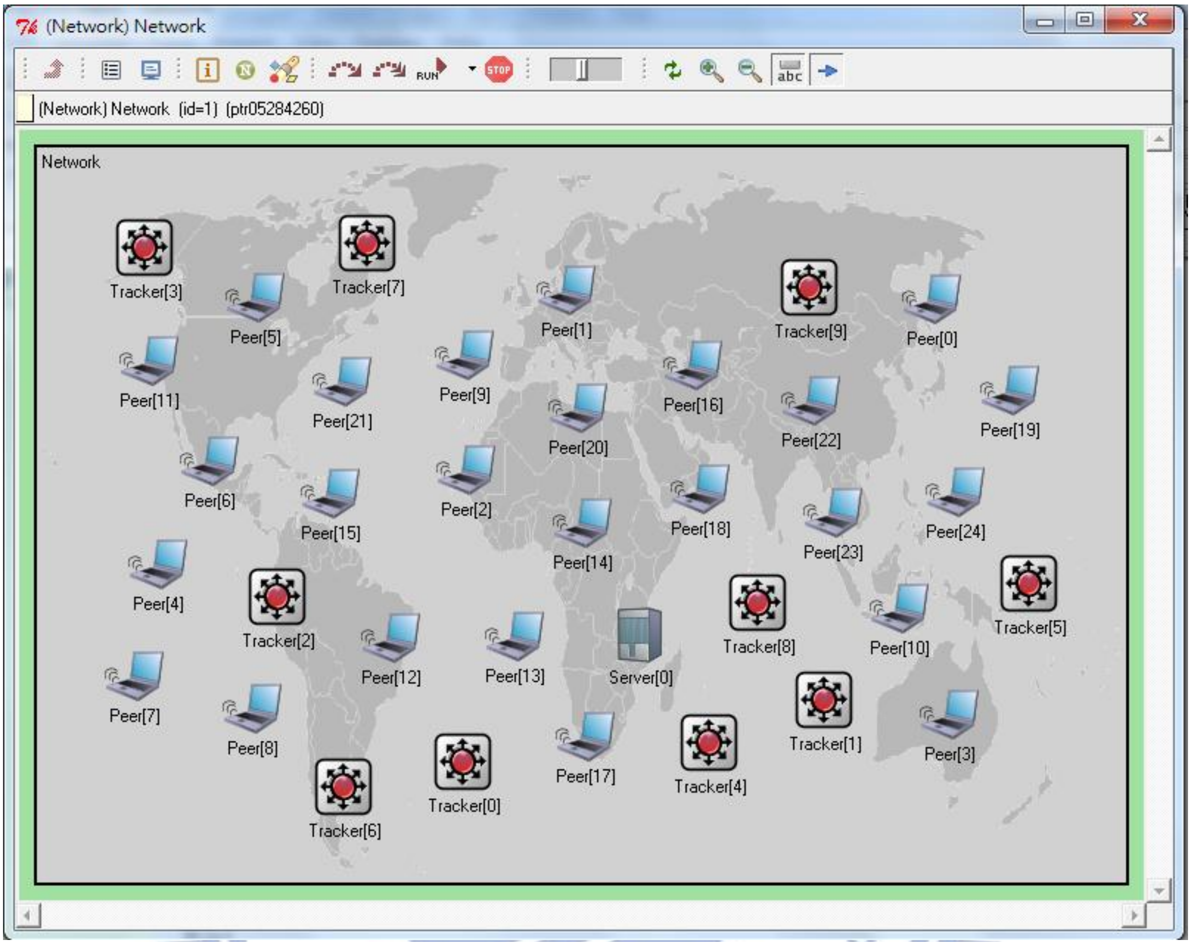


Figure 4.1 Network topology of the simulation.

4.1.1 Basic Components

In addition to the three basic components in our P2P live streaming network, there are five basic modules in peer's application: 1) Membership manager, which maintains partial view of the overlay. Every time Membership manager refreshes peer list, it will apply our extended ICMP service. Our scheme is appended into a new functionality in Membership manager. 2) Partnership manager, which establishes and maintains TCP/UDP connection, or partnership, with other peer. It also exchanges the availability of stream data in the buffer map (BM) with other peers. 3) Stream manager, which is the key component for data delivery. Besides providing stream data to the media player, it also makes decisions on where and how to retrieve the stream data. The central design in this system is based on the data-driven notion (focus on the status of the buffer), in which every peer node periodically exchanges its data availability information with a set of partners to retrieve unavailable data, while also supplying available data to others. 4) Buffer, which is the temporary storage of cached chunks before data is sent to the media player for playback. 5) Player, which is the module used to simulate the media player when peer watches the video.

4.2 Simulation Preconditions

The scenarios of our simulation are divided into 2 parts. The first scenario of our experiment applies to the real network. We develop a C program to record the response time of all of our destinations. The second scenario of our experiment is using the OMNet++ simulator to observe the response time from which the partner selection of P2P IPTV system can be benefited via our proposed approach.

4.2.1 Scenario 1

We compare our proposed scheme with three other current approaches listed in Table 4.1. In fact, the Xcast protocol itself is simple. All the Xcast router needs to do are extracting addresses, looking up the routing table, re-encapsulating and duplicating packets by the addresses. Table 4.2 shows the parameters used in our experiment.

Table 4.1 Four schemes in our experiment

Protocol	Server Selection method
Round-Robin	Successively pairing server
Anycast name resolvers	Final ICMP response server
Sequential unicast ICMP	First ICMP response server
Proposed scheme	First ICMP response server

Table 4.2 Simulation Variables for Scenario 1.

Parameters	Variables
The number of destinations	20000
The address space per IP header	4 bytes
The size of IP header	40 bytes
The size of ICMP header	8 bytes

In the following, we briefly describe the features of each protocol mentioned above.

- **Round-Robin**

This method is an arrangement of choosing all elements in a group equally in some rational order, usually from the top to the bottom of a list and then starting again at the top of the list and so on.

- **Anycast name resolvers**

The main concept of this approach is that the senders are waiting for all ICMP response from all destinations. It does not select the appropriate candidate until receiving the last reply message from all of destinations.Table 4.3 shows the difference of proposed scheme and anycast name resolvers.

Table 4.3 Difference of Proposed Scheme and Anycast name resolvers.

	Proposed Scheme	Anycast name resolvers
maintenance	real time	once in a while
decision	response time	distance, response time, load of server

- **Sequential unicast ICMP**

The approach is a sequential unicast to all the destinations in a group. It does not select the appropriate candidate until receiving the first reply message from all of destinations.

4.2.2 Precision Analysis for Anycast

When sequential unicast ICMP service sends echo message to all destinations, the delay between each message is inevitable. There is one situation that the response time of each echo message may not be as accurate as expected, because the response time can be affected by the delay between each sequential echo message. The shorter response time may be misidentified as the longer one due to the initial delay of this echo message. Therefore, in our experiment we propose that the precision for a class is the number of destination divided by the total number of elements labeled as belonging to the positive class (the number of items correctly labeled).

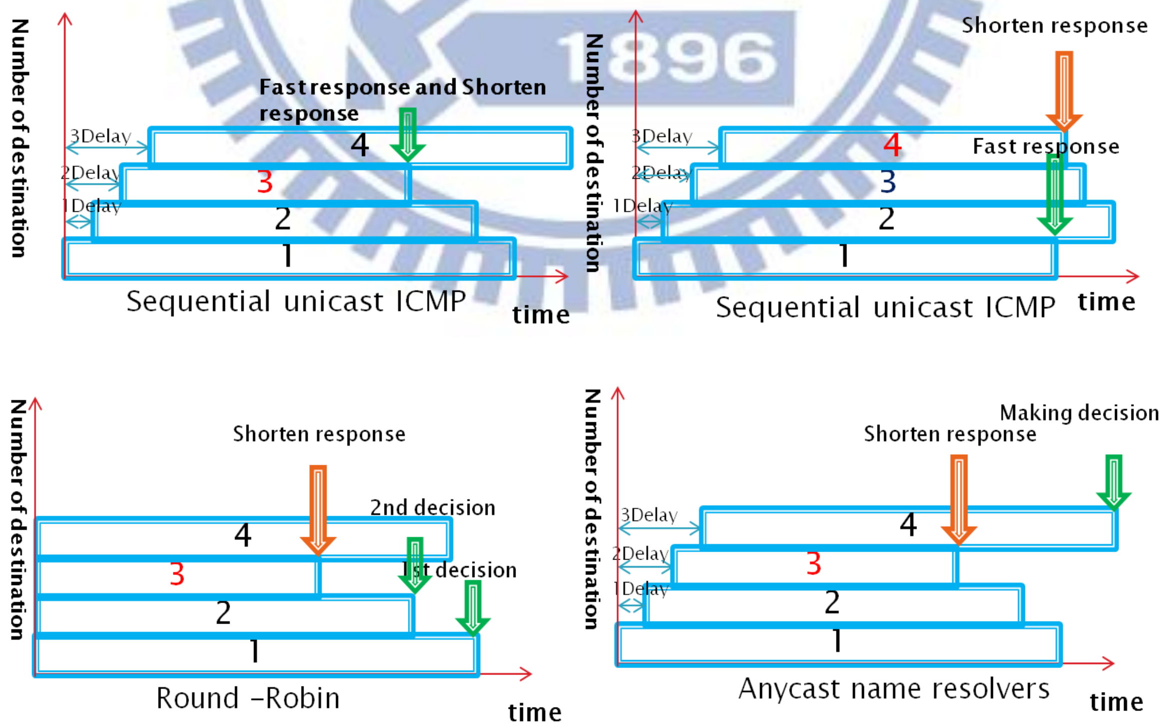


Figure 4.2 Precision of three traditional scheme.

4.2.3 Header Processing and Delay

The source in Xcast encodes the list of destinations in the Xcast header, and then sends the packet to a router. Each router along the path parses the header, partitions the destinations based on each destination's next hop, and forwards a packet with an appropriate Xcast header to each of the next hops. The Xcast packet header processing time in a router (Xhdrt) is approximately proportional to the number of entries in the list of destinations presented in the Xcast packet. It could be defined by the following equation:

Xhdrt= Nl X Uhdrt

Where Nl is the number of leaf node routers on the tree (number of entries in the list of destinations) and Uhdrt is the header processing time needed for a unicast packet. Using the Xcast protocol, only branch messages need extra header processing time. The packet header processing (and thus delay) in Xcast is minimized [19].

4.2.4 Scenario 2

Table 4.4 Simulation Variables for Scenario 2.

Parameters	Variables
The number of peer	20000
The number of tracker	10
The number of partners	5
The number of channels	50
The number of peer in the peer list	1~20

In the second simulation, we want to evaluate the performance of our approaches in P2P IPTV service. The simulation environment is based on OMNeT++ simulator, the parameter setting is depicted in Table 4.4. In our simulation, there are 50 channels, 10 trackers, and the number of partners of each peer is 5 (the same as channel number), plus 1~20 peers in the peer list. Additionally, our scenario has 20000 nodes connected in mesh topology with random link bandwidth between 400 kbps and 1 Mbps.

4.3 Simulation Results for Scenario 1

4.3.1 Result for Improvement of Anycast

In this section, we present the results based on the scenario introduced in previous section. In this simulation, we want to ensure that our scheme works well for the anycast service by Explicit Multicast. Our scheme is designed for large group, but it is still suitable and harmless for small group. It can reduce a lot of traffic volume when a group has many Xcast addresses. We will compare our scheme with sequential unicast ICMP, anycast name resolvers and round robin in real networks.

We assign a sequence number to each anycast group for observing the first response time. Figure 4.3 shows each response time (ms) in four different approaches. We can see that the anycast name resolvers generates the largest response time. The reason is that the source has to wait for the last response from all destinations and then decide the fastest (the smallest value) one among all the destinations. Our proposed approach (Extended ICMP service using Xcast) generates the shortest response time (ms).

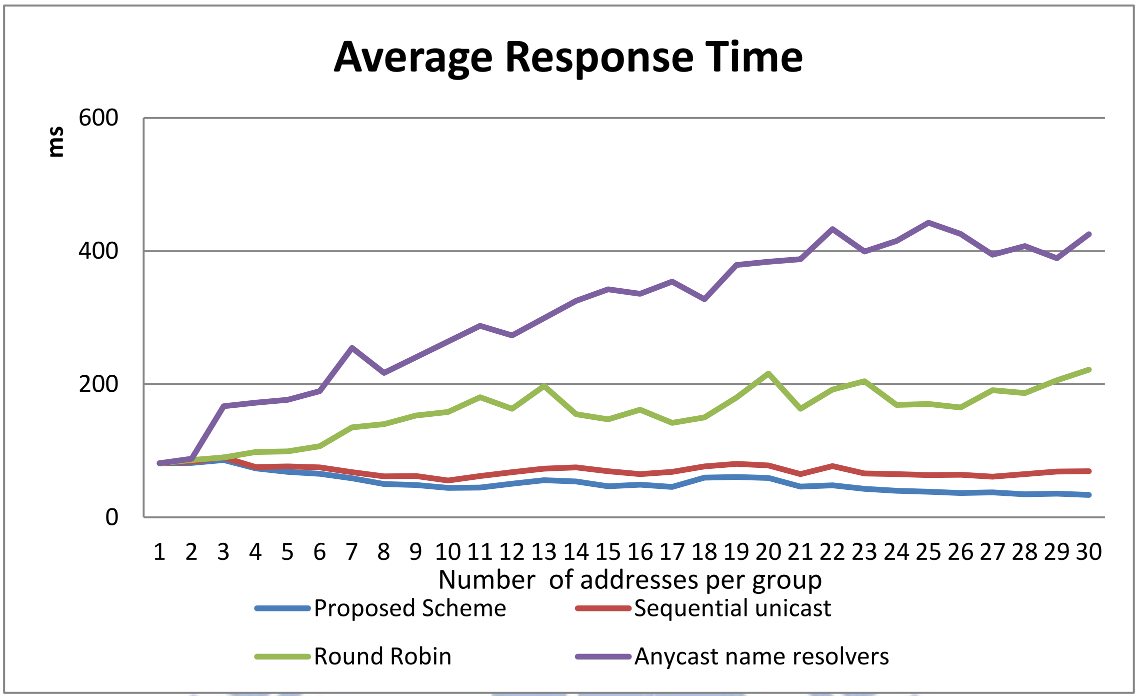


Figure 4.3 Response time in four schemes with different number of destinations in group.

Because the response time can be obviously noted that the performance of our approach is far better than those of Round Robin and Anycast name resolver in Figure 4.3. In Figure 4.4, we demonstrate the simulation results of our proposed approach and the scheme of sequential unicast only. For each message sent from the source, the scheme of sequential unicast incurs delay. Therefore, with the number of group increases, the performance of our proposed approach (the decision of fastest response time from all destinations) performs better and becomes more obvious especially in larger group.

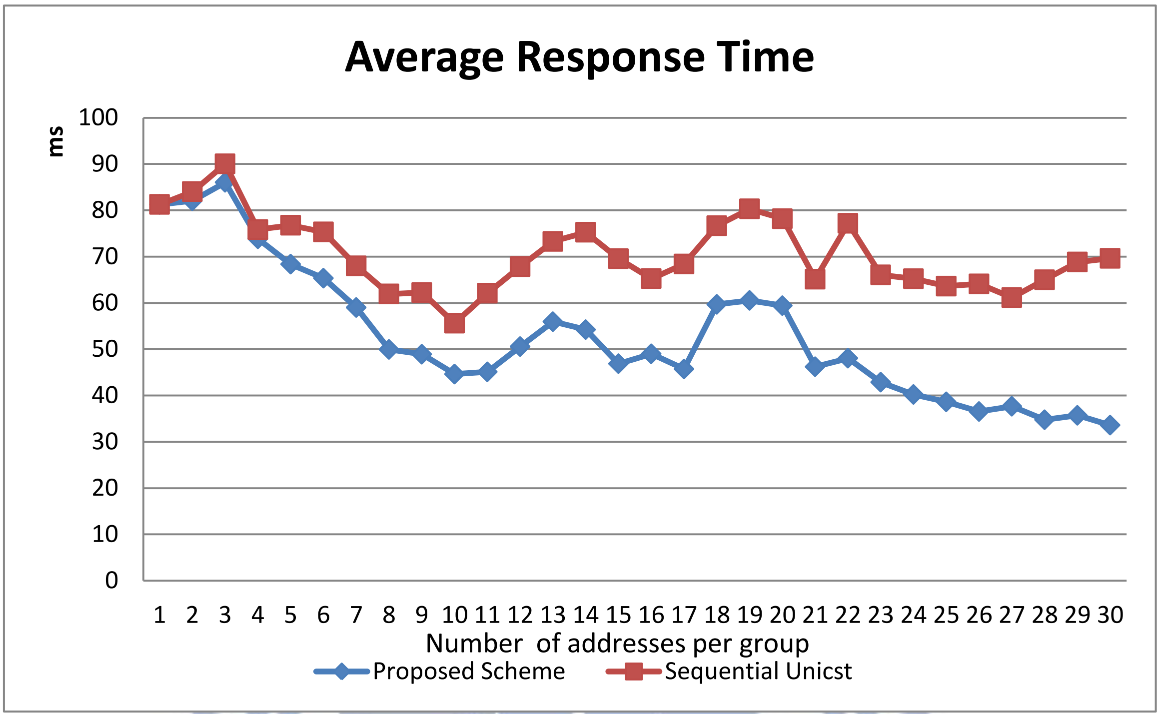


Figure 4.4 The first response in two schemes with different number of destinations in group.

Figure 4.5 shows the total traffic volume (in bytes) in two different approaches. We can see that the scheme of unicast generates larger traffic volume (in bytes) than that of our approach. Our scheme of Extended ICMP service using Xcast generates smaller traffic volume. The curve of total traffic volume (in bytes) is similar to that of total hop count, which is resulted from the total traffic volume that is the sum of each hop * data volume.

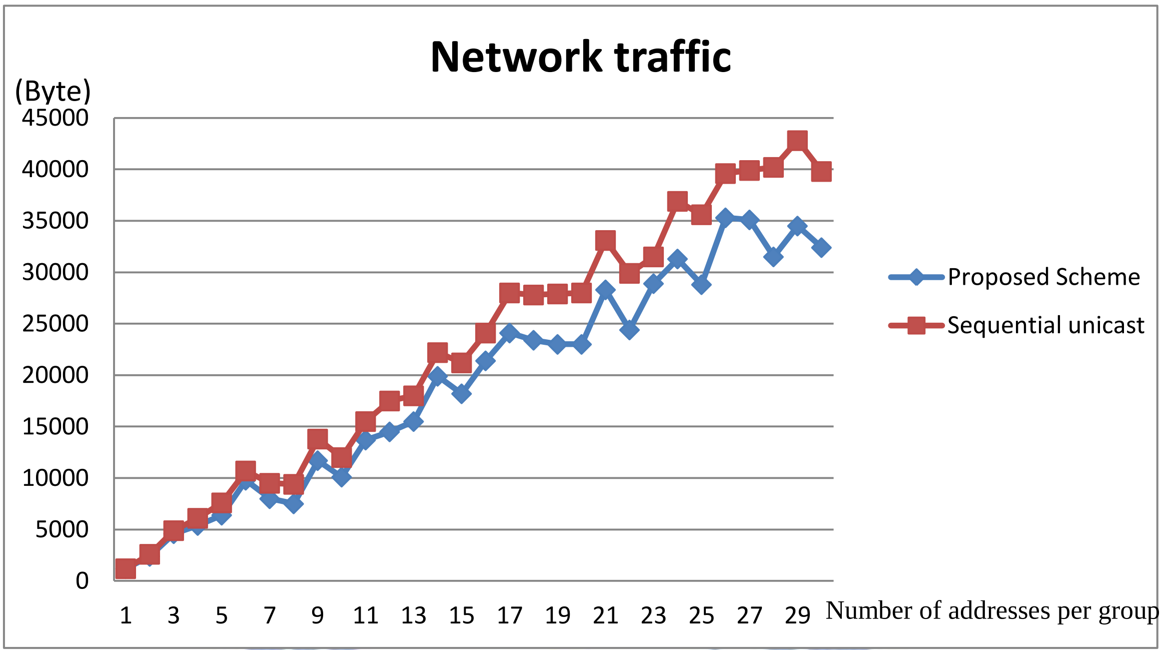


Figure 4.5 Traffic volumes in two schemes with different number of group.

Figure 4.6 shows the relation between the precision and different number of addresses per group of two schemes. With the increasing of number of addresses per group, the delay is becoming larger. Therefore, the precision of determining the shortest response time will be affected by the number of addresses per group.

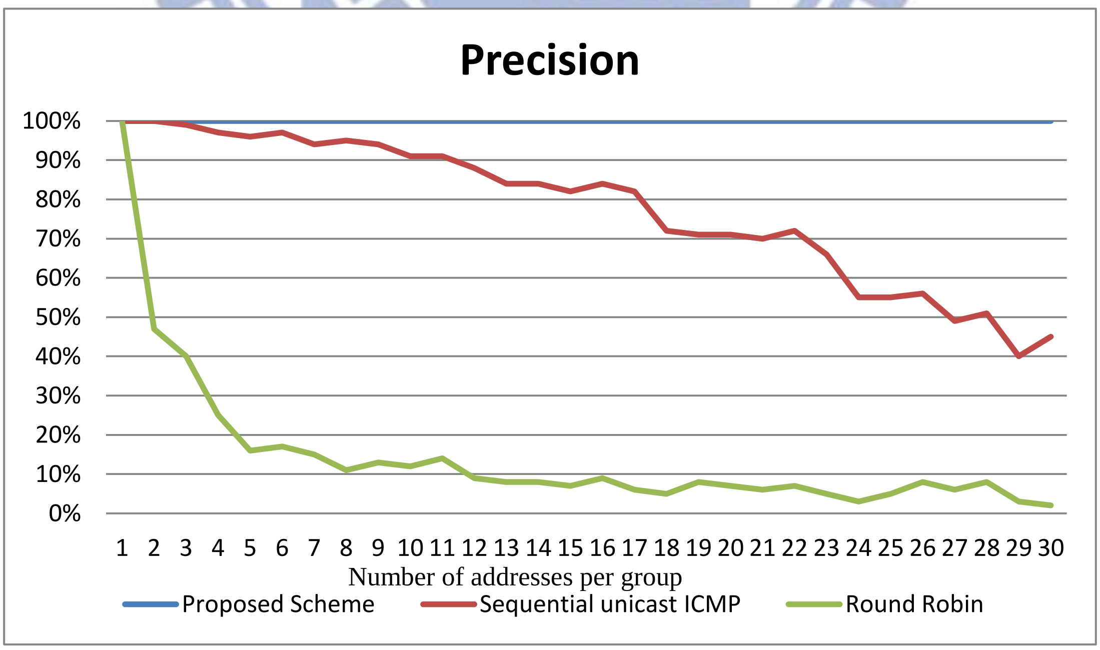


Figure 4.6 Precision Comparison of two schemes.

4.3.2 Result for Improvement of Manycast

In order to ensure that the manycast works in real network, our experiment is designed such that one client communicates simultaneously with 1~30 of 5 equivalent servers in a group. Then, we demonstrate the results as illustrated in Figure 4.7.

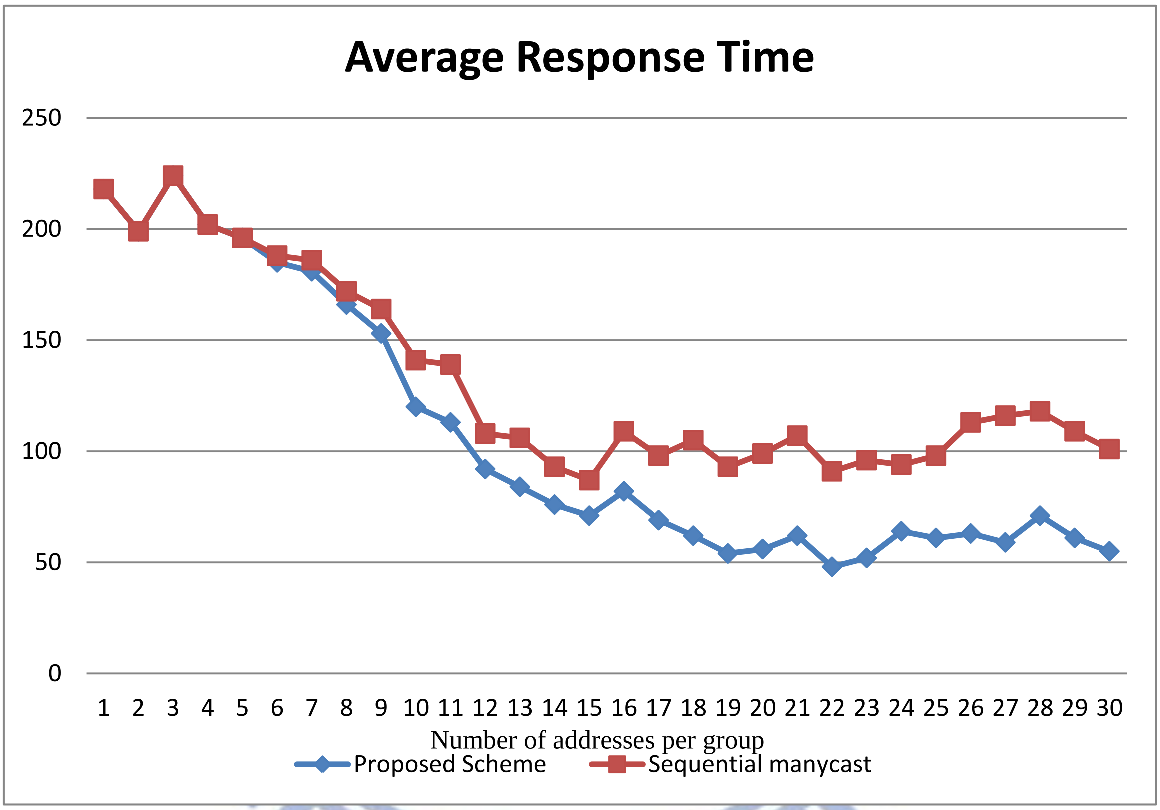


Figure 4.7 Response time of two schemes with different number of groups.

Figure 4.7 shows the response time (ms) normalized by the results of unicast. We can see that the results of sequential manycast and proposed scheme are decreasing when there are more destinations in the system. This is because the more destinations are in the system, the higher possibility we can determine the fastest response from serving server. We measure the fifth response time as source receives by the sequence numbers in each group. Moreover, we measure the overall network traffic of unicast and our proposed approach.

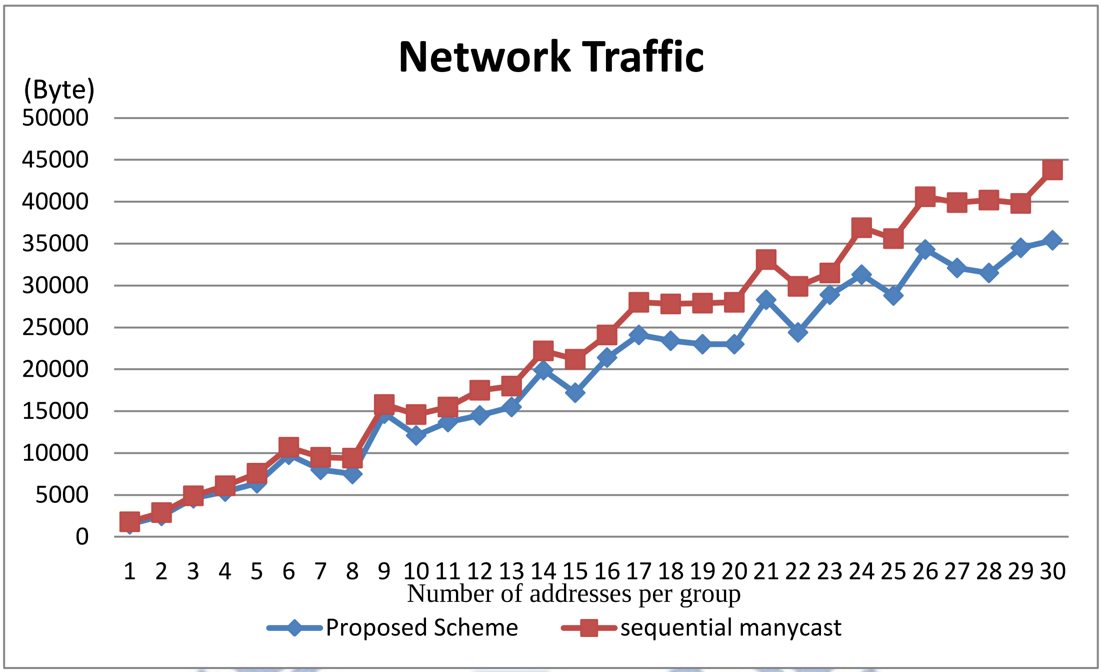


Figure 4.8 Overall network traffic transmits fixable packets.

Figure 4.8 shows the improvement of overall network traffic between our approach and unicast. The network traffic of sequential unicast and our proposed scheme are not increasing monotonically when there are more destinations in the system. This is because when there are more destinations, the probability of finding the similar routes is getting higher and we can take the benefits from these similar routes without additional routing procedure. In addition, with the increase of the number of addresses in each group, unicast shows linear growth in network traffic and our scheme will clearly reducing the network traffic.

4.4 Simulation Results for Scenario 2

4.4.1 Simulation Result for the Improvement of Tracker and Partner Selection in P2P IPTV System

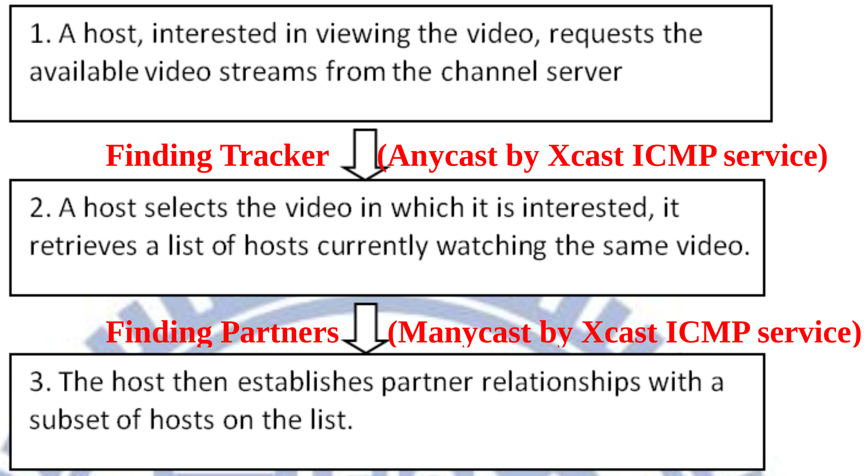


Figure 4.9 IPTV System Flow.

This section presents the simulation results of the proposed fast partner selection by using anycast and manycast of extended ICMP service by Xcast. Our proposed approaches are applied to the parts of finding tracker and finding partners respectively as shown in Figure 4.9.

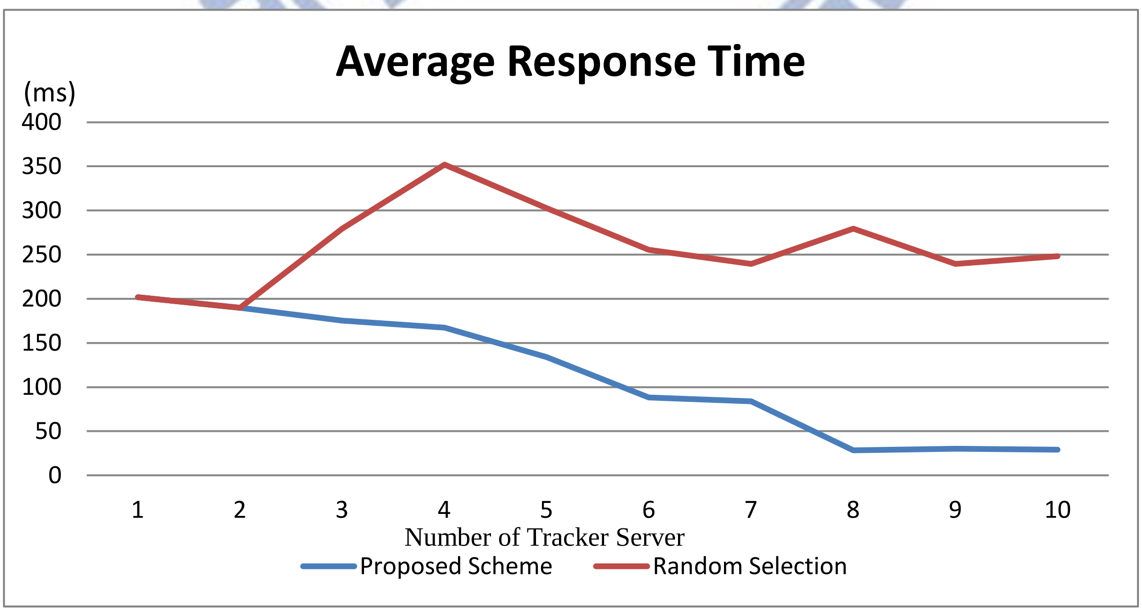


Figure 4.10 Comparison of end node Tracker Server selection delay.

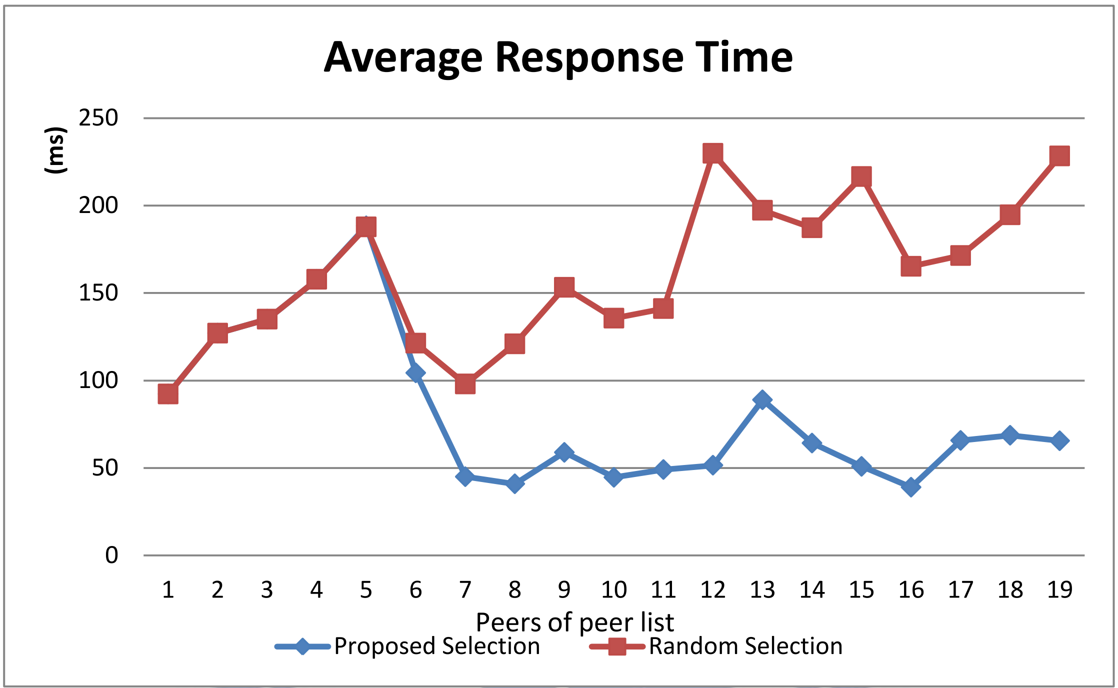


Figure 4.11 Comparison of end node partners selection delay.

We compare the proposed approaches of finding tracker using Anycast by Xcast ICMP service in Figure 4.10 and finding partners using Manycast by Xcast ICMP service in Figure 4.11 with the traditional random selection in both finding tracker and partners. There is one problem we want to address in Figure 4.10 and Figure 4.11. With the increase of the number of tracker or the number of peers in the peer list, the response time of traditional random selection scheme may not increase. The circumstance is that the peers or trackers with longer response time exist in the smaller group. Therefore, the average response time in smaller group may be longer than that in larger group.

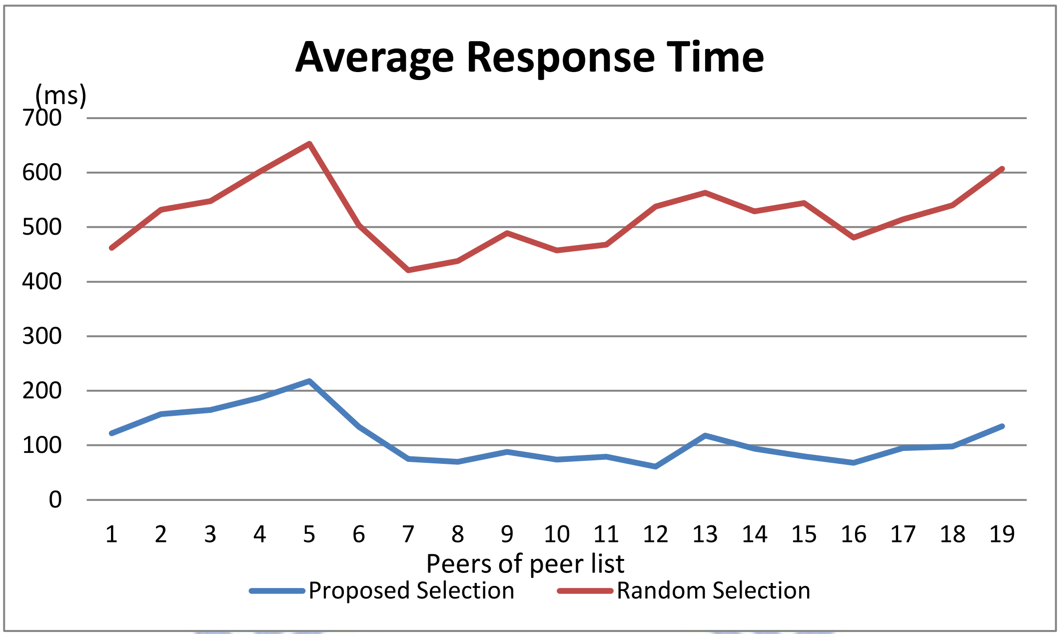


Figure 4.12 Comparison of end node tracker and partners selection delay.

Figure 4.12 shows the average response time after each peer determining its partners via the procedure of finding tracker and finding partners between randomly selection and our proposed approaches. The result shows that our approaches apply to the partner selection can quickly form an overlay network. In some situation, the startup delay may be reduced due to the quickly-formed overlay network. Because we can discover the real and good neighbors that can provide immediate streaming service to the requesting peer.

Chapter 5 Conclusion & Future Work

In this thesis, we propose a new scheme for the improvement of anycast and manycast protocols on the Internet. We discuss the characteristics of anycast addresses and several applications of anycasting. We particularly point out there were no existed routing protocols to handle anycast addresses.

We collect 20,000 routes of 20,000 different IPs to construct the topology for experiment. We design a simulator for our scheme and set up an experiment to observe the performance of our scheme, and the results are as expected.

To evaluate the efficiency of the proposed scheme, we use the existing Xcast header and add it into ICMP header. We implement the anycast and manycast features, apply our anycast selection rule and manycast selection rule to our simulator. The result shows that our scheme is able to reduce the number of query messages, network traffic, speeding up the response from destinations, and increasing precision successfully. Then, we apply our proposed scheme in P2P IPTV system to obtain closest partners with peer. However, our proposed scheme still has some defective situations that we can improve in the future, such as no mechanism to choose the best service providers equipped with good bandwidth or fast transmission.

In future work, we plan to evaluate new schemes for more complex header processing, and choose the best service providers that are with good bandwidth and fast transmission.

References

[1] J. Postel, "Internet Protocol," RFC 791, Sep. 1981.

[2] S. Deering, and R. Hinden, "RFC: 1883 Internet Protocol Version 6 (IPv6) Specification," Internet Engineering Task Force, Dec. 1995.

[3] S. Deering, "Host Extensions for IP Multicasting," RFC 1112, Aug. 1989, http://www.ietf.org/rfc/rfc1112.txt.

[4] C. Partridge, T. Mendez, and W. Milliken, "Host Anycasting Service," IETF RFC 1546, Nov. 1993.

[5] R. Hinden and S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture," IETF RFC 1546, Apr. 2003.

[6] R. Boivie, N. Feldman, Y. Imai, W. Livens, D. Ooms, and O. Paridaens, "Explicit multicast (Xcast) basic specification," IETF, Draft, June. 2004, draft-ooms-xcast-basic-spec-06.txt.

[7] S. Sarat, V. Pappas and A. Terzis, "On the Use of Anycast in DNS," ICCCN, pp. 71-78, Arlington, VA, Oct. 2006.

[8] C. Carter, S. Yi, and R. Kravets, "Manycast: exploring the space between anycast and multicast in ad hoc network," MOBICOM, pp. 273-285, Feb. 2003.

[9] D. Mills, "Network Time Protocol (Version 3) Specification, Implementation and Analysis," IETF RFC1305, Mar. 1992.

[10] Hei X., Liu Y., et al."IPTV over P2P streaming networks: The mesh-pull approach," IEEE Communications Magazine, Vol. 46 , No. 2, pp. 86-92, Feb. 2008.

[11] S. Doi, S. Ata, H. Kitamura, and M. Murata," IPv6 anycast for simple and effective service-oriented communications," IEEE Communications Magazine, Vol. 42, No. 5, pp. 163-171, May. 2004.

[12] S. Weber, L. Cheng, "A survey of anycast in IPv6 networks," IEEE Communications Magazine, Vol. 42, No. 1, pp. 127-132, Jan. 2004.

[13] D. Johnson, C. Perkins, J. Arkko, "Mobility Support in IPv6," RFC3775, June 2004.

[14] M. OE and S. Yamagugi, "Implementation and Evaluation of IPv6 Anycast," Proceeding of 10th Annual Internet Society Conference, Jul. 2000.

[15] J. Postel, "Internet Control Message Protocol," in RFC 792, Sep. 1981.

[16] E-D. Katz, M. Butler, R. McGrath, "A Scalable HTTP Server: The NCSA Prototype," Computer Networks and ISDN Systems, Vol. 27, No. 2, pp.155-164, Nov. 1994.

[17] E. W. Zegura, M. H. Ammar, Z. Fei, and S. Bhattacharjee, "Application-Layer Anycasting: a Server Selection Architecture and Use in a Replicated Web Service," IEEE/ACM Transactions on Networking, Vol. 8, No.4, pp. 455-466, Aug. 2000.

[18] X. Hei, C. Liang, J. Liang, Y. Liuand and K-W. Ross, "A Measurement Study of a Large-Scale P2P IPTV System," Multimedia IEEE Transactions, Vol. 9, No.8, pp.

1672-1687 , Dec. 2007.
[19] A. Boudani and B. Cousin, “SEM: A New Small Group Multicast Routing Protocol,” in Proc. the 10th International Conference on Telecommunications (ICT2003), Tahiti, Feb. 2003.