社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

IPv6 ネットワークにおける エニーキャストルーティングプロトコルの設計

土居　聡† 阿多　信吾†† 北村　浩††† 村田　正幸†††† 宮原　秀夫†

† 大阪大学 大学院情報科学研究科　〒 560–8531 大阪府豊中市待兼山町 1–3
†† 大阪市立大学 大学院工学研究科　〒 558–8585 大阪府住吉区杉本 3–3–138
††† NEC ネットワーク開発研究所　〒 108–8557 東京都港区芝浦 2–11–5
†††† 大阪大学 サイバーメディアセンター　〒 560–0043 大阪府豊中市待兼山町 1–30
E-mail: †{s-doi,miyahara}@ist.osaka-u.ac.jp, ††ata@info.eng.osaka-cu.ac.jp, †††kitamura@da.jp.nec.com, ††††murata@cmc.osaka-u.ac.jp

あらまし　IPv6 の持つ新しい機能の 1 つとして, エニーキャスト通信機能がある. エニーキャスト通信を使えば, 複数のサーバの中から最適なサーバと自動的に通信可能となる. しかし, この最適なサーバ選択を実現するには, 新たなルーティングプロトコルのサポートが必要となるため, 現状では利用できない. そこで本稿では, エニーキャスト通信とマルチキャスト通信との類似性を元に, 既存のインターネットへの適用性を考慮した新たなエニーキャストルーティングプロトコルを提案する.

キーワード　IPv6, エニーキャストルーティング, エニーキャストメンバーシップ

A Protocol Design for the Anycast Communication in the IPv6 Network

Satoshi DOI†, Shingo ATA††, Hiroshi KITAMURA†††, Masayuki MURATA††††, and Hideo MIYAHARA†

† Graduate School of Information Science and Technology, Osaka University
1–3 Machikaneyama, Toyonaka, Osaka, 560–8531, Japan
†† Graduate School of Engineering, Osaka City University
3–3–138 Sugimoto, Sumiyoshi–ku, Osaka 558–8585, Japan
††† Development Laboratories, NEC Corporation　2–11–5 Shibahara, Minato–ku, Tokyo 108–8557, Japan
†††† Cybermedia Center, Osaka University　1–30 Machikaneyama, Toyonaka, Osaka 560–0043, Japan
E-mail: †{s-doi,miyahara}@ist.osaka-u.ac.jp, ††ata@info.eng.osaka-cu.ac.jp, †††kitamura@da.jp.nec.com, ††††murata@cmc.osaka-u.ac.jp

Abstract　Anycast if defined as one of new IPv6 features. Using anycast communication make it possible that clients automatically communicate to the “appropriate” server among the candidate servers. However, the best server selection is not impossible since existing routing protocol cannot deal with anycast communication. In this paper, we design a new routing protocol for anycast communication, based on the analogies between anycast and multicast. Our proposed method is applicable to existing Internet.

Key words　IPv6, Anycast Routing, Anycast Membership

1. Introduction

1.1 Overview of Anycast Communications

Anycast [1] is one of the new IPv6 (IP version 6 [2]) features that supports service-oriented address assignments in IPv6 networks. An anycast address is not determined by the location of the node, but by the type of service offered at the node. In anycast communications, the client can automatically obtain the appropriate node corresponding to a specific service without knowledge of the location of the server.

According to the protocol specification of IP Version 6 [1], there are three types of IP address; 1) unicast, 2) multicast, and 3) anycast. The communication forms of these addresses are summarized in Table 1.

A unicast address is a unique identifier for each network interface, and multiple interfaces must not be assigned the same unicast address. Packets with the same destination address are sent to the same node. A multicast address, on the other hand, is assigned to a group of nodes, i.e., all members of the group have the same multicast address. Packets for the multicast address are sent to all multicast members simultaneously.

Like a multicast address, a single anycast address can be assigned

Table 1 IPv6 address types

comparison item	unicast	multicast	anycast
communication form	point to point	point to multipoint	point to point
target of address	single	multiple	type of service
membership (role: client/server)	single (both)	multiple (client)	multiple group (server)

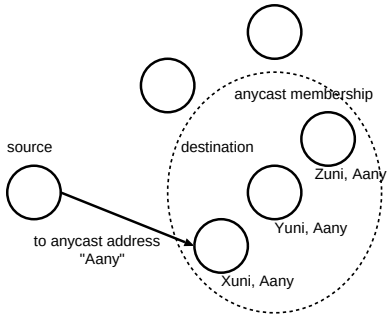

Figure 1 Anycast communication

to multiple nodes (called *anycast membership*). However, unlike multicast, only one anycast member can communicate with the assigned anycast address at any one time.

Figure 1 shows an example of an anycast communication. In Figure 1, there are three nodes associated with the anycast address A_{any}. When the node sends a packet whose destination address is A_{any}, the packet is sent to one of three nodes (X_{uni} in this figure), that is, not to all nodes. The merit of anycast is that the receiver node may vary according to the node and/or network condition. In this case, if the node X_{uni} is down, the packet for A_{any} can be sent to another node (Y_{uni} or Z_{uni}) by updating the routing information. The main idea behind anycast communication can be found in the separation of the logical service identifier from the physical node equipment. The anycast address is assigned on a type-of-service basis and enables a service to act as a *logical node* appropriate for the service. The following applications are subject to the use of anycasting.

（1） Dynamic Node Selection

By performing the routing control appropriately, the sender node can communicate with the optimal node (chosen from multiple anycast nodes) by simply specifying the anycast address. For example, if we assign the same anycast address to the WWW server and its mirror sites, end users can access the site nearest to their location.

（2） Well-known Anycast Address for Specific Services

By defining and assigning the Well-known Anycast Address to widely used applications (e.g., domain name services, proxy services, etc.), the user can use these services without setting the address of the server.

（3） Active Load Balancing using Multiple Servers

Load balancing among the multiple servers can be performed by giving the same anycast address to the servers. If routing information is updated by the load, active load balancing is realizable.

Anycast communication has the interesting features noted above; however, the current use of anycast addresses is quite limited. One of the main reason is that there are many points in the current definition of anycasting that are still unclear. Moreover, anycast communication has problems with its protocol specifications and its routing mechanism. Especially, providing a routing mechanism for anycast address is a key requirement for the wide use of anycast communication. However, currently no protocol standard or consensus has arisen for the routing mechanism.

1.2 Problems of Anycast Routing

There are several reasons why the routing architecture of anycast communication is still an open issue.

（1） *Node Selection Criteria*

The most important problem for anycast communication is how to transfer a packet with a specific anycast address to an *appropriate* node. Throughout this paper, we call it as *Anycast Routing*. The meaning of *appropriate* differs by the kind of applications, e.g., if the application requires a faster response, the propagation delay is most important. That is, the nearest node among the anycast membership should be chosen. On the other hand, if the application performs a complex calculation, the node having more CPU resources should be selected. For this reason, the protocol of anycast routing should handle various kinds of metrics specified by the applications. The criteria for anycast routing strongly affects the capability of anycast communication.

（2） *Scalability of Routing Protocols*

Basically, an IP (Internet Protocol) router has a routing table to decide the output interface of the arrived packet. That is, the router forwards the arrived packet according to its destination address by searching the output interface from the routing table. To reduce the size of the routing table, the router aggregates multiple entries which have the same prefix of destination address (or network), and the same output interface (i.e., the same direction). However, the routing entry for the anycast address cannot be aggregated because the locations of anycast membership are widely distributed regardless from their prefix. Hence, routing entries for anycast address should be held on the router individually. When the anycast address becomes widely used, it is easy to imagine an explosion of the routing table.

（3） *Discovery of Anycast Membership*

Maintaining the anycast membership is also an important issue. The easy way to construct an anycast membership is that the node intended to join the anycast membership simply advertises the routing entry for the associated anycast address to the router. However, such approach can sometimes lead to a serious security problem. That anycast can add/delete the anycast entry of the routing table might be harmful. For example, an anycast server added by a malicious user may cause packets to be blackholed. In this way, the anycast server added/deleted affects other servers having the same anycast address.

（4） *Packet Reachability*

Any node in the Internet should be able to communicate with at least one node of the membership associated with the anycast address (if the node exists). But poor coordination between routers, may cause a packet sent to an anycast address to be unable to arrive anywhere.

1.3 Research Goals

Keeping the above-mentioned problems in mind, we propose a new routing scheme for anycast communications. The main motivation of our project is to make anycast addresses more useful without (or with a minimum of) any application modification and/or proto-

col extension. Unlike other routing mechanism for anycast communication [3], our proposed routing architecture has the following advantages:

（1） *Support of Gradually Shifting*

Because there are enormous numbers of routers in the Internet, it is unfeasible scenario that all routers handle anycast addresses. Our proposed architecture can work correctly even if just one router support the anycast address (called *Anycast Router*). Of course, the impact of anycast routing will increase when more anycast routers are deployed.

（2） *No Extention of Protocol Specification*

Our proposed scheme does not require any modification and/or extension of current anycast protocol specifications.

（3） *Reachability Guaranteed*

In the proposed architecture, any node in the Internet can communicate with at least one node of the membership associated with the anycast address.

（4） *Implementability*

To make implementing the anycast routing mechanism easier, we chose an approach to modify the current existing protocols for multicast communication. As shown in Table 1, anycast and multicast have some similar properties. By considering the differences between multicast and anycast, we modify some existing multicast protocols to support anycast communications.

This paper comprises five sections. In Section 2., we show our proposed anycast routing architecture. In Section 3., we describe the implementation issues of our architecture. And we compare our proposed protocols in Section 4. Finally, we provide a brief summary with future research topics in Section 5..

2. Anycast Routing Architecture

The advantage of anycast communication is that the packet is automatically forwarded to the appropriate node according to network and/or node conditions. Thus, maintaining the routing information of anycast addresses is an important task. For this reason, we propose a new anycast routing architecture. Our proposed routing architecture is first described in this section.

2.1 Basic Concept

To solve the problems mentioned in Section 1., our routing architecture has the following features.

（1） Choosing Anycast Address from Existing Unicast Address

According to the IPv6 specification [1], anycast and unicast addresses share the same addressing space. That is, the two addresses are not syntactically distinguishable. This has both strengths and weaknesses: For an anycast router, it is difficult to decide whether the destination address of the arrived packet is anycast or unicast. However for a unicast router (i.e., a current router which cannot handle the anycast address), the router can simply forward the packet without any special operations even if the destination address is anycast. We take advantage of the latter's nature to guarantee the reachability. More specifically, we choose a *default node* from anycast membership before assigning an anycast address. We then set the anycast address of the membership to be the unicast address of the *default node*. When the anycast router receives the packet destined for the anycast address (We call the packet as *anycast packet*), the anycast router sends the anycast packet to the *appropriate* node among the anycast membership. Otherwise, the unicast router only tries to forward the anycast packet to the *default node*. The anycast packet departed from an arbitrary node is sent at least to the default node. Reachability is thus guaranteed.

（2） Gradually Transient Model by Increasing the Number of Anycast Routers

In our architecture, the merit of anycast routing can be enjoyed only if the route between the sender and the default node has an anycast router. Consequently, if we want to improve the effect of anycast routing, we need to deploy additional anycast routers. The effectiveness is proportional to the number of anycast routers.

（3） Modify the Existing Multicast Routing Protocols to Support Anycast

To reduce the complexity of implementation, we adopt an approach to that modifies the multicast routing protocols to support anycast routing, since there are many similar points between anycast and multicast. For example, membership management and routing table construction procedures are a common feature between anycast and multicast . The modifications are based on the differences between anycast and multicast.

（4） Introducing *Scope* to Keep the Scalability

To control the routing table size and keep the scalability, we introduce a scope that limits the range of transmitting anycast data/control packets.

2.2 Proposed Architecture

Figure 2 shows the overview of our proposed routing architecture. In the architecture, there are two types of routing topologies. *Unicast network* is the current existing network topology in which both unicast and anycast packets are forwarded on the basis of unicast address. *Anycast network* is an overlaied logical topology, in which anycast-aware routers (called *anycast router*) are connected to each other and only anycast packets are forwarded by treating the address of a packet as anycast.

In an anycast network, since they are not physically (i.e., directly) connected, they are connected via various kinds of logical peer-to-peer connections (e.g., virtual path, tunneling, or encupseling, ...). An anycast router is upper-compatible and can perform anycast routing functions in addition to the capabilities of unicast routers. An anycast router has an extra routing table (called *anycast routing table*) to handle the anycast address. An anycast routing table consists at least (*anycast address*, *next anycast router's address*) pairs. When the packet has arrived at the anycast router, the anycast router first checks the anycast routing table to find the entry regarding the destination address of the packet. If the address is found in the anycast routing table, the packet is treated as *anycast packet* and forwarded to the next anycast router according to the anycast routing table. Otherwise, the packet is forwarded by using the unicast routing mechanism.

An example of anycast routing is shown in Figure 2. In this figure, we assume that the node selection criteria is the number of hops. Namely, a smaller hop count is more appropriate in this case. In Figure 2, the solid line square is denoted as a router, and a square with "AR" is an anycast router. A blank square stands for a unicast router. There are two anycast members for the anycast address `3ffe:5::5`. Note here that `3ffe:5::5` is also the unicast address of anycast server A1. In this case, the node A1 is the *default node* of the anycast membership for `3ffe:5::5`. The other node

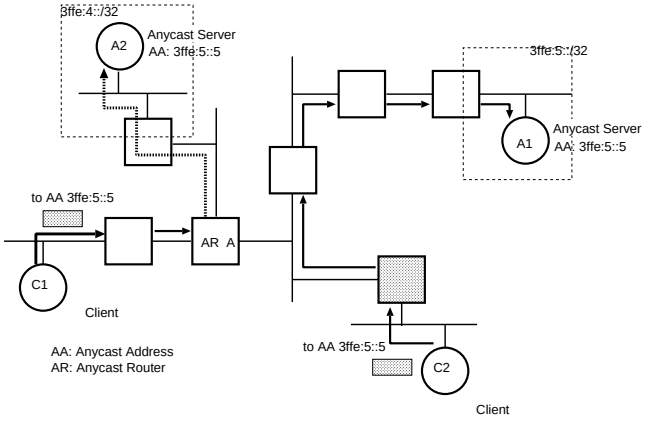

Figure 2 Proposed Architecture

A2 is in a different network (`3ffe:4::/32`). We now consider cases where two nodes (C1 and C2) send the packet destinied for the anycast address `3ffe:5::5`. The difference between the two cases is whether an anycast router exists on the route to the default node A1. In case of C1, the packet is first forwarded to the router A1 by using unicast routing (denoted by solid arrow). The intermediate router A1 is an anycast router and it can detect the packet is an anycast packet. According to the anycast routing (denoted by dashed arrow), the anycast router R1 then forwards to node A2, which is the nearer node from C1. On the other hand, in case of C2, since there is no anycast router between C2 and A1, the packet is simply forwarded to A1 using the unicast routing only. Note that a more appropriate node (A2) exists in this network. For example, if we replace the router next to C2 (expressed by the gray filled box) to an anycast router, the packet could be transmitted to the more appropriate node A2 by using anycast routing.

Thus, our design proposal can operate, even when there are a small number of anycast routers. Moreover, if the anycast router numbers increase, better routing may be achieved. Finally, when all routers in the network become anycast routers, flexible anycast routing which adopts a policy control using various metric will become possible.

3. Routing Protocol Specifications

In this section, we describe the routing protocols for our proposed anycast routing architecture. As described before, there are many similar characteristics between anycast and multicast, that is, many functions of the routing protocol for multicast can also be applied to the one for anycast. However, there are some, but important differences between anycast and multicast. For example, in anycast routing, packets for an anycast address are required to be transmitted to only one of anycast members. In multicast routing, on the other hand, packets for a multicast address must be transmitted to all multicast members. In this paper, we especially focus on the such differences to modify the current existing routing protocols for multicast. We consider that the easy of implementation is also important to spread the use of anycast. Because anycast routing inherently has multiple paths toward the destination, exchanging and updating routing information technique used in multicast routing protocols are applicable to the anycast routing protocol. There are several protocols for unicast or multicast routing available now. As shown in Table 2, they are classified into three types: (1) distance vector, (2) link state, and (3) core based tree. In the distance vector algorithm, a router has a list of routers which are directly connected to. By exchanging the list with other adjacent routers, the router can know all routers to an arbitrary destination. While the link state algorithm utilizes a list of connected links instead of a list of routers. By exchanging the list of links, the router can know the whole topology of the network. The router then makes a shortest path tree (SPT) by using Dijkstra's shortest path first [4] algorithm. Based on the SPT, the router finally constructs the routing table. The core based tree is a kind of hierarchical algorithm. It first chooses one or more *core* routers from all routers. On behalf of other routers, the *core* router centralizes all routing information. The other router only holds the routing information to where the router is belonged to. Each router only sends a packet to the *core* router. Only the *core* router can decide the route for the destination address.

Table 2 Classification of Routing Protocols

	Distance-Vector	Link-State	Core-Based-Tree
Unicast	RIP [5]	OSPF [6]	
Multicast	DVMRP [7]	MOSPF [8]	PIM-SM [9]
Anycast	DVARP	AOSPF	PIA-SM

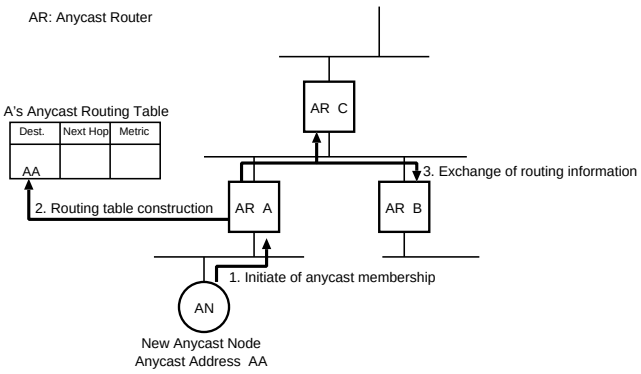

Figure 3 Overview of Anycast Routing Protocol

Since each of above algorithms has both advantages and disadvantages, we first define three anycast routing protocols derivered from above types. There are (1) Distance Vector Anycast Routing Protocol (**DVARP**), (2) Anycast extension of OSPF (**AOSPF**), and (3) Protocol Independent Anycast Sparse Mode (**PIA-SM**). We then compare among these protocols and show the guideline for choosing protocols in the next section.

A routing protocol for anycast communication functionally consists of following two processes:

（1） *Initiate of anycast membership*

The anycast router corrects the information of nodes which are intended to join some anycast memberships.

（2） *Constructing and updating routing table*

According to the information corrected in Step.1, the anycast router then constructs its own routing table. After that, anycast routers exchange the routing information each other, and reconfigure their own routing tables.

Figure 3 shows the overview of our anycast routing protocol.

In what follows, we describe detail of above steps separately.

3.1 Initiate of Anycast Membership

Like multicast, a host intending to participate to (or leave from) the anycast membership must have a capability to notify the sta-

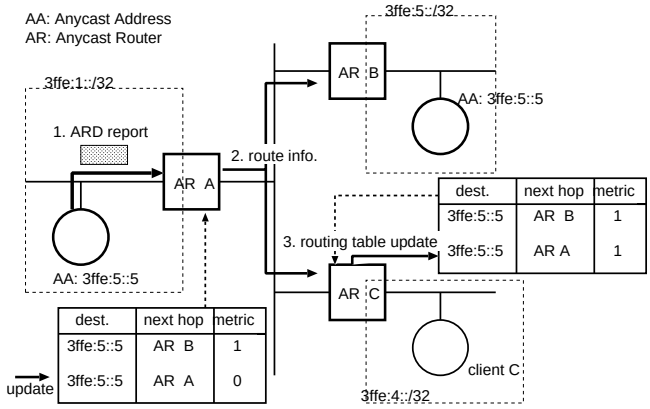

Figure 4 Example of DVARP

tus (join/leave) to the nearest anycast router. The method to find a host participating to an anycast membership (called *anycast host* below) is different according to the location of the anycast host. If the anycast host and the anycast router are on the same segment, the extended version of MLD (Multicast Listener Discovery) [10] is proposed [11]. We call it ARD (Anycast Receiver Discovery). An anycast host generates an MLD report message to the anycast router before joining the anycast membership. On the other hand, the anycast host sends an MLD leave message prior to leaving the membership. Because the destination address field of MLD packets are set to the link-local all routers address (`FF02::2`), this method is only can be used in the same scenario. Therefore, another method is needed to make a notification from the anycast host if the host and the anycast router are on the different segment. However, there is no implementation or proposal for this problem today. Note again that the method for correcting anycast hosts sometimes leads a serious security problem. The anycast router should have some mechanisms to avoid illegal and/or spoofed anycasst hosts notifications.

3. 2 Constructing and Updating Routing Table

DVARP

In DVMRP, since the multicast membership changes much dynamically, it is hard to specify the route which multicast packets will traverse before beginning the transmission. Therefore, flooding (or broadcasting) approach is effective.

On the other hand, the change of anycast membership is not so frequent rather than multicast. The routing information of anycast is more stable. Therefore, DVARP doesn't use flooding method but exchanges routing information periodically like RIP [5].

Figure 4 shows an example of updating routing table of DVARP. The operation of DVARP is shown below.

（1） If the anycast router detects changes of the anycast membership, the anycast router updates/creates the routing entry in its own routing table.

（2） Each DVARP router sends its own routing information to its adjacent routers periodically.

（3） If the router receives the routing information from adjacent routers, the router updates entries in the routing table.

The routing table of DVARP is shown in Table 3. This table has some (Destination, Next Hop, Metric, Flag, Timer) entries.

AOSPF

The routing table of AOSPF is shown in Table 4. The operation of AOSPF is described below.

Table 3 Routing Table of DVARP

Destination	Next Hop	Metric	Flag	Timer
3ffe:0:0:1::1	3ffe::1	3	0	60
	3ffe::2	5	0	100
3ffe:0:0:2::2	3ffe::2	2	0	120

Table 4 Routing Table of AOSPF

Destination	Next Hop	Metric
3ffe:0:0:1::1	3ffe::1	10
	3ffe::2	15
3ffe:0:0:2::2	3ffe::2	25

（1） If the anycast router detects the change of anycast membership, the anycast router updates/creates its entry in own link state database.

（2） When detecting any membership changes, Each AOSPF router sends the link state update to the adjacent AOSPF routers immediately.

（3） After updating/creating own link state database, the router uses Dijkstra SPF algorithm and calcurates the shortest path tree from the router. Then, the anycast router creates/updates its routing table from the shortest path tree.

These operations of AOSPF simmilar to those of DVARP except using the Dijkstra SPF algorithm. Another difference between these two protocols is the frequency of the routing information exchanges: The DVARP exchanges periodically while the AOSPF exchanges at the event of topology changes. This difference strongly affects the convergence time of the routing table. When a route change occurs in the AOSPF, the change is transmitted faster than the DVARP.

PIA-SM

PIA-SM (Protocol Independent Anycast-Sparse Mode) uses Core-Based-Tree algorithm like PIM-SM [9]. In this algorithm, the membership management is performed by the *core* router. We call this core node as Rendezvous Point (RP) like PIM-SM. An RP is selected among all PIA-SM routers and has the responsibility of managing anycast memberships. The packet toward an anycast address is once transmitted to the RP. After transferred to the RP, the packet can be transmitted to the appropriate anycast receiver by the RP. The registration information on RP equivalent to the routing table of DVARP or AOSPF consists of the anycast address, the next hop and the metric fields.

Figure 5 shows an example of a new registration to the RP of PIA-SM. Below, we describe the operations of the RP and other PIA-SM routers.

（1） If the anycast router detects the changes of anycast membership, the PIA-SM router reports the change of anycast memberships to the RP, which detected following two type s of message packets: PIA-Join and PIA-Prune. PIA-Join message represents a new anycast receiver to join to the membership. PIA-Prune message represents that the node does no longer join.

（2） If PIA-SM router (not RP) receives PIA-Join or PIA-Prune, it creates or cuts the corresponding anycast membership and send PIA-Join or PIA-Prune to upper PIA-SM routers toward the RP, respectively. If the PIA-SM already has corresponding entry and the downstream PIA-SM router differs, the next hop is added to existing entry for multipath routing.

（3） If the RP receives PIA-Join or PIA-Prune, it creates or cuts

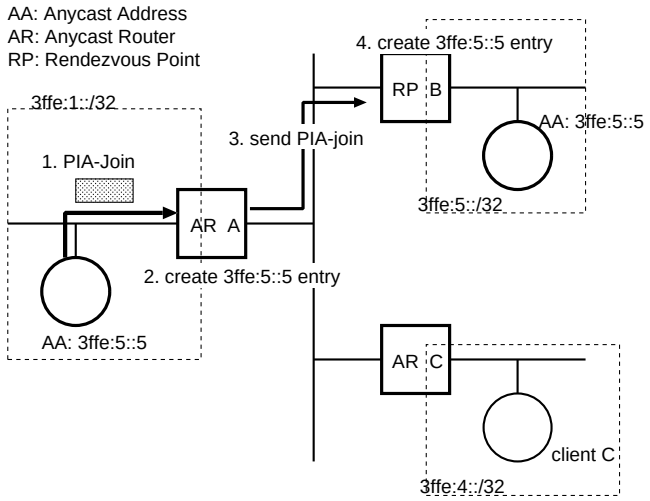

Figure 5 Example of PIA-SM

Table 5 Comparisons

		DVARP	AOSPF	PIA-SM
overhead	network	$O(gm)$	$O(gm)$	(RP):$O(ng)$
	router	$O(gs)$	$O(gs) + O(l * log(gm))$	(RP):$O(gs)$
convergence		hop by hop		none
implementability		not available		available

n: the number of all nodes in the network, g: the number of anycast group, m: the mean number of nodes which share the same anycast address, s: the mean number of anycast routing entries, l: the number of all links

the corresponding anycast membership. If the RP already has corresponding entry and the downstream PIA-SM router differs, the next hop is added to existing entry for multipath routing.

4. Comparisons of Anycast Routing Protocols

In this section, we compare our proposed protocol: DVARP, AOSPF, PIA-SM described in Section 3.. There are following three objectives in our comparison.

- Protocol Overhead (e.g., CPU load, memory consumption)
- Convergence Time from Membership Changes
- Implementability of Protocols

Table 5 summarizes comparison results. In the protocol overhead, both DVARP and AOSPF consume many network resources. These protocols' traffic consumption is liner to the number of anycast group and the number of nodes which share the same anycast address. Therefore, these protocols are applicable to the small network with high available bandwidth.

On the other hand, in PIA-SM, the traffic consumption never occurs because only the RP has the routing information and other PIA-SM routers have no routing information. Therefore, PIA-SM is more scalable than other two protocols. However, PIA-SM have another problem that anycast packets are not transferred through the oprimal path because the anycast packet is always transferred through the RP. Another problem of PIA-SM is the traffic concentration around the RP. These problems causes an extra packet transmission delays. From these perspective, PIA-SM is applicable to the large network, e.g., the Internet.

In the convergence time, DVARP is takes long time for route convergency. AOSPF is takes less time for route convergency than DVARP. In PIA-SM, since all routing information is kept only on the RP, it is not necessary to exchange routing information.

In the implementability, the implementation of PIM-SM is available now in IPv6. Other implementations of multicast routing protocols: DVMRP, MOSPF are not available in IPv6 as far as we know.

Above these results, each routing protocol have merits and demerits. In the near future, when the application using anycast communication is established, the routing protocol suitable for this application should be applied.

5. Conclusion

In this paper, we have considered a new routing architecture of anycast communication. From our survey, there are several problems in the current specification to realize this anycast routing. Our proposal design has transit model to the network which can treat anycast communications.

As future research topics, we implement our proposed routing protocols and verify that the anycast communication can be useful without (or with minimum) any application modification and/or protocol extention.

References

[1] R. Hinden and S. Deering, "IP version 6 addressing architecture," *RFC2373*, July 1998.

[2] S. Deering and R. Hinden, "Internet protocol, version 6 (IPv6) specification," *RFC2460*, December 1998.

[3] D. Xuan, W. Jia, W. Zhao, and H. Zhu, "A routing protocol for anycast messages," in *Proceedings of IEEE Transactions on Parallel and Distributed Systems*, pp. 571–588, June 2000.

[4] E. W. Dijkstra, "A note on two problems in connection with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.

[5] G. Malkin and R. Minnear, "RIPng for IPv6," *RFC2080*, January 1997.

[6] R. Coltun and D. Ferguson, "OSPF for IPv6," *RFC2740*, December 1999.

[7] D. Waitzman, C. Partridge, and S. Deering, "Distance vector multicast routing protocol," *RFC1075*, November 1988.

[8] J. Moy, "MOSPF: Analysis and experience," *RFC1585*, March 1994.

[9] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. gung Liu, P. Sharma, and L. Wei, "Protocol independent multicast-sparse mode (PIM-SM): Protocol specification," *RFC2117*, June 1998.

[10] S. Deering, W. Fenner, and B. Haberman, "Multicast listener discovery (MLD) for IPv6," *RFC2710*, October 1999.

[11] B. Haberman and D. Thaler, "Host-based anycast using MLD," *Internet draft* `draft-haberman-ipngwg-host-anycast-01.txt`, May 2002. (Expired November 2002).