Load-Balanced Anycast Routing

Ching-Yu Lin, Jung-Hua Lo, and Sy-Yen Kuo
Department of Electrical Engineering
National Taiwan University,
Taipei, Taiwan
sykuo@cc.ee.ntu.edu.tw

Abstract

For fault-tolerance and load-balance purposes, many modern Internet applications may require that a group of replicated servers dispersed widely over the world. The anycast as a new communication style defined in IPv6 provides the capability to route packets to the nearest server. Better quality of service (QoS) can be achieved by this kind of computing paradigm. DNS, Web Service, and Distributed Database System are three most well known examples. However, before anycasting can be realized, more researches need to be done. The anycast routing scheme is one of the most important issues. In this paper, we propose a load-balanced anycast routing scheme based on the WRS (Weighted Random Selection) method. We suggest that the server capability should be propagated along with other fields in the routing tables. An anycast routing algorithm should take into account the network transmission capability as well as the server processing capability for the selection of a target server. Three weight determination strategies are given. We also develop a simple algorithm to calculate the weights of WRS to achieve optimization under both the heavy and the light system traffic environment. Our approach is locally optimized to minimize the average total delay and well balanced for the server load.

1. Introduction

Many modern Internet applications may require that a group of replicated servers dispersed in different locations. These servers have the same information and provide the same services. A request issued from the client can be sent to and satisfied by any one of them. Usually, the nearest one is the most desirable. This kind of systems are designed for both fault-tolerance and load-balance objectives. The network transmission delay and the service processing time can be minimized to accomplish a better service quality by carefully choosing the target server. Domain Name Service (DNS), Web Service, and Distributed Database System are three most well known examples.

As compared with unicast, broadcast, and multicast, network-layer anycast is a newer service type defined in IPv6 to meet the user requirement. In IETF RFC 1546 [1], Partridge, Mendez, and Milliken describe the anycast service as below. *A host transmits a datagram to an anycast address and the internetwork is responsible for providing best effort delivery of the datagram to at least one, and preferably only one, of the servers that accept datagrams for the anycast address*. By moving the task of finding an appropriate server from client software to network, anycast can greatly simplify the effort of Internet applications. However, more researches still need to be done before anycasting can be realized. Furthermore, the routing scheme is one of the most important issues.

Recently research on anycasting includes many categories: anycast architecture, anycast routing algorithm, server selection policy, and so on. The architecture related topics include network-layer anycast, application-layer anycast [2], anycast in wireless Ad Hoc networks, and active anycast [3]. And the anycast routing algorithm related topics usually combine with the server selection policy together to achieve the objectives of load-balance and Quality of Service (QoS) [4, 5]. A survey can be found in [6].

In [4], Xuan et al. proposed an anycast routing protocol that is composed of two sub-protocols: the routing table establishment sub-protocol and the packet forwarding sub-protocol. In the routing table establishment sub-protocol, they considered four methods (Shortest-Shortest Path, Minimum-Distance, Source-based Tree, and Core-based Tree) to prevent the loop problem caused by multiple paths among routers. In the packet forwarding sub-protocol, they take a Weighted Random Selection (WRS) approach for multi-path selection to balance the network traffic. The simulation showed that the loop-prevention methods and the WRS approach have great impact on the performance in terms of average end-to-end packet delay. They mainly dealt with the network congestion problem of anycast. They

Proceedings of the Tenth International Conference on Parallel and Distributed Systems (ICPADS'04)
1521-9097/04 $ 20.00 IEEE

also tried to distribute the network traffic as even as possible and hold the loop-free property simultaneously. In their approach, routers select the target servers using the distance information in their routing tables. However, a router can do the same much better and only a slightly extended effort is needed.

Two main contributions of our paper are: (1) In our approach, we deal with both the network capability and the server capability simultaneously. We suggest that the server capability information should be propagated along with other information contained in routing tables. This makes routers have the potential capability to provide better load-balance and Quality of Service. Three weight determination methods are also presented for WRS. (2) A simple weight calculation algorithm is presented to achieve optimization under both light traffic and heavy traffic environments. Base on the queueing theory, our approach is locally optimal for load-balance and has the minimum average total delay.

The rest of this paper is organized as follows. In Section 2, we briefly describe the basic knowledge of routing schemes. We also present the problem model description and the theoretic derivations in this section. Then we present our load-balanced anycast routing scheme in Section 3. Some remarks are also discussed here. Evaluations and analysis are given in section 4. Section 5 contains the conclusions.

2. Problem Model and Mathematical Derivations

2.1 Preliminary

The main function of the network layer defined in OSI (Open Systems Interconnection) Reference Model is routing packets from the source host to the destination host. The routing algorithm is the major part of the network layer software responsible for determining which output link an incoming packet should be transmitted on [7].

In 1990, OSPF (Open Shortest Path First) protocol [8] proposed by Internet Engineering Task Force (IETF) has become the standard of Internet routing protocol and supported by most router vendors. OSPF is an open standard, supports a variety of distance metrics (physical distance, delay, etc.), and is a dynamic algorithm.

A routing scheme can be divided into two stages: (1) build the routing table, (2) select the outgoing interface. In the first stage, a link state advertisement sub-protocol is needed and the routing information will be propagated. The routing table including the destination, cost, and next hop fields will be established. In the second stage, when a packet is received by a router, the router must apply its routing algorithm to choose an outgoing interface and forward the packet.

A good routing scheme should work both correctly and efficiently. There are several strategies for the routing algorithm optimization. Minimizing the mean packet delay and maximizing the total network throughput are two good candidates. Minimizing the mean packet delay can improve the Quality of Service. Meanwhile, maximizing the total network throughput can increase the network utilization. These two goals are not always attainable at the same time. In reality, they are usually in conflict with each other.

For the traditional unicast routing scheme, the client host itself determines the destination address. The only task of the network is to route the packet to the assigned destination host as fast as possible. The optimal routing scheme is the one with the minimum transmission delay. However, for an anycast routing scheme, the network may have more than one choice of the destination server for an anycast address. The server having the minimum transmission time but with a heavy loading may not be the best choice. On the other hand, the one having the smallest service processing time but being very far away may not be favorable either. The user response time includes the network transmission time and the server processing time. From the Quality of Service view, the final goal of an anycast routing scheme should minimize the end-to-end user response time. Both the network capability and server capability information are beneficial for the routing scheme.

2.2 Problem Model

A network consisting of a number of nodes and links is usually considered as a connected graph $G = (V, E)$ where V is a set of vertices representing the hosts (and/or routers) and E is a set of edges representing the links. An intermediate node is called router R which is responsible for packet transmission, and a boundary node is called host H. In our discussion, a host can be a client machine or a server machine. A client machine is where the request packet is issued, and a server machine processes the request and sends back the response packet. The sequence of routers through which a packet is transmitted is called a path P. Each edge is associated with a numerical value called distance d. The distance is usually assigned with the delay time or the bandwidth of the link.

Figure 1 shows the problem model. R is a router in the network. S_1 to S_N are a group of servers with a specific anycast address and map to the entries of routing table on R. We assume that the packet arrival pattern is a Poisson Process with arrival rate λ. Using the Weighted Random Selection method, the incoming packets are distributed into the outgoing interfaces I_1 to I_N with the destination

servers S_1 to S_N according to the corresponding weights W_1 to W_N. Furthermore, each procedure that sends packets from R to S_j and processed at S_j can be modeled as a *M/M/1* queueing system [9, 10].

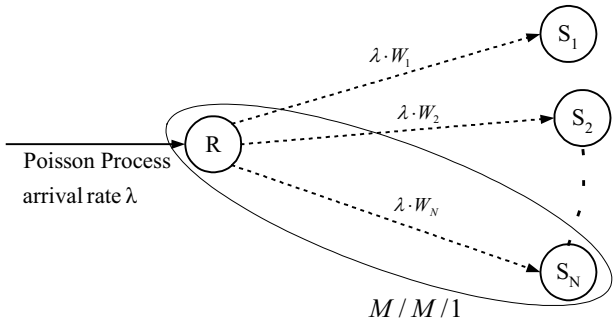

Figure 1. Problem model

2.3 Mathematical Derivations

We would like to minimize the average delay time in the system as much as we can. From the queueing theory [9, 10], we have the average delay time T_i of each *M/M/1* system as follows:

$$T_i = \frac{1}{\mu_i - \lambda W_i}, \text{ } \mu_i \text{ is the average service rate.}$$

Therefore, we can model the optimization problem for anycast routing as below. The optimization problem can be represented as follows:

Minimize:

$$\sum_{i=1}^{N} W_i \frac{1}{\mu_i - \lambda W_i}, \quad (1)$$

Subject to

$$\sum_{i=1}^{N} W_i = 1, \quad W_i \geq 0, \quad i = 1, 2, \ldots, N. \quad (2)$$

To solve the above problem, the Lagrange multiplier method [11] can be applied. Eq. (1) and Eq. (2) can be simplified as follows:

Minimize:

$$L(W_1, W_2, \ldots, W_N, a) = \sum_{i=1}^{N} W_i \left(\frac{1}{\mu_i - \lambda W_i}\right) + a\left(\sum_{i=1}^{N} W_i - 1\right) \quad (3)$$

By the Kuhn-Tucker conditions, the necessary conditions for a minimum of Eq. (3) to exist are as follows:

$$\frac{\partial}{\partial W_i} L(W_1, W_2, \ldots, W_N, a) = \frac{\mu_i}{(\mu_i - \lambda W_i)^2} + a = 0 \quad (4)$$

$$\frac{\partial}{\partial a} L(W_1, W_2, \ldots, W_N, a) = \sum_{i=1}^{N} W_i - 1 = 0$$

The solution of W_i from Eq. (4) is

$$W_i = \frac{1 - \sum_{j=1}^{N} \mu_j / \lambda + \sqrt{\mu_i} \sum_{j=1}^{N} \sqrt{\mu_j} / \lambda}{\sum_{j=1}^{N} \sqrt{\mu_j} / \sqrt{\mu_i}}, i = 1, 2, \ldots, N. \quad (5)$$

Similar derivations can be found in the Appendix 3 of Xuan's work [4]. In section 3.3, we will propose a simple optimal algorithm to calculate W_i.

3. Load-Balanced Anycast Routing Scheme

When an anycast packet arrives at a router, the router needs to select an outgoing interface according to the information contained in its routing table. The sufficiency of input information and the appositeness of algorithm determine the performance of the routing scheme. Based on WRS, we can improve the routing scheme from three aspects. (1) We suggest that the server capability should be propagated along with other information contained in a routing table entry. By taking into account the network link capability and the server capability simultaneously, the routing scheme get more information to achieve better load-balance and Quality of Service. (2) We present three weight assignment methods: the first one takes account of network congestion only, the second one takes account of server load, and the third one takes account of both at the same time. (3) We propose a simple algorithm to calculate the weights of WRS. This algorithm holds the optimization property of WRS under both heavy loading traffic and light loading traffic environments. The first aspect is applied to the routing table establishment stage, and the second and third aspects are applied to the outgoing interface selection stage of the routing scheme. We explain our load-balanced anycast routing approach below.

3.1 Routing Table Establishment Stage

The routing table establishment stage constructs the routing table to provide the needed information in the routing scheme. An entry in the common routing table usually includes the destination address, distance, and next-hop fields. The entries with the addresses that match the destination field in the packet are the candidates for the routing algorithm. The candidate entries for an anycast address can be multiple in two ways: (1) a single anycast address with multiple target server addresses, (2) a target server with multiple routing paths. The distance field presents the information of the transmission time needed from router to destination. The routing algorithm prefers to select the entry with smallest distance away from

candidates and forward the packet to the outgoing interface in the next-hop field.

Choosing the entry with shortest path can minimize the transmission delay. However, it also tends to make the network congested and the server overloaded since all the packets are sent to the same target server along the same routing path. Furthermore, the response time is the summation of transmission delay and server processing time. The minimization of the response time is a more appropriate goal than the minimization of the transmission time only.

Therefore, to minimize the average response time as much as possible, we would like to balance the network traffic and the server loading by distributing incoming packets over more than one outgoing link. To achieve that, the information on the network link capability and the server capability are needed. So we suggest that the server capability field should be added and can be measured by the maximum number of packets that can be processed per unit time. Moreover the distance field can be used as a measurement of the network link capability. We assume that a routing table or topology information exchange protocol like RIP or OSPF is included in the routing scheme. Thus these extended information can be propagated and updated. Some extensions may be needed for these table information exchange protocols and are not included in this paper.

Figure 2 is a sample network used for discussion. H_1 to H_4 are client machines that issue anycast packets. R_1 to R_{12} are routers. And S_1 to S_4 are server machines that have the same anycast address A_1 and can deal with the packets. Table 1 is the modified routing table at R_1.

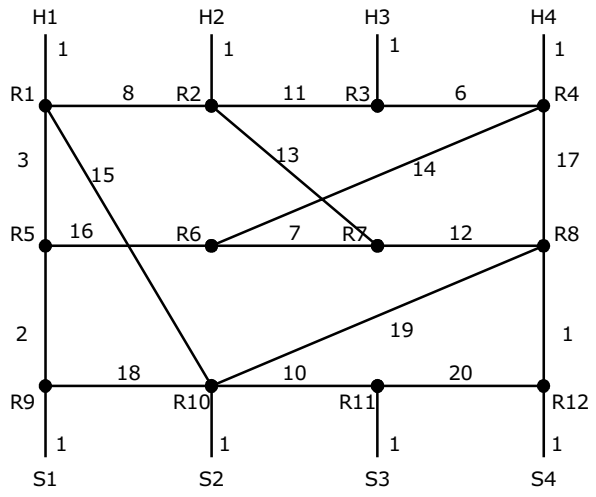

Figure 2. Sample Network.

Table 1. Routing Table at Router R_1.

Destination	Distance	Next Hop	Server Capability*
A1, S1	*6*	*R5*	*6*
A1, S2	*16*	*R10*	*15*
A1, S3	*26*	*R10*	*9*
A1, S4	*35*	*R2*	*8*

3.2 Outgoing Interface Selection Stage

Once the routing table has been built on a router, it can be used to route packets. As mentioned above, the candidate entry could be multiple for an anycast packet and we want packets be distributed into more than one outgoing interface. We introduce a routing algorithm called Weighted Random Selection (WRS) method here. In WRS, every outgoing interface is assigned a weight and selected randomly. The probability of an outgoing interface been selected is proportioned to its corresponding weight. By carefully determining the weights, we can control the distribution of packets among the outgoing interfaces. By this way, the network traffic and the server loading can be balanced.

Eq. (5) in section 2.3 gives us an optimal solution for the determination of W_i. Assume that for an anycast address, there are N candidate entries in the routing table of router R. Let us index these N candidate entries by *1*, *2*, …, *N*. Then the values in the distance fields of these entries can be denoted as $D_1, D_2, \ldots, D_N$. The values in the server capability fields that are suggested to be added in section 3.1 can be denoted as $C_1, C_2, \ldots, C_N$. The weights of these entries are $W_1, W_2, \ldots, W_N$, respectively.

The weight assignment method is the key step of the WRS algorithm. Usually we assume the packets arrive at the router R is a Poisson process with rate λ. Therefore, for each routing path from R to S_i modeled as a *M/M/1* system, the arrival rate $\lambda_i = \lambda * W_i$. There are several strategies to determine the service rate μ_i. We describe three of them below.

Method 1. $\mu_i = \frac{1}{D_i}$

D_i is the needed transmission time from router R to server S_i. Method 1 takes account of the network congestion problem only. The assignment of W_i balances the traffic into different outgoing links and locally optimizes the average transmission delay. When the server loading is light, this method works better.

Method 2. $\mu_i = C_i$

C_i is the number of packets that server S_i can serve per unit time. Method 2 considers the server overload

problem only. The assignment of W_i balances the server load and locally optimizes the average server processing time. When the network traffic is light, this method works more efficiently.

Method 3. $\mu_i = \dfrac{1}{D_i + 1/C_i}$

D_i + 1/ C_i is the summation of the network transmission time and the server processing time and is called the total delay (or the response time). Method 3 takes account of both the network congestion and the server overload problems. The assignment of W_i balances the network traffic and the server load simultaneously and locally optimizes the average delay.

Substituting the service rate μ_i into Eq. (5), we get the weight assignment equations of the above three methods respectively. We suggest that an anycast routing algorithm should adopt method 3 as their routing strategy to achieve the best performance on QoS and server load-balance. The evaluations and comparisons of these three methods are given in section 4.

3.3 Weight Calculation Algorithm

By applying the Lagrange multiplier method we can get solution of the optimization problem. The weight W_i of the WRS method can be obtained by substituting related parameters into Eq. (5). We need all $W_i \geq 0$ since the probability of a packet transmitted into interface i must be positive. However, this requirement cannot be guaranteed by the Lagrange multiplier method. Table 2 gives an example to show this problem. It occurs frequently under at a light traffic (with relatively smaller arrival rate λ) environment. To minimize the average delay in Eq. (1) and still satisfy the condition in Eq. (2), the Lagrange multiplier method forces the weight of the interface with longer delay to become negative and obtains a bigger weight (maybe > 1) on the interface with smaller delay.

The simple algorithm we proposed below can prevent this problem and achieve optimization whatever the arrival rate λ be. For general cases, this algorithm ends in a couple of rounds.

Table 2. Problem of Lagrange multiplier method.

Interface i	Service Rate μ_i	W_i (Eq. 5)	W_i (Algorithm*)
1	*100*	*-0.3149*	*0.0000*
2	*1*	*-0.0360*	*0.0000*
3	*10000*	*1.3509*	*1.0000*
□ Arrival Rate λ = 2000			

Algorithm*:

Step 1: Set $r = 0$.

Step 2: Calculate the following equations

$$W_i = \frac{1 - \sum_{j=1}^{N-r} \mu_j / \lambda + \sqrt{\mu_i} \sum_{j=1}^{N-r} \sqrt{\mu_j} / \lambda}{\sum_{j=1}^{N-r} \sqrt{\mu_j} / \sqrt{\mu_i}}, i = 1,2,..., N-r.$$

Step 3: Rearrange the index i such that

$$W_1^* \geq W_2^* \geq ... \geq W_k^* \geq 0 > W_{k+1}^* \geq W_{k+2}^* \geq \ldots \geq W_{N-r}^*.$$

Step 4: If $k = N - r$ then stop, else update

$r = r + k$ and W_{k+1}^*, W_{k+2}^*, …, $W_{N-r}^* = 0$.

Step 5: Go to Step 2. □

Moreover, the optimal solution has the form:

$$W_i^* = \frac{1 - \sum_{j=1}^{N-r} \mu_j / \lambda + \sqrt{\mu_i} \sum_{j=1}^{N-r} \sqrt{\mu_j} / \lambda}{\sum_{j=1}^{N-r} \sqrt{\mu_j} / \sqrt{\mu_i}}, i = 1,2,..., N-r.$$

$W_i^* = 0, otherwise.$

3.4 Remarks

We give some remarks for our approach below.

1. Application Layer vs. Network Layer Approach

Anycasting can be implemented either in the application layer or the network layer. An application layer approach determines the destination address of the target server at the client machine where the packet has been issued. It provides the potential for optimizing the end-to-end response time. However, a large amount of extra works needs to be done by the application itself. For example, the application needs to maintain the information on server load, network topology, transmission time, and so on. Several new protocols may be needed for the communication in the system.

The network layer approach combining with the existing IP protocols can significantly reduce the effort by the application software. Due to the lack of global information of the application traffic, the router can only locally optimize the traffic directly through it. However, from the theory of statistics, a locally optimal solution can provide a near optimal approach.

2. Dynamic vs. WRS Load-Balancing Approach

A dynamic load-balancing approach monitors the states of system and dynamically dispatches tasks. It quickly reflects the change of system and can relatively closes to the real optimization. However, it consumes more resources and usually is more complicated.

The WRS approach is based on the probability model and dispatches tasks according to the weights. In our approach, the weight is pre-calculated according to the capability of network link and server. In most cases, these two parameters do not change frequently. The information update mechanism provided by the common routing table information exchange protocol can be used as usual.

3. Determining D_i and C_i

A lot of works have been done on the measurement of the network link capability and the server capability. To explain the load-balanced anycast routing scheme, the reciprocal of distance field in the modified routing table (Table 1) is selected to represent the network link capability D_i. This makes the transferring from the network link capability to the transmission delay become easy and possible. Meanwhile, the maximum number of packets that can be processed per unit time recorded in the server capability field is selected to represent the C_i. To apply this load balanced anycast routing scheme, one may choose the most suitable measurement method.

4. Loop-Prevention Problem

The candidate entries of a routing table for an anycast address may be multiple for two reasons: (1) a single anycast address with multiple target server addresses, (2) a target server with multiple routing paths. That makes the routing path become multiple. We need a mechanism to determine the order among routers. Four methods have been presented to do that, the Shortest-Shortest Path (SSP) method, the Minimum Distance (Min-D) method, the Source-Based Tree (SBT) method, and the Core-Based Tree (CBT) method. All four methods can be combined with our approach. We define the candidate entries used in our approach are all the eligible ones that have been verified by any one of the above four methods.

5. Flow Type Traffic Problem

If a sequence of packets arrived on a router have been sent to the same server, these packets can be identified as a flow in IPv6. The leading packet(s) is processed and routed by the routing algorithm normally. Once the flow is identified, the following packet with same flow label in its IP header will be routed according to the previous result, and then transmitted to the same outgoing interface. Our approach is compatible with this mechanism and can be used in the routing process at the leading packet(s) of a flow.

4. Evaluations and Analysis

Considering the router R, we randomly generalize the entries in the routing table in our evaluation model. The size of an anycast group is assumed to be ten. Furthermore, the distance and server capability are both normalized to be one. The average server load and the average delay of three weight determination methods are listed in Table 3. And for comparison purpose, the fixed method with $r=1$ in [4] is recalculated and adjusted to fit into our evaluation parameters.

	μ_i	Average Delay	Average Server Load
Method 1	$\mu_i = \frac{1}{D_i}$	$\sum W_i \cdot (T_i + 1/C_i)$	$\frac{\lambda \cdot W_i}{C_i}$
Method 2	$\mu_i = C_i$	$\sum W_i \cdot (T_i + D_i)$	$\frac{\lambda \cdot W_i}{C_i}$
Method 3	$\mu_i = \frac{1}{D_i + 1/C_i}$	$\sum W_i \cdot T_i$	$\frac{\lambda \cdot W_i}{C_i}$
□ $T_i = \frac{1}{\mu_i - \lambda W_i}$			

Table 3. Evaluation Parameters.

Figure 3, 4, and 5 depict the evaluation results. Figure 3 shows the relationship between the average server load and the arrival rate. Figure 4 shows the tendency of the average delay versus the arrival rate. Figure 5 is an enlarged version at Y coordinate axis of Figure 4 to show clearly the differences between various methods. We point out below some observations from the figures.

1. The assumptions and results of Method 1 and fixed method in [4] are similar. They both consider the network delay problem only and ignore the limitations of the server capability. The only difference between them is that Method 1 uses the *Algorithm** to rearrange the weights if their values become to negative. The curves of these two methods in three figures are very close.
2. Since both Method 1 and fixed method in [4] do not take account of the server capability problem, they may transmit too many packets to some servers and make these servers overloaded. In

Figure 3, the average server loads of these two methods can exceed the normalized value 1 and those of other two methods won't. Method 2 takes account of the server capability only and Method 3 takes account both the server capability and the network delay, so their values of average server load are much smaller than others.

3. Figure 4 shows the relationship between the average total delay and the arrival rate. The average total delay rapidly increases when the arrival rate becomes close to one. That means the total delay may become extremely large if the traffic is very heavy.

The average server load can be taken as a measurement of the server load-balance characteristic. In the meanwhile, the average total delay can be taken as a measurement of the user Quality of Service. From the evaluation results, we know the WRS-based routing algorithms for anycast are desirable and should consider both the network transmission delay and the server processing delay.

5. Conclusions

To obtain a better performance on QoS and server load-balance, we improve the routing algorithm from three aspects. We conclude that the server capability should be propagated along with other fields in the routing table. A strategy considering both the network congestion and the server overload was also presented for the determination of the weights for WRS method. Last but not least, we showed the problem of finding the optimal weights under light traffic environment and proposed a simple algorithm to solve the problem. Evaluations and analysis were given to show that our approach could achieve better performance on the average delay and the average server load.

6. References

[1] C. Partridge, T. Mendez, and W. Milliken, "Host Anycasting Service," *IETF RFC 1546*, November 1993.

[2] E. Zegura, M. Ammar, Z. Fei, and S. Bhattacharjee, "Application-Layer Anycasting: A Server Selection Architecture and Use in a Replicated Web Service," *ACM/IEEE Transactions on Networking*, 8(4): 455-466, 2000.

[3] H. Miura, and M. Yamamoto, "Server Selection Policy in Active Anycast," *IEICE Transactions on Communications*, Vol. E84-B. No. 10, October 2001.

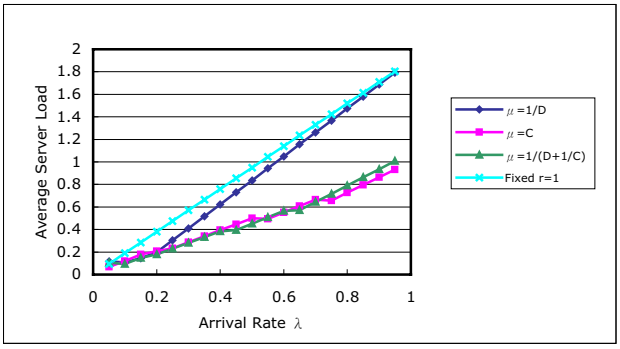

Figure 3. Average Server Load.

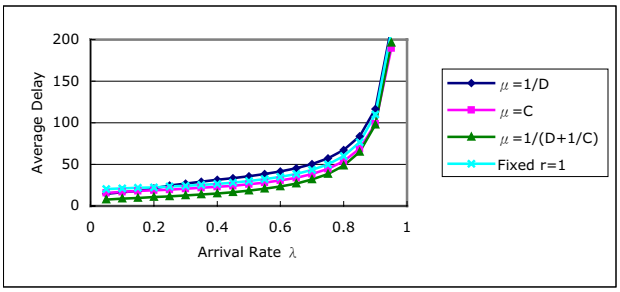

Figure 4. Average Delay.

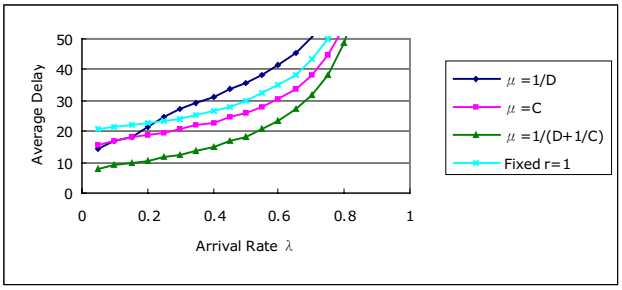

Figure 5. Average Delay with Enlargement.

[4] D. Xuan, W. Jia, W. Zhao, and H. Zhu, "A Routing Protocol for Anycast Messages," *IEEE Transactions on Parallel and Distributed Systems*, Volume: 11 Issue: 6, Jun 2000.

[5] W.T. Zaumen, S. Vutukury, and J.J. Garcia-Luna-Aceve, "Load-Balanced Anycast Routing in Computer Networks," *Proceedings Fifth IEEE Symposium on Computers and Communications (ISCC 2000)*, July 2000.

[6] S. Yu, W. Zhou, and Y. Wu, "Research on Network Anycast," *Proceedings of the Fifth International Conference on Algorithms and Architectures for Parallel Processing*, 2002.

[7] Andrew S. Tanenbaum, *Computer Networks, third edition*, Prentice-Hall, 1996.
[8] J. Moy, "OSPF Version 2," *IETF RFC 1247*, July 1991.
[9] Donald Gross, and Carl M. Harris, *Fundamentals of Queueing Theory, third edition*, Wiley-Interscience, 1998.
[10] Sheldon M. Ross, *Introduction to Probability Models, 7th edition*, San Diego: Harcourt/Academic Press, 2000.
[11] M. S. Bazaraa and C. M. Shetty, *Nonlinear Programming: Theory and Algorithm*, John Wiley & Sons, 1993.