

Rerouting기법들 간에 Orthogonal 관계를 통한 차별적인 서비스 제공에 관한 연구

한 정 수[†] · 정 진 옥^{††}

요 약

사용자가 요구하는 서비스들은 그들의 QoS에 따라 다양하게 분류되어질 수 있으며, 이렇게 각기 다른 특성을 갖는 서비스들이 네트워크 상에 발생할 수 있는 장애 상황에 대처하여 안전하고 효율적인 서비스를 제공하기 위해서는 이들 연결에 대한 재라우팅(rerouting) 기법이 사용되어야 한다. 본 논문은 장애가 발생할 수 있는 IP 네트워크 상에서 차별화 된 서비스 제공을 위해 각 서비스들의 QoS에 따라 이들의 우선순위를 분류하고 이에 맞는 재라우팅 기법을 연구하고자 한다. 이를 위해서 재라우팅을 시작하는 시작점에 따라 Source Rerouting, Link Rerouting 기법을, 재라우팅을 수행하는 시간에 따라 Immediate Rerouting, Random Rerouting을 연구하여 이들에 대한 효과적인 orthogonal 관계에 대한 연구를 통해 서비스의 우선순위에 따라 보다 높은 성능을 갖는 DRIT와 DRDT 재라우팅 기법을 연구하고자 한다. 결과적으로 차별화 된 서비스를 지원하기 위해 각 재라우팅 기법을 차별적으로 적용한 DRDT 보다 높은 성능을 보이고 있다.

Providing Differentiated Services through Orthogonal Relationship among Rerouting Mechanisms

Jeong-Soo Han[†] · Jin-Wook Chung^{††}

ABSTRACT

Rerouting mechanisms must be used by connections in order to provide QoS (Quality of service) characterization of services, which provides mean for reliable and efficient transfer of services under fault generating network. Also, user's services can classify according to their QoS characterizations. In this paper, we study classification of user services according to their characterization for providing differentiated services, and propose rerouting mechanisms under fault generating network. For this, we study various rerouting mechanisms including rerouting locus of start (Source Rerouting, Link Rerouting), rerouting timing of start (Immediate Rerouting, Random Rerouting) and their orthogonal relationship, eventually we propose new rerouting mechanisms such as DRIT, DRDT which show higher performance according to priority of services than others. Our simulation shows that rerouting mechanism (DRDT), applied differentiated mechanisms is better performance to provide differentiated service.

키워드 : 재라우팅(Rerouting), 자원예약 프로토콜(RSVP), QoS, 라우팅 매트릭스(routing matrix)

1. 서 론

사용자가 요구하는 서비스는 일반적인 데이터에서 영상, 화상 등 멀티미디어 서비스까지 그 종류가 다양하고, 이들이 요구하는 QoS(Quality of Service) 역시 delay, bandwidth, jitter, packet loss 등으로 각각 다양하다[1, 2]. 따라서, 이들이 요구하는 QoS에 따라 서비스 우선순위를 분류하고 이들을 차별화 된 정책에 의해 서비스해야 한다. 이러한 차별화 된 정책의 궁극적인 목적은 각 서비스의 QoS에 맞게 빠른 서비스를 제공함과 동시에 안전한 서비스를 제공함으로써 네트워크 상에서 각 서비스의 처리율(through-

put)을 높이는 것이다. 더욱이 사용자가 요구하는 서비스를 기존의 IP 네트워크 상에서 제공한다는 것은 그 한계가 있기 때문에 새로운 프레임워크에 대한 연구가 발생하게 되었다. 이에 IETF(Internet Engineering Task Force)에서는 사용자의 서비스를 각 플로우(flow)로 분류하고 이에 대한 QoS를 제공하기 위해서 각 플로우에 대한 상태를 유지함과 동시에 네트워크 자원을 예약하는 RSVP(Resource reSerVation Protocol) 프로토콜을 사용하는 Intserv(Integrated Service), 그리고 이에 대한 문제점을 해결하기 위한 새로운 프레임워크 Diffserv(Differentiated Service)를 통해 서비스를 여러 클래스(class)로 분류하고 이를 에지(edge) 라우터에서 그 행동(per-hop behavior)을 결정함으로써 각 서비스에 대한 QoS를 제공하려는 노력이 생기게 되었다[3,

[†] 준 회원 : 성균관대학교 대학원 전기전자 및 컴퓨터공학부
^{††} 종신회원 : 성균관대학교 전기전자 및 컴퓨터공학부 교수
논문접수 : 2001년 12월 19일, 심사완료 : 2002년 5월 23일

4]. 이들 연구들은 차별화 된 서비스들을 그 QoS에 맞게 연결 설정을 수행함으로써 네트워크 상에서 각 서비스의 처리율을 높이고자 하는 연구들이다[3]. 또한, 이렇게 인터넷 상에서 서비스에 대한 QoS를 제공하려는 연구 중에 하나가 연결 설정에 대한 재라우팅(Rerouting) 기법이다. 재라우팅은 연결지향 서비스에 대해서 각 서비스가 연결 설정 과정을 수행하는 동안 네트워크 상에 발생하는 장애 상황 때문에 올바르게 연결 설정을 수행할 수 없을 경우에 우회로(detour)를 결정하는 라우팅 기법이다. 이러한 장애 상황은 네트워크 장비나 회선의 단절과 같은 물리적인 장애로 인해 연결 설정을 수행할 수 없는 경우와 네트워크 자원 부족 등으로 인해 올바르게 연결 설정을 수행할 수 없는 논리적인 장애로 나눌 수 있는데, 본 논문에서는 연결 설정에 대한 재라우팅의 원인을 물리적인 장애로 제한하기로 한다.

재라우팅 기법에는 재라우팅을 시작하는 위치(locus)와 재라우팅을 수행하는 시점(timing), 그리고 재라우팅 시도(retry) 횟수에 따라 다양하게 분류될 수 있다. 이러한 기법들은 네트워크 자원과 관련하여 라우팅 성공률(routing success rate)와 라우팅 시간(routing time)사이에서 trade-off가 발생하게 된다. 이러한 이유로 재라우팅 기법들을 모든 서비스에게 일괄적으로 적용하는 것은 사용자 측면에서 사용자가 요구하는 서비스에 대해 QoS 제공이 불가능할 뿐 아니라 네트워크 측면에서 자원을 효율적으로 사용할 수 없는 문제가 발생하게 된다. 따라서 사용자 서비스를 QoS에 따라 분류하고 이들을 각기 다른 재라우팅 기법들로 적용하여 사용자와 네트워크 측면에서 효율적인 서비스를 제공해야 한다. 이를 위해 본 논문에서는 다양한 재라우팅 기법들에 대한 orthogonal 관계를 연구하고 이를 차별화 된 서비스에 적용함으로써 높은 처리율 제공하고자 한다.

본 논문에서 제안하는 네트워크 모델은 Intserv나 Diff-serv[4]에서 제안하는 특정 네트워크 모델에 국한하지 않고 다만, 사용자 서비스를 그 QoS에 맞게 여러 클래스로 분류하고 이에 대한 연결 설정과 함께 자원을 예약하는 방식을 사용함과 동시에 이에 대한 효율적인 재라우팅 기법을 연구하는 것에 국한하기로 한다.

본 논문의 구성은 다음과 같다. 2장에서 재라우팅과 관련된 관련연구 및 동향을 살펴보고, 3장에서는 기존의 IP 네트워크 상에 재라우팅을 제공하기 위한 방법론과 성능을 평가할 수 있는 항목들에 대해 살펴보기로 한다. 그리고 4장에서는 사용자 서비스를 그 QoS에 맞게 여러 클래스로 분류함과 동시에 재라우팅 기법에 대한 orthogonal 관계를 연구하고 이를 통해 새로운 재라우팅 알고리즘 제안한다. 5장에서 이들을 시뮬레이션을 통해 기존의 기법들과 비교, 분석하게 된다. 결론은 6장에서 제시하도록 한다.

2. 관련 연구

장애가 발생하는 네트워크 환경 하에서 재라우팅에 대한 연구[5,6]는 각 연결에 대한 QoS 보장이라는 관점에서 수행되어졌다. 특히, [5]에서는 재라우팅 시간과 재라우팅 장소라는 두 가지 관점에서 GPC(Guaranteed Performance Communication) 네트워크 상에서 single-fault의 recovery에 대한 연구가 수행되어졌으며, [6]는 [5]의 연구를 한층 발전시켜 multi-fault 환경과 retry model 항목을 추가함으로써 여러 곳에서 fault가 발생하였을 때 재라우팅 기법을 통해 fault recovery에 대한 연구를 수행하였다. 또한, [7]에서는 재라우팅 기법에 대한 접근 방법으로 Link rerouting과 end-to-end rerouting으로, centralized와 distributed schemes 그리고, pre-computation과 dynamic computation등 3가지 측면으로 나누어 분류하고 있다. 그러나 이러한 연구들은 모든 연결을 동일시 하여 서비스함으로써 서비스에 대한 특성을 제대로 반영하지 못하고 있다.

각 연결의 QoS 제공을 위해 사용되는 자원 예약 프로토콜인 RSVP는 자원을 확인 과정과 예약 과정을 분리하여 수행하는 soft-state 예약 과정을 수행하게 된다[8]. [6]에서는 자원예약을 위해 사용하는 프로토콜을 RSVP에 국한하지 않고 단지, forward 방식과 backward 방식으로 나누어 분류한 후, 각 성능을 비교 설명함으로써 재라우팅 시 자원 예약을 위해서는 전진예약(forward reservation) 방식이 우수한 성능을 제공한다는 것을 제시하고 있다.

본 논문에서는 기존의 재라우팅 기법에 사용되는 방식에 대한 연구를 통해 새로운 관계를 갖는 재라우팅 기법을 제안한다. 이는 서비스의 특성에 맞는 차별적인 서비스 제공을 그 목적으로 하고 있으며, 이 때 사용되는 자원 예약방식은 전진예약 방식을 사용한다.

3. 재라우팅 기법

본 장에서는 장애 발생 시 각 연결에 대한 QoS 제공을 위해 사용되는 재라우팅 기법에 대해서 살펴보고, 이들의 성능을 분석하기 위한 성능 항목에 대해 알아보기로 한다.

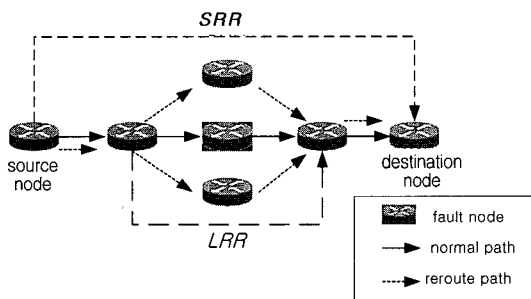
3.1 방법론

네트워크 상에 장애가 발생했을 경우 새로운 패스를 결정하기 위해 재라우팅 기법들을 선택할 경우에는 여러 가지 고려해야 할 사항이 있다. 첫째, 신속한 라우팅 연결이며, 둘째로는 같은 네트워크 자원으로 많은 연결을 서비스할 수 있도록 효율적인 기법이어야 한다는 것이다. 마지막으로 다양한 서비스들의 그 특징에 맞게 적절한 기법을 사용해야 한다는 것이다. 이에 본 논문에서 네트워크 상에 장애가 발생했을 경우 각 연결에 대해 사용하는 재라우팅 기법으로 크게 재라우팅을 시작하는 장소에 따른 기법과 재라우팅을 시작하는 시점에 따른 기법 등으로 나누어 생

각한다. 본 절에서는 각 기법들의 방법론과, 이들의 장단점을 살펴보기로 한다.

3.1.1 재라우팅 시작 장소에 따른 분류

재라우팅을 시작하는 장소에 따라 Source Rerouting(이하 SRR)과 Link Rerouting(이하 LRR)으로 분류할 수 있다. 먼저 end-to-end rerouting이라고도 하는 SRR은 각 연결에 대해 재연결되는 구간(path segment)이 송신지 노드에서부터 장애가 발생한 장소(회선이나 노드)의 다음 노드이다. 이러한 SRR은 송신지 노드에서 다시 라우팅을 수행하기 때문에 재연결되는 구간상에 재연결의 목적지(장애가 발생한 장소의 다음 노드)까지 많은 패스가 존재(각 연결의 QoS에 맞는 다른 패스가 존재)할 수 있어 라우팅 실패율(routing blocking rate)이 감소할 수 있다. 그러나, 장애를 감지하고 송신지 노드에서 다시 연결 설정을 수행해야 하므로 라우팅에 걸리는 시간(routing time)은 그만큼 오래 걸리게 된다. 이와 달리 재연결되는 구간이 장애가 발생한 장소의 전(before) 노드에서부터 장애 발생 장소의 다음(after) 노드까지인 LRR은 재연결되는 구간이 SRR에 비해 짧기 때문에 각 연결의 QoS를 보장할 패스가 확률적으로 적어 라우팅 실패율이 높아질 수 있다. 하지만 짧은 재연결 구간에 의해 라우팅에 소요되는 시간은 그만큼 감소하게 되는 장점을 가지고 있다. (그림 1)은 이러한 SRR과 LRR에 대한 동작 그림을 보여주고 있다.



(그림 1) SRR과 LRR의 동작도

3.1.2 재라우팅 수행 시작 시점에 따른 분류

네트워크 상에 장애가 발생했을 경우, 각 연결에 대해서 재라우팅을 수행해야 하는데 재라우팅 기법을 언제 수행하는가에 따라 Immediate Rerouting(이하 IRR)과 Random Rerouting(이하 RRR)로 분류할 수 있다. 먼저 IRR의 재라우팅 수행 시점은 장애 발생에 의해 라우팅 연결이 실패하는 즉시 재라우팅 기능을 수행하게 되어 재라우팅 수행에 걸리는 시간이 다른 방식에 비해 짧게 걸리는 장점을 가지고 있으나 다른 연결들이 네트워크 자원을 점유하는 시점에 다시 재라우팅을 수행할 확률이 많기 때문에 라우팅 연결 실패율이 비교적 높다. 이에 비해 RRR은 라우팅 연결이 실패할 경우 재라우팅 기능을 랜덤 시간(random)만큼 대기한 후 수행하는 방식이기 때문에 라우팅 수행시간은 IRR보

다 비교적 오래 걸리는 반면, 랜덤 시간 안에 다른 연결들이 네트워크 자원을 해제할 수 있는 확률이 있기 때문에 라우팅 연결 실패율이 IRR보다는 비교적 적다.

3.2 성능평가 항목

재라우팅 기법들의 성능을 평가하기 위한 항목으로는 일정한 네트워크 자원(대역폭 등)으로 수용 가능한 연결 수를 의미하는 라우팅 성공율(Routing Success Rate)과 서비스의 특징에 맞게 신속한 라우팅 연결을 위한 라우팅 시간(Routing Time) 등으로 분류할 수 있다. 먼저 라우팅 성공율은 서비스를 요청한 총 연결 수에 대해 라우팅 성공으로 인해 서비스 가능한 연결 수로 정의할 수 있다. 여기서 라우팅 성공은 정상적인 라우팅 연결 성공과 재라우팅 연결 성공을 모두 포함하고 있으며, 재라우팅 기법에 따라 상대적인 성능을 갖게 된다. 라우팅 시간은 서비스되는 총 연결에 대해서 평균 소요 시간을 말한다. 이 역시 라우팅에 소요되는 시간은 정상적인 라우팅과 재라우팅 연결 모두를 포함한다. 이 항목 역시 재라우팅 기법에 따라 상대적인 성능을 갖게 된다. 즉 LRR보다는 SRR 기법이 라우팅 성공율은 높을 수 있지만 라우팅 시간은 오래 걸리게 되며, RRR보다는 IRR 기법이 라우팅 시간에는 장점이 있지만 라우팅 성공율에 대해서는 단점을 보이고 있다.

4. 제안 모델

이번 장에서는 본 논문에서 제안하는 모델을 설명하기 위해, 먼저 사용자 서비스에 대한 분류를 수행하고 이들의 연결에 사용되는 재라우팅 기법간의 관계를 연구한다.

4.1 사용자 서비스 분류

본 논문에서 제시하는 사용자 서비스를 그 특성에 따라 분류하는 과정은 간단하다. 일단 인터넷 상에 동작하는 다양한 서비스 트래픽의 특성을 살펴보고 이들을 그 특성에 맞게 서비스 클래스로 분류하는 것이다. [9]에서는 인터넷 상의 트래픽의 종류를 8가지 클래스로 분류하고 이들에 대해 다양한 환경에서 그 특성을 살펴보고 있다. 본 논문에서는 이러한 서비스들을 총 3가지 서비스 클래스로 크게 분류하기로 한다. 단, 분류된 데이터 패킷의 표시(marking)

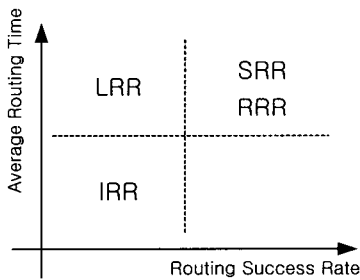
<표 1> 인터넷 서비스의 특성에 따른 클래스 분류

대 분류	인터넷 서비스	클래스 분류		
		클래스 1	클래스 2	클래스 3
Real-time streaming Transfer	Audio,video	○		
	Internet telephony	○		
	Games	○		
Real-time Block Transfer	Chatting		○	
	Image		○	
	Web pages		○	
Non real-time	E-mail(text)			○
	File Transfer			○

및 데이터 해석(interpreting)에 대한 사항은 본 논문에서 제외하기로 한다. <표 1>은 인터넷 상에 동작하는 다양한 서비스 트래픽 특성 및 종류와 분류된 클래스에 대한 관계를 보여주고 있다.

4.2 재라우팅 기법들간의 orthogonal 관계

이미 3장에서 언급한 바와 같이 재라우팅을 위한 기법들에는 크게 2가지로 분류할 수 있으며 각 분류에는 또한 각각 2가지 기법들이 현재 연구되고 있다. 또한, 이러한 기법들은 각각의 특징으로 인해 각 기법들과 비교하여 장,단점이 존재하게 된다. 더구나 각 재라우팅 기법들이 재라우팅을 시작하는 장소와 시작하는 시점과 같이 어떠한 연관성도 갖지 않는 관계이므로 이러한 기법들 중에 단순히 하나만 적용한다는 것은 사용자의 서비스를 제대로 만족시키지 못하는 상황이 발생할 수 있다. 따라서 사용자가 원하는 서비스를 만족시킬 수 있도록 하기 위해 서로 독립적인(orthogonal) 관계를 갖고 있는 재라우팅 기법들간의 연구가 필요하다. (그림 2)는 재라우팅의 각 기법들에 대해 라우팅 성공율과 평균 라우팅 시간과 같은 성능 항목에 대해서 독립적인 관계가 있음을 보여주고 있다. 또한, 각 기법들의 사용되는 성능 항목들 역시 서로 독립적인 관계에서 비교됨을 알 수 있다. 즉, LRR은 재연결을 위한 구간으로 인해 라우팅 성공율은 낮은 대신 짧은 구간으로 인해 라우팅 시간은 짧아진다. 그러므로 LRR 기법이 (그림 2)에서 좌측 상단에 놓여진다.



(그림 2) 재라우팅 기법과 성능 항목들간의 orthogonal 상관도

따라서, 본 논문에서는 (그림 2)의 상관도와 <표 1>에서 분류한 서비스를 기반으로 가장 효율적이며 보다 우수한 성능을 보일 수 있는 기법들을 제안하고자 한다. 본 논문에서 새롭게 제안하는 기법은 두 가지로 DRIT(Differential Rerouting with Immediate Timing)와 DRDT(Differential Rerouting with Differential Timing)가 그것이다. 이들은 모두 서비스 클래스별로 차별적인 기법들을 적용하는 재라우팅 기법들이다.

4.2.1 DRIT(Differential Rerouting with Immediate Timing)

사용자의 서비스를 클래스별로 재라우팅의 시작 장소를 차별화하여 적용함과 동시에 재라우팅 시작 시점에 대해서

는 모든 클래스에 대해 IRR 기법을 동일하게 적용하는 기법이다. 즉, 재라우팅 시작 장소에 대해서 클래스 1번과 클래스 2번에 대해서는 SRR을, 클래스 3번에 대해서는 LRR을 적용한다. 이는 재연결 구간 상에 많은 적절한 패스(feasible path)를 갖을 확률이 있기 때문에 각 연결에 대한 라우팅 성공율이 높은 SRR을 우선순위가 높은(time sensitive) 클래스 1번과 클래스 2번에 적용하고, 다른 서비스들보다 우선순위가 낮은(time insensitive) 클래스 3번에 대해서는 다른 클래스들의 재라우팅에 영향을 적게 미치도록 하기 위해 LRR 기법을 적용한다. 또한, 이러한 모든 클래스들의 재라우팅 시작 시점에 대해서는 IRR 기법을 동일하게 적용한다.

4.2.2 DRDT(Differential Rerouting with Differential Timing)

이 기법은 재라우팅을 시작하는 장소에 대해서는 DRIT와 동일하지만, 재라우팅을 시작하는 시점에 대해서는 각 클래스별로 차별화된 기법을 사용하는 것이다. 즉 우선 순위가 높은 클래스들(클래스 1번과 클래스 2번)에 대해서는 즉각적으로 재라우팅 기능을 수행하는 IRR 기법을, 우선순위가 낮은 클래스에 대해서는 IRR 기법을 사용하는 클래스들과 충돌(collision)이 발생하지 않도록 랜덤 시간 대기함으로써 IRR 기법을 사용한 클래스들이 가능하면 많이 네트워크 자원을 소유할 수 있도록 해주는 RRR 기법을 사용하는 기법이다. DRDT 기법은 세가지 클래스들에 대해 우선순위가 높은 클래스들(1, 2)에 대해서는 IRR 기법을 낮은 우선순위의 클래스(3)에 대해서 RRR 기법을 사용하므로, 클래스 1, 2에 대해서는 우수한 성능을 가지고 사용자 서비스를 제공할 수 있으나, 클래스 3에 대해서는 랜덤한 시간을 추가해야 하는 오버헤드에 의해 라우팅 시간이 오래 걸리게 된다. 하지만 이러한 방법을 통해 우선순위가 높은 클래스들을 더 많이 그리고 더 빨리 서비스할 수 있어 차별화된 사용자 서비스 제공에는 알맞은 기법일 수 있다.

위의 두 기법은 독립적인 관계에 있는 기법들을 차별적으로 적용함으로써 차별화된 사용자 서비스들을 각 특성에 맞게 더 많이 더 효율적으로 보내고자 하는 방안으로 제안하였다. 이들에 대한 실험과 결과는 다음 장에서 살펴보기로 하겠다.

5. 실험 및 분석

이 장에서는 본 논문에서 제안하는 두 가지 재라우팅 기법들과 기존의 재라우팅 기법들간의 성능을 파악하기 위해서 여러 가지 성능항목을 통해 그 성능을 비교 분석하게 된다.

5.1 시뮬레이션

본 논문의 시뮬레이션에서 사용하는 네트워크 범위는 노드 수를 20노드로, 회선은 전이중 방식으로 155Mbps(OC-

3)로 구성하였으며, 각 회선의 큐잉 요소(Queuing component)는 무시하였다. 또한, 회선의 지연은 대칭적이나 회선의 비용은 비대칭적으로 설정하였다. 즉, $D(u,v) = D(v,u)$ 이지만, $C(u,v) \neq C(v,u)$ 이다. 이는 회선의 길이가 같으면 지연도 같게 되지만, 회선이 비용은 전이중 방식으로 다르게 표현되기 때문이다. 본 논문에서 실험 네트워크를 생성하기 위해 NS-2(Network Simulator-2)와 함께 제공하는 GT-ITM 토폴로지 생성기[10]를 사용하여 순수 랜덤 그래프(pure random graph)를 생성하였다. 생성된 그래프의 평균 노드 정도(degree)는 3이상으로 설정했으며, 시뮬레이션 횟수는 연결 요청 횟수로 5000회를 선정하였으며, 시뮬레이션의 신뢰성을 높이기 위해서 5000회 실험을 3회 반복하여 평균값으로 실험치를 도출하였다. 다음은 본 논문 시뮬레이션을 위해 사용한 가정들이다.

- 분류된 클래스마다 요구하는 대역폭은 각각 다음과 같다.
 - 클래스 1 : 2Mbps
 - 클래스 2 : 1.5Mbps
 - 클래스 3 : 700Kbps
- 전체 연결 요청 수 중에서 각 클래스별 연결 요청 발생 수는 각각 동등하게 발생하기로 한다.
- 시뮬레이션 시 네트워크 상의 장애 노드(single-fault)는 임의의 노드를 선정하여 지속적으로 실험 한다.
- 각 연결 요청 도착간격은 $\lambda = 5$ 인 exponential distribution을 따른다.
- 연결 설정 후 데이터 전송 지속시간(duration)은 임의의 시간을 선택한다.
- 다중 송신지와 다중 목적지를 기본으로 한다. 즉, 연결 설정 요청은 20 노드에서 랜덤으로 발생할 수 있으며 수신지는 송신지를 제외한 모든 노드가 될 수 있다.
- 모든 회선의 지연시간은 일정하다. 따라서 송신지에서 목적지까지의 지연시간은 중간 경로 수로 결정된다.
- 모든 방식에서 사용되는 라우팅 테이블 갱신 주기는 모두 같다.
- 라우팅에 사용되는 매트릭스로 가용 대역폭(usable bandwidth)를 사용한다.

이러한 가정을 토대로 본 시뮬레이션에서는 기존의 재라우팅 기법인 SRR, LRR과 함께 본 논문에서 제안하는 DRIT, DRDT 등 4가지 기법에 대한 성능을 비교한다. 비교하는 성능 항목으로는 라우팅 성공율과 라우팅에 소요되는 평균 시간이며, 또한 각 클래스별로 라우팅 성공율과 라우팅 소요 평균 시간에 대한 성능을 분석한다.

5.2 성능 항목 및 실험 분석

본 논문에서 분석하는 성능항목에 대한 계산 알고리즘은 다음과 같다.

총 라우팅 성공율(a)은 총 연결 요청 수(N)에 대해 정상적인 라우팅 성공 연결 수(N_r)와 재라우팅 성공 연결

수(N_{rr})를 더한 비율로 식 (1)과 같이 표시할 수 있으며

$$a = \frac{(N_r + N_{rr})}{N} \times 100 \quad (1)$$

정상적인 라우팅 총 성공율(a_r) 및 재라우팅 총 성공율(a_{rr})은 식 (2)와 같다.

$$a_r = \frac{N_r}{N} \times 100, \quad a_{rr} = \frac{N_{rr}}{N} \times 100 \quad (2)$$

클래스별($c(i)$) 라우팅 성공율($a_{c(i)}$)은 라우팅에 성공한 총 연결 수($N_r + N_{rr}$)에 대해 클래스별로 라우팅 성공한 연결 수($S_r(c(i)) + S_{rr}(c(i))$)에 대한 비율로 식 (3)과 같이 표현할 수 있다. 여기서, $S_r(c(i))$ 는 클래스별 정상적인 라우팅에 성공한 총 연결 수이며, $S_{rr}(c(i))$ 는 클래스별 재라우팅에 성공한 총 연결 수로 표현한다.

$$a_{c(i)} = \frac{(S_r(c(i)) + S_{rr}(c(i)))}{(N_r + N_{rr})} \times 100 \quad (3)$$

또한, 클래스별 정상적인 라우팅 총 성공율($a^r_{c(i)}$)과 재라우팅 총 성공율($a^{rr}_{c(i)}$)은 식 (4)와 같이 표현할 수 있다.

$$a^r_{c(i)} = \frac{S_r(c(i))}{(N_r + N_{rr})} \times 100, \quad a^{rr}_{c(i)} = \frac{S_{rr}(c(i))}{(N_r + N_{rr})} \times 100 \quad (4)$$

다음으로, 성공한 연결에 대한 평균 라우팅 시간($a(R_t)$)은 라우팅에 성공한 총 연결 수($N_r + N_{rr}$)에 대한 성공한 정상적인 라우팅 시간(t_r)과 성공한 재라우팅 시간(t_{rr})을 더한 값으로 식 (5)와 같이 나타내며,

$$a(R_t) = \frac{(t_r + t_{rr})}{(N_r + N_{rr})} \quad (5)$$

정상적 라우팅 평균시간($a_r(R_t)$) 재라우팅 평균시간($a_{rr}(R_t)$)은 식 (6)과 같다.

$$a_r(R_t) = \frac{t_r}{(N_r + N_{rr})}, \quad a_{rr}(R_t) = \frac{t_{rr}}{(N_r + N_{rr})} \quad (6)$$

마지막으로 클래스별 평균 라우팅 시간($a(R_t, c(i))$)은 성공한 총 연결에 사용된 총 라우팅 시간($N_r + N_{rr}$)에 대한 클래스별 라우팅 시간($t_r, c(i) + t_{rr}, c(i)$)으로 식 (7)과 같이 나타낼 수 있다. 여기서 클래스별 정상적인 라우팅에 성공한 시간을 $t_r, c(i)$ 로 하며, 클래스별 재라우팅에 성공한 시간을 $t_{rr}, c(i)$ 로 정의한다.

$$a(R_t, c(i)) = \frac{(t_r, c(i) + t_{rr}, c(i))}{(N_r + N_{rr})} \quad (7)$$

클래스별 정상적 라우팅 평균시간($a^r(R_t, c(i))$)과 재라

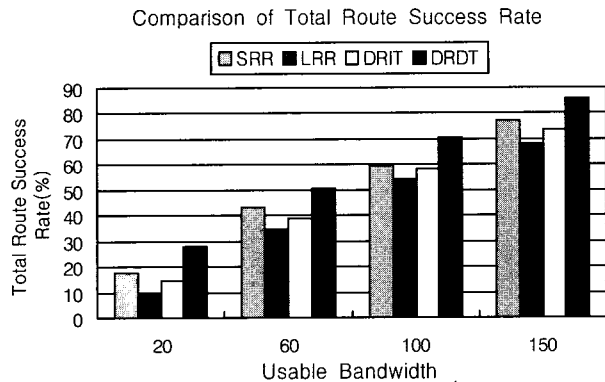
우팅 평균시간($a''(R_i(c(i)))$)은 식 (8)과 같다.

$$a''(R_i(c(i))) = \frac{t_r c(i)}{(N_r + N_{rr})}, \quad a'''(R_i(c(i))) = \frac{t_{rr} c(i)}{(N_r + N_{rr})} \quad (8)$$

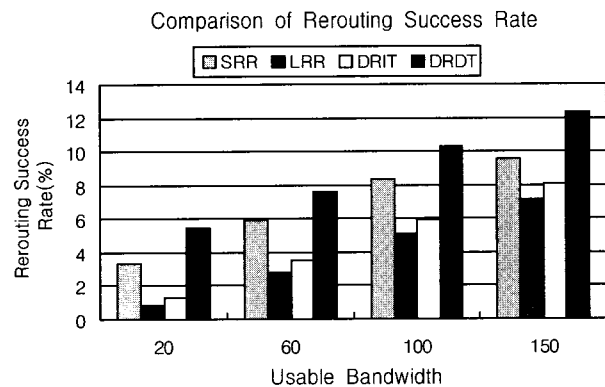
이와 같은 성능항목 계산 알고리즘을 통해 본 논문에서는 다음과 같은 4가지 시나리오를 기반으로 각 기법의 성능을 분석하기로 한다.

5.2.1 시나리오 I : 각 기법들의 총 라우팅 성공률(Routing Success Rate) 분석

시나리오 I은 식 (1), 식 (2)의 방식을 사용하여 SRR, LRR, DRIT, DRDT등 4가지 기법들에 대해 가용 대역폭 상의 총 라우팅 성공율을 분석하는 것이다. 이러한 총 라우팅 성공율은 정상적인 라우팅 성공율과 재라우팅의 성공율을 포함한 것이다. (그림 3)에서 보는 바와 같이 각 서비스를 차별화하고 이를 차별적으로 재라우팅 기법을 적용한 DRDT 기법이 가장 좋은 성능을 보이고 있으며, 그 다음으로는 SRR, DRIT, LRR 순으로 성능을 보이고 있다. 물론 정상적인 라우팅 성공율은 모두 비슷한 성능을 보이겠지만 (그림 4)에서와 같이 재라우팅 성공율은 각 기법에 따라 큰 차이를 보이고 있다. 이는 재라우팅 시 우선순위가 높은 클래스에 보다 빠르게(IRR) 그리고 보다 많은 패스가 존재하는 기법(SRR)들을 각각 적용한 DRDT가, 그 다음은 모든



(그림 3) 각 기법들에 대한 총 라우팅 성공률(α)



(그림 4) 각 기법들에 대한 재라우팅 성공율(α_{rr})

클래스를 동일한 기법(보다 많은 패스가 존재할 수 있는 기법)으로 적용한 SRR이, 우선순위에 따라 재라우팅 시작 위치를 달리하지만 동일한 시작 시점으로 인해 각 연결이 동시에 같은 네트워크 자원의 요청으로 인해 실패할 확률이 높은 DRIT가, 마지막으로 재라우팅 가능한 경로가 SRR 보다 적고, 또한 동시 요청으로 인해 실패 확률이 높은 LRR 순으로 성능을 보이고 있다.

5.2.2 시나리오 II : 클래스별 총 라우팅 성공율 분석

시나리오 II은 식 (3), 식 (4)의 방식을 사용하여 SRR, DRIT, DRDT 기법에 대해 가용 대역폭 상에 클래스별 총 라우팅 성공율과 재라우팅 성공율을 분석하는 것이다. 시나리오 I의 결과에서 보는 바와 같이 기존의 SRR 기법이 LRR보다 좋은 성능을 보이고 있으므로 LRR에 대한 분석은 불필요한 것이다. <표 2>, <표 3>, <표 4>에서 보는 값들은 시나리오 I에서 사용한 각 가용 대역폭 상에 성공한 총 라우팅 요청 수를 기반으로 SRR, DRIT와 DRDT 기법에 대한 각 클래스별 총 라우팅 성공율과 재라우팅 성공율을 보여 주고 있다. 따라서, 표의 값들은 가용 대역폭에 대해서 각 클래스별로 값을 비교하는 것이 옳은 방법일 것이다. SRR 기법에 대한 성능을 보이고 있는 <표 2>을 살펴보면 각 클래스별로 고른 분포의 라우팅 성공율(Total)과 재라우팅 성공율(Rerouting)을 보이고 있다. 이것은 SRR이 트래픽을 구분하지 않고 동일한 방법으로 재라우팅하기 때문이다. 이에 비해 트래픽을 클래스로 분류하여 각 클래스별로 재라우팅 시작 위치와 시작 시점을 달리 적용한 DRIT와 DRDT에 대한 결과를 살펴보면 <표 3>와 <표 4>과 같다. 클래스별로 IRR과 RRR을 차별적으로 적용한 DRDT 기법이, 모든 클래스에 대해 재라우팅 시작 시점으로 IRR 기법을 적용한 DRIT 기법보다 우선순위가 높은 클래스에 대해 보다 높은 성능을 보여주고 있다. 이들의 결과를 분석해 보면, DRIT 기법이 클래스 1,2에 대해서는 SRR 기법을 사용하고 클래스 3에 대해서는 LRR 기법을 적용한다. LRR 기법을 사용하는 클래스 3이 낮은 성능을 보이고 있지만, 클래스 1,2가 SRR 기법을 동일하게 적용함으로써 인해 클래스 1,2 또한 3과의 연결 요청으로 인해 라우팅 실패 확률이 생길 수 있어 성능 면에서 클래스별로 큰 차이는 보이지 않고 있다. 하지만 <표 4>의 DRDT 기법을 살펴보면 재라우팅 시작 시점을 달리 적용하여 우선순위가 높은 클래스(1, 2)에 대해서는 IRR 기법과 낮은 우선순위에 대한 클래스(3)에 대해서는 RRR 기법을 적용함으로써 높은 우선순위의 클래스들에 대해서 라우팅할 수 있는 기회를 더 많이 부여하기 때문에 우선순위가 높은 클래스들이 낮은 우선순위의 클래스보다 성능 면에서 우수함을 보이고 있다. 또한, 클래스 3에 대해서 SRR 기법이 다른 기법보다 라우팅과 재라우팅 성공율이 높은 이유는 보다 많은 네트워크 자원을 찾아 라우팅/재라우팅 할 수 있는 확률이 높으며 또한, 시간적인 면에서 IRR 기법이 적용되기 때문에 높은 성능을 보이고 있다.

〈표 2〉 SRR기법의 클래스별 라우팅 성공율($\alpha_{c(i)}$) 및 재라우팅 성공율($\alpha^{rr}_{c(i)}$)

(단위 : %)

가용 대역폭	클래스 1		클래스 2		클래스 3	
	Rerouting	Total	Rerouting	Total	Rerouting	Total
20	3.59	34.28	3.12	33.58	2.57	32.14
60	3.00	33.49	3.24	34.21	2.98	32.3
100	2.98	33.67	2.58	32.59	3.01	33.74
150	3.01	33.21	3.21	34.02	2.67	32.77

〈표 3〉 DRIT기법의 클래스별 총 라우팅 성공율($\alpha_{c(i)}$) 및 재라우팅 성공율($\alpha^{rr}_{c(i)}$)

(단위 : %)

가용 대역폭	클래스 1		클래스 2		클래스 3	
	Rerouting	Total	Rerouting	Total	Rerouting	Total
20	4.2	35.58	4.12	34.56	3.32	29.84
60	4.54	34.94	4.32	34.23	3.21	30.81
100	5.1	35.07	4.82	33.93	2.05	30.93
150	4.83	35.35	4.91	34.2	3.82	30.2

〈표 4〉 DRDT기법의 클래스별 라우팅 성공율($\alpha_{c(i)}$) 및 재라우팅 성공율($\alpha^{rr}_{c(i)}$)

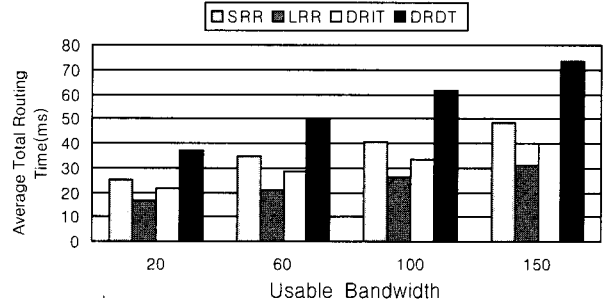
(단위 : %)

가용 대역폭	클래스 1		클래스 2		클래스 3	
	Rerouting	Total	Rerouting	Total	Rerouting	Total
20	7.58	39.98	6.12	37.59	1.25	22.43
60	6.98	38.24	5.68	37.17	1.53	24.59
100	7.26	38.46	6.59	37.79	0.98	23.75
150	8.02	39.24	6.97	37.28	1.34	23.48

5.2.3 시나리오 III : 각 기법들 상의 각 연결에 대한 라우팅 평균시간 분석

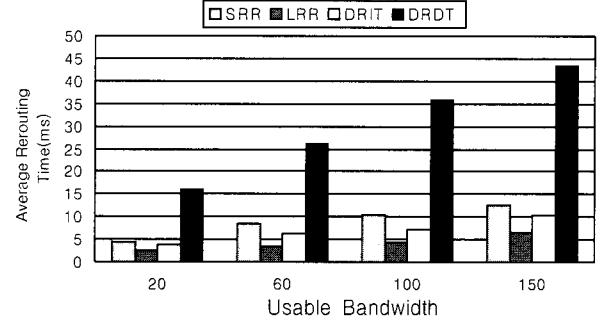
시나리오 III은 식 (5), 식 (6)의 방식을 사용하여 SRR, LRR, DRIT, DRDT 등 4가지 기법들 상에서 각 연결에 대해 가용 대역폭 상의 라우팅 평균시간을 분석하는 것이다. 라우팅 평균시간은 정상적인 라우팅 평균시간과 재라우팅 평균시간을 포함한 것이다. (그림 5)와 (그림 6)에서 보는 바와 같이 라우팅 시간 측면에서는 LRR 기법이 보다 좋은 성능을 보이고 있으며, 그 다음으로는 DRIT, SRR, DRDT 순으로 성능을 보이고 있다. 이는 시나리오 I의 라우팅 성공율과 관련이 있기도 하지만, 각 기법이 가지고 있는 특징으로 인한 성능 차이이기도 하다. 즉, 모든 서비스에 대해서 동일하게 LRR 기법(보다 짧은 재라우팅 구간을 갖는 기법)을 사용할 경우 연결설정이 성공한 각 연결에 대한 라우팅 평균시간이 가장 짧아 좋은 성능을 보이게 된다. 다음으로는 우선순위별로 SRR과 LRR 기법을 달리 적용하는 DRIT가, 세 번째로는 모든 서비스에 대해 동일하게 SRR 기법(보다 긴 재라우팅 구간을 갖는 기법)을 사용할 경우가, 마지막으로 우선순위별로 SRR과 LRR을 적용하고, 또한 우선순위가 높은 클래스와 충돌 가능성을 줄이기 위해 우선순위가 낮은 클래스에 대해 RRR 기법(랜덤시간이 추가 소요되는 기법)을 적용하는 DRDT 순으로 성능을 보이고 있다.

Comparison of Average Total Routing Time



(그림 5) 각 연결에 대한 총 라우팅 평균시간($a(R_i)$)

Comparison of Average Rerouting Time



(그림 6) 각 연결에 대한 재라우팅 평균시간($a_{rr}(R_i)$)

5.2.4 시나리오 IV : 클래스별 각 연결에 대한 라우팅 평균시간 분석

시나리오 IV은 식 (7), 식 (8)의 방식을 사용하여 LRR, DRIT, DRDT 등 4가지 기법들 상에서 각 연결에 대해 가용 대역폭 상의 총 라우팅 평균시간과 재라우팅 시 평균시간을 분석하는 것이다. 시나리오 III의 결과에서 알 수 있듯이 라우팅 시간 측면에서는 LRR 기법이 가장 좋은 성능을 보이고 있어 여기서는 LRR과 DRIT 그리고 DRDT 기법에 대한 성능을 비교 분석하기로 한다. 또한, 시나리오 II에서 보는 바와 같이 가용 대역폭 당 라우팅 성공율을 기반으로 시나리오 IV와 같은 성능을 도출할 수 있다. 결과를 분석하여 보면, <표 5>의 LRR 기법은 모든 트래픽을 동일하게 취급하여 전송하기 때문에 클래스별로 균등한 성능을 보이고 있다. 또한, <표 6>의 DRIT의 성능은 라우팅 시간 측면에서 LRR 기법을 사용하는 클래스 3이 우수한 성능을 보이고 있는 것처럼 생각되나, 시나리오 II의 결과에서 보듯이 라우팅 성공율은 저조한 성능을 보이고 있다. 마지막으로 <표 7>의 DRDT 기법은 세가지 클래스들에 대해 우선순위가 높은 클래스들(1, 2)에 대해서는 LRR 기법을 낮은 우선순위의 클래스(3)에 대해서는 RRR 기법을 사용하므로, 클래스 1, 2에 대해서는 우수한 성능을 가지고 사용자 서비스를 제공할 수 있으나, 클래스 3에 대해서는 랜덤한 시간을 추가해야 하는 오버헤드에 의해 라우팅 시간이 오래 걸리게 된다. 하지만 이러한 방법을 통해 우선순위가 높은 클래스들을 더 많이 그리고 더 빨리 서비스할 수 있어 차별화된 사용자 서비스 제공에는 알맞은 기법일 수 있다.

〈표 5〉 LRR기법의 클래스별 총 라우팅 평균시간($a(R; c(i))$) 및 재라우팅 평균시간($a^r(R; c(i))$) (단위 : ms)

가용 대역폭	클래스 1		클래스 2		클래스 3	
	Rerouting	Total	Rerouting	Total	Rerouting	Total
20	1.5	15.63	2.79	16.01	3.1	16.63
60	2.56	18.77	4.24	20.98	3.42	19.04
100	2.97	24.55	3.12	25.24	4.2	26.55
150	6.5	29.9	7.27	31.74	6.97	30.63

〈표 6〉 DRIT기법의 클래스별 총 라우팅 평균시간($a(R; c(i))$) 및 재라우팅 평균시간($a^r(R; c(i))$) (단위 : ms)

가용 대역폭	클래스 1		클래스 2		클래스 3	
	Rerouting	Total	Rerouting	Total	Rerouting	Total
20	4.21	25.23	3.89	24.1	2.36	16.72
60	8.15	32.24	8.36	33.92	4.68	20.12
100	10.74	37.12	11.26	38.72	5.75	25.14
150	12.78	46.2	13.34	45.8	6.98	29.5

〈표 7〉 DRDT기법의 클래스별 총 라우팅 평균시간($a(R; c(i))$) 및 재라우팅 평균시간($a^r(R; c(i))$) (단위 : ms)

가용 대역폭	클래스 1		클래스 2		클래스 3	
	Rerouting	Total	Rerouting	Total	Rerouting	Total
20	3.21	21.73	4.17	20.42	41.2	69.21
60	6.15	25.36	6.36	25.63	65.75	95.36
100	8.74	33.76	9.18	32.12	89.54	114.2
150	9.47	39.12	10.13	41.72	110.5	140.2

6. 결론 및 추후 연구 방향

본 논문은 사용자 서비스를 차별적으로 서비스하기 위해 기존의 재라우팅 기법들의 개념을 정립하고 이들간의 orthogonal한 관계를 연구함으로써 보다 많은 서비스 그리고, 보다 빠른 서비스를 제공하기 위한 재라우팅 기법을 연구한다. 이러한 재라우팅 기법은 재라우팅 시작 위치와 시작 시점에 따른 기법들을 포함한다. 또한, 재라우팅 기법이 수행되는 환경으로는 기존의 IP 네트워크 상에서 노드나 회선의 장애로 인해 라우팅 연결을 수행하지 못하는 경우로 정의하였다. 본 논문에서는 먼저, 사용자 서비스를 우선순위에 맞게 분류하고 이를 3가지 클래스로 정의하였으며, 두 번째는 기존의 재라우팅 기법들의 장단점을 파악하고 이들간의 상호 보완적인 관계를 연구함으로써 사용자의 우선순위에 맞게 재라우팅 기능을 수행하는 새로운 DRIT와 DRDT 재라우팅 기법을 제안하였고, 또한 각 기법들을 비교 분석하기 위해서 성능 항목에 대한 분석 방법을 제시하였다. 마지막으로 이들간의 성능을 분석하기 위해 4가지 시나리오를 기반으로 시뮬레이션하여 결과를 도출하였다. 결론적으로 우선순위에 따른 사용자 서비스를 올바르게 제공하기 위해서는 각 기법들에 대해 재라우팅 시작 위치와 시작 시점을 차별적으로 적용한 DRDT 기법이 다른 기법들에 비해 우수한 성능을 보이고 있다.

추후 연구과제로는 재라우팅 기법을 수행하는 경우를 네트워크에 대한 물리적인 장애와 더불어 네트워크 자원 등

과 같은 논리적인 장애에 의한 재라우팅 기법과 이를 적용하기 위해 기존의 서비스되어지는 사용자의 연결에 대한 처리 등에 관한 기법을 추가로 연구해야 할 것이다. 그래야만 물리적, 논리적 장애가 발생할 수 있는 네트워크 상에서도 사용자 서비스를 올바른 제공할 수 있을 것이다.

참고 문헌

- [1] Z. Wang and J. Crowcroft, "QoS routing for supporting multimedia applications," IEEE Journal Select Areas Communication, Sept., 1996.
- [2] Donna Ghosh, Venkatesh Sarangan, and Raj Acharya, "Quality-of-Service Routing in IP Networks," IEEE Transactions on Multimedia, Vol.3, No.2, June, 2001.
- [3] Peyravian, M, "Providing different levels of network availability in high-speed networks," Global Telecommunications Conference, 1994. GLOBECOM '94, Communications : IEEE, Vol.2, pp.941-945, 1994.
- [4] D. Black et al., "An architecture for differentiated services," Internet Draft <<draft-ietf-diffserv-arch-00.txt>>, May., 1998.
- [5] C. Parris and A. Banerjee, "An investigation into fault recovery in guaranteed performance service connections," in Proc, SUPERCOMM/Int Conf. Communication '94, New Orleans, LA, pp.175-181.
- [6] Anindo Banerjee, "Fault Recovery for Guaranteed Performance Communications Connections," IEEE/ACM Trans on Networking, Vol.7, No.5, pp.653-668, Oct., 1999.
- [7] R. Doverspike, "A multi-layered model for survivability in intra-LATA transport networks," in Proc, IEEE GLOBECOM '91, Phoenix, AZ, pp.2095-2031.
- [8] L. Zhang et al., "RSVP : A new resource reservation protocol," IEEE Commu. Mag., Vol.31, pp.8-18, Sept., 1993.
- [9] Karam, M. J ; Tobagi, F. A., "On traffic types and service classes in the Internet," in Proc, IEEE Globecom'00 Tel Aviv, Israel, Vol.1, pp.548-554.
- [10] <http://www.isi.edu/nsnm/ns/ns-topogen.html>.



한정수

e-mail : jshan@songgang.skku.ac.kr
 1997년 성균관대학교 공과대학 정보공학과 졸업(학사)
 1999년 성균관대학교 전기전자 및 컴퓨터 공학부 석사 졸업
 현재 성균관대학교 전기전자 및 컴퓨터 공학부 박사수로

관심분야 : 네트워크 관리, 트래픽 분석, 인터넷 QoS, QoS 라우팅



정진욱

e-mail : jwchung@songgang.skku.ac.kr
 1974년 성균관대학교 전기공학과 학사
 1979년 성균관대학교 대학원 전자공학과 석사
 1991년 서울대학교 대학원 계산통계학과 박사
 1982년~1985년 한국과학기술 연구소 실장

1981년~1982년 Racal Milgo Co. 객원연구원
 1985년~현재 성균관대학교 전기전자 및 컴퓨터공학부 교수
 2001년~현재 한국정보처리학회 회장
 관심분야 : 컴퓨터 네트워크, 네트워크 관리, 네트워크 보안