

P2P 기반의 사용자 주문형 스트리밍 서비스를 위한 세그먼트 그룹 관리 구조

이종득¹, 정택원^{1*}

¹전북대학교 응용시스템공학부

Group Management Structure of Segments for P2P-based On Demand Streaming Services

Chong-Deuk Lee¹ and Taegwon Jeong^{1*}

¹Division of Applied System Engineering, Chonbuk National Univ.

요 약 최근에 P2P 기반의 분산 환경에서 동적인 분산 콘텐츠 서비스를 제공하기 위한 많은 연구가 수행되고 있다. 그러나 콘텐츠 객체들의 분산 서비스를 위해서는 QoS 문제와 세그먼트의 동적 관리 문제가 발생되고 있다. 본 논문에서는 P2P 기반의 분산 콘텐츠 서비스를 위한 새로운 세그먼트 관리기법을 제안한다. 제안된 기법은 SGM(Segment Group Manager)를 통하여 그룹을 관리하게 되며, 이 구조는 P2P상에서 서비스가 수행되는 세그먼트들을 결정하여 스트리밍을 효율적으로 관리하게 된다. 이때 같은 그룹으로 구성된 세그먼트들은 비슷한 네트워크 조건하에서 효율적인 QoS를 수행하기 위하여 상호 협력한다. 이러한 기능은 거리기반 관리구조와 관계성 기반 관리구조에 의해서 수행된다. 거리기반 관리구조는 검색효율을 개선하기 위한 관리 기법이고, 관계성 기반 관리구조는 세그먼트의 서비스율을 개선하기 위한 기법이다. 제안된 기법의 시뮬레이션 결과 평균 검색시간이 TPD(Truncated Pareto Distribution)기법에 비해서 8%, 랜덤 기법에 비해서는 30%가 개선되었고, 세그먼트의 서비스율은 10% ~ 30% 개선되었음을 알 수 있다.

Abstract There is a lot of recent research to provide services of dynamic distributed contents in P2P-based distributed environment. Distributed services of contents objects, however, have problems in QoS and dynamic management of segments. This paper proposed a new segment management method for the service of P2P-based distributed contents. The proposed method manages groups by the SGM(Segment Group Manager). The SGM manages streaming efficiently by the grouping of segments to be served in P2P environment. The segments in the same group cooperate for improved QoS using the management structure based on the distance and relationship. The distance-based management structure is for the improvement of retrieval efficiency while the relationship-based management structure is for the improvement of service ratio. The simulation results of the proposed method showed improvements in average transmission efficiency and average service rate. The improvement is 8% - 30% in average transmission efficiency and 10% - 30% in average service rate.

Key Words : P2P, Segment, SGM, Streaming

1. 서론

최근에 P2P 기반의 분산 응용 도메인 환경에서 사용자 주문형 서비스가 일반화 되면서 분산 네트워크에 걸리는 부하가 증가하게 되었고 이로 인하여 QoS의 성능이

떨어지는 문제가 발생되고 있다. 분산 환경에서 이러한 QoS 문제를 개선하기 위하여 프록시 서버(proxy server) 기법을 이용하고 있다. 프록시 서버 기법은 네트워크의 가장 자리에 위치해 있으면서 클라이언트로 향하는 데이터를 캐싱하여 원거리의 클라이언트와 스트리밍 서버 사

*교신저자 : 정택원(ttwjeong@yahoo.co.kr)

접수일 09년 05월 25일

수정일 (1차 09년 06월 30일, 2차 09년 07월 09일)

게재확정일 09년 07월 22일

이의 거리로 인한 문제를 해결하기 위한 기법이다[1-3]. 이 기법은 분산 서비스 환경의 성능을 개선하기 위한 기법으로 사용되고 있지만 대역폭과 서비스 스트림 관리 문제가 발생되고 있다. 또한 분산 환경의 P2P 기반 사용자 주문형 서비스를 위해서는 끊임 없는 스트림 서비스와 지터(jitter) 문제 최소화 및 QoS가 보장되어야 한다 [1]. 특히 P2P 기반의 분산 응용 서비스 환경은 사용자 서비스 요구를 다양하게 제공하고 있으며 서비스 품질에 대한 욕구를 증가시키고 있다. 이로 인하여 기존의 P2P 기반에서 제공되던 멀티미디어 서비스는 무선네트워크로 영역이 점차 확대되어 가고 있다.

프록시 서버에서 QoS를 보장하기 위해서는 스트림 데이터를 세그먼트로 관리해야 하며, 블록 세그먼트 관리를 위해 원-패스 마이닝(one-pass mining) 기법[4]이 사용되고 있다. 원-패스 마이닝 기법은 실시간으로 수행되는 세그먼트의 스트림 변화를 인식하기 어려우며, 이것은 데이터 스트리밍이 시간에 따라 동기화가 수행되기 때문이다. 또한 이 기법은 상황에 맞는 세그먼트들을 실시간으로 분류하지 못한다는 문제점이 발생되고 있으며, 이것은 분류구조를 단순히 스트림 상에서 발생하는 문제뿐만 인식하고 있기 때문이다. 원-패스 마이닝 기법에서 발생하는 문제를 개선하기 위해 점진적 갱신기법(Incremental updating)[5]이 제안 되었으며 이 기법은 세그먼트 스트리밍이 수행될 때 슬라이딩 윈도우(sliding window) 문제로 인하여 스트림 서비스의 정확도가 떨어지는 문제점이 발생하고 있다. 세그먼트 스트리밍을 위한 효율적인 관리는 동기화가 수행될 때 이루어져야 하며 이에 대한 관리 테스트 스트리밍 과정은 시스템이 구축될 때 수행되어야 한다. 이러한 기능 수행을 위해[6,7]에서는 세그먼트 분류 기법을 제안하였으며 이 기법은 세그먼트 관리를 위한 구조를 학습 분류과정과 비학습 분류과정으로 구분하여 분류하였다. 그리고 학습 분류는 다시 정적 학습 분류와 동적 학습 분류로 구분하여 분류하였다. 정적분류학습은 일정시간 간격에 따라 스트리밍이 수행되는 분류 학습으로서 분류의 정확도는 증가되지만 적응성이 떨어지는 문제가 발생하고 있다. 그리고 동적 분류학습은 분류의 정확도는 떨어지지만 적응성이 좋은 장점을 가지고 있다. 스트리밍을 수행하고 관리하는 과정에서 버스트 시간이 짧은 세그먼트 스트리밍과 버스트 시간이 긴 세그먼트 스트리밍은 분류 및 관리 구조에 중요한 영향을 미치게 되며, 또한 QoS에도 영향을 미치게 된다.

따라서 본 논문에서는 P2P 네트워크 기반의 사용자 주문형 서비스를 위하여 스트리밍 서비스 오버헤드를 줄이고 QoS를 향상시키기 위한 새로운 관리 프레임구조를 제안한다. 제안된 관리 프레임구조는 P2P 네트워크 환경

에서 피어(Peer)가 다른 피어 그룹과 데이터를 공유하며 이웃 피어나 디렉토리 서버에 질의를 통하여 원하는 세그먼트를 탐색하게 된다. 원하는 세그먼트의 위치를 탐색하게 되면 피어는 다른 피어의 프록시 서버로부터 세그먼트를 탐색하여 그룹 구조를 인식하게 된다. 제안된 관리 구조에서 각각의 세그먼트들은 다른 세그먼트들과의 관계성을 파악하게 되며 SGM(Segment Group Manager)을 수행한다. SGM의 세그먼트들은 서로 유사한 특성을 가지도록 구조화하며, 구조화를 통하여 세그먼트 스트리밍 서비스를 위한 QoS 요구사항들이 만족되게 된다. 제안된 그룹 관리 구조는 시스템 성능을 개선하고 P2P 기반의 멀티미디어 분산 서비스율을 증가시키며, 거리기반 관리구조와 관계성 기반 관리구조를 통하여 세그먼트들을 관리하게 된다.

본 논문의 구성은 다음과 같다. 2장에서 관련 연구에 대하여 고찰하고, 3장에서는 제안된 SGM 구조를 기술하고, 4장에서는 제안된 SGM에 대한 성능 평가에 대하여 논하고, 끝으로 결론을 기술한다.

2. 관련연구

분산 응용 환경에서 주문형 스트리밍 서비스는 프록시 캐싱 서버의 성능을 향상시키는 중요한 기법으로 이와 관련된 많은 연구들이 수행되고 있다[8-10]. P2P 기반의 주문형 스트리밍을 위한 서비스 기법으로는 CMD (Centralized Multimedia Distribution) 분산 서비스 기법, CDN(Content Distribution Network) 기반의 분산 서비스 기법, 프록시 캐싱 분산서비스 기법 등이 일반적으로 많이 사용되고 있다[4].

CMD 분산 서비스 기법에서는 스토리지와 I/O 용량을 확대하여 중앙 서버의 서비스 기능을 개선한다. 이 기법은 기존의 웹 기반 분산 서비스 기법에서 주로 이용되는데, 분산 환경에서 QoS에 중요한 영향을 미치는 네트워크 병목 현상을 줄이는 데 어려움이 있다.

이러한 CMD의 문제점을 개선하기 위하여 Akamai[11]가 프록시 서버들을 이용하여 네트워크의 에지들을 구성하는 CDN 기반의 분산 서비스 기법을 제안하였다. 이 기법은 서버들을 제어하여 사용자 요구사항들을 효율적으로 재배치하기 위한 기법으로서 사용자의 요구사항을 반영하여 응답시간을 줄이고 서버들 사이의 로드(Load) 균형을 맞춘다. 이때 서버와 네트워크 링크를 포함한 인프라 구조는 효율적인 성능을 보장하고 있다. 그러나 이 기법은 로드 균형으로 인한 효율적인 분산 서비스를 제공하는 데에 많은 문제점이 제기되고 있다. 이

기법에서 에지 서버의 용량은 멀티미디어 서비스를 제공 하기에는 충분하지 못하며 특히 스트리밍 미디어 서비스에는 많은 문제점이 발생되고 있다. CDN 플랫폼은 수많은 프록시 서버들이 위치해 있으므로 이 위치들을 파악하여 서버상의 스트리밍 데이터를 파악하여 관리하기란 현실적으로 어려움이 있다.

proxy 캐싱 기법은 자주 사용된 데이터를 캐싱하여 멀티미디어 분산 서비스 시스템에서 발생하는 네트워크 병목 현상을 줄이기 위한 기법으로서 클라이언트들은 대역폭을 소모하지 않고 에지 서버로부터 스트리밍 서비스를 수행한다.

CDN 플랫폼에서 내용분산 서비스 기법은 내용 전송 네트워크 서비스 기법으로서 서버 중심 서비스 기법이다. 이 기법에서 최초의 서버는 복제되며 네트워크 공간상에서 지역적 또는 원격으로 배치되게 된다. 캐싱기법과 CDN 기법은 멀티미디어 데이터를 전송하는데 있어서 용량 문제가 제기되고 있다. Napster[12], Gnutella [13], FreeNet[14], CenterSpan[15]와 같은 P2P 시스템은 용량 문제가 아닌 효율적인 QoS를 제공할 수 없다는 문제점을 가지고 있다.

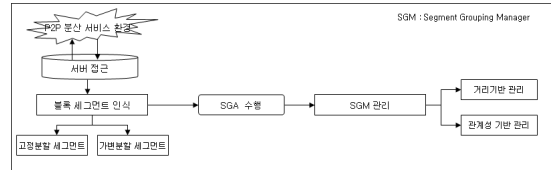
3. 제안된 SGM(Segment Group Manager) 구조

QoS 지향의 주문형 스트리밍 서비스를 위해서는 분산 서비스 환경에서 서로 관련 있는 세그먼트들을 관리하기 위한 구조가 요구되며, 세그먼트 그룹 관리 구조는 분산 환경에서 이웃한 호스트와 피어(peer)들을 응용 그룹으로 클러스터하여 관리하는 기능을 수행한다. 응용 그룹안의 세그먼트들은 서로 접근 관련성을 가지고 있기 때문에 세그먼트 스트리밍 전송에 대한 QoS를 만족하게 된다. 일반적으로 분산 환경에서 같은 게이트웨이를 사용하는 호스트들은 하나의 그룹으로 클러스터가 가능하다.

P2P 기반의 주문형 스트리밍 서비스에서 중요한 기능 중에 하나는 스트리밍이 수행되는 자원의 관리이며, 이 기능은 시스템 전송 성능을 개선하여 보다 높은 서비스율을 보장하게 된다. 따라서 이와 같은 기능을 수행하기 위한 제안된 구조는 그림 1과 같으며, 이 구조는 P2P분산 환경에서 서버 접근을 통하여 블록 세그먼트들을 인식하고 인식된 블록 세그먼트들에 대해서 SGA를 수행한다.

SGA를 수행한 후 SGM관리를 하게 되며, 이때 거리기반관리와 관계성 기반 관리가 수행된다. 본 논문에서는 거리기반 관리와 관계성기반 관리가 효율적으로 수행되기 위해 거리기반관리는 그룹에서 세그먼트 블록과 세그

먼트_{s_i} 간의 평균 접근거리 acc_d(s_i)에 의해 수행되며, 관계성 기반 관리는 알파라벨관계, 유사관계, 그리고 호환관계에 의해 수행된다. 이들 관계성은 블록에서 세그먼트들의 관계성을 파악하여 세그먼트들에 대한 간섭을 줄임으로서 효율적인 스트리밍을 수행하게 된다. 그러나 제안된 기법은 세그먼트들의 관계성으로 인해 세그먼트들에 병목현상이 발생할 경우 혼잡(congestion)의 문제가 발생할 수 있다.



[그림 1] 제안된 시스템 구조

3.1 블록 세그먼트 인식

서버 접근을 통해서 인식된 블록 세그먼트들은 캐싱이 수행되는 블록과 수행되지 않는 블록을 결정하게 된다. 이 과정은 프록시 지터를 해결하기 위한 자원 사용량을 최소화하여 네트워크 대역폭과 트래픽, 시간 지연과 같은 QoS를 개선하기 위한 과정이다. SGM을 수행하기 위해서는 캐싱을 위한 세그먼트 블록이 준비되어야 하고 블록 접근은 순차적이어야 한다. 그리고 SGM 구조-클라이언트 링크 대역폭은 충분히 확보되어야 하고 프록시에서의 클라이언트 미디어 스트리밍 서비스는 충돌이 발생되지 않아야 한다. 이때 SGM을 위한 각각의 블록 세그먼트들은 서버로부터 패치 되며, 패치된 세그먼트들에 대해 인코딩율과 네트워크 대역폭을 고려하여 블록 세그먼트 인식을 결정하게 된다. 서버로부터 블록 인식 과정은 식(1)의 과정을 거쳐 수행되며, 인코딩율과 대역폭의 관계를 결정하기 위한 과정은 식(2)와 같다.

$$BP(x) = \sum_{i=1}^n BSL_i - \frac{BSL(n+1) \times (Er - Bw)}{Bw} \dots(1)$$

$$\frac{\sum_{i=1}^n BSL_i - BP(x) + BSL(n+1)}{Er} \geq \frac{BSL(n+1)}{Bw} \dots(2)$$

그리고 인식된 전체 블록 세그먼트들에 대한 버퍼링을 결정하기 위한 과정은 식(3)과 같다.

$$\frac{\sum_{i=1}^n BSL_i - BP(x)}{Er} \times Bw \quad \dots\dots\dots(3)$$

식(1), (2), (3)에서 $BP(x)$ 는 임의의 블록 세그먼트들에 대한 블록분할, BSL_i 는 i 번째의 블록 세그먼트 크기, Er 은 블록 세그먼트에 대한 평균 인코딩 율, Bw 는 평균 네트워크 대역폭이다.

각 블록 분할이 결정되면 분할된 블록들에 대해서 SGA(Segment Group Aware)과정을 거쳐 SGM을 수행하기 위한 참조블록(Reference Block)을 수행한다. 참조블록은 고정분할 참조블록(Fixed Partition Reference Block, R_fP)과 가변분할 참조블록(Variable Partition Reference Block, R_vP)으로 구분하여 수행한다.

3.1.1 고정 분할 참조 블록(R_fP)

고정 분할 참조블록은 블록 세그먼트 분할을 같은 크기로 분할하는 블록구조로서 분할을 수행할 $BP(x)$ 에 대해서 인코딩율과 대역폭을 고려하며 고정 분할을 위한 참조 블록 $R_fP(x)$ 는 다음과 같이 수행한다.

(정의1)

$$R_fP(x) = BSL_i(n+1) \times T_f - \frac{Er}{Bw} \times BSL_i \text{ 이다.}$$

여기서 T_f 는 분산 서비스 환경에서 고정 분할 미디어 스트리밍을 위한 시간 동기화이다.

그리고 고정 분할 참조블록 R_fP 가 결정되면 이에 대한 버퍼 크기 $R_fP(x)(B_{uffer})$ 는 다음과 같이 결정한다.

(정의2)

$$R_fP(x)(B_{uffer}) = (T_f \times BSL_i) \times \frac{Er - Bw}{Er} \text{ 이다.}$$

3.1.2 가변 분할 참조 블록(R_vP)

가변 분할 참조블록은 블록 세그먼트 분할을 서로 다른 크기로 분할하는 블록구조로 분할을 수행할 $BP(x)$ 에 R_vP 를 고려하며 가변 분할을 위한 참조 블록 $R_vP(x)$ 는 다음과 같이 수행한다.

(정의3)

$$R_vP(x) = BSL_i(n+1) \times T_v - \frac{Er}{Bw} \times BSL_i \text{ 이다.}$$

여기서 T_v 는 분산 서비스 환경에서 가변 분할 멀티미디어 스트리밍을 위한 시간 동기화이다.

그리고 고정 분할 참조블록 $R_fP(x)$ 가 결정되면 이에 대한 버퍼 크기 $R_fP(x)(B_{uffer})$ 는 다음과 같이 결정한다.

(정의4)

$$R_fP(x)(B_{uffer}) = (T_v \times BSL_i) \times \frac{Er - Bw}{Er} \text{ 이다.}$$

$R_fP(x)$ 와 $R_vP(x)$ 에서 만일 $R_fP(x)=0$ 또는 $R_vP(x)=0$ 이면 세그먼트는 분할되지 않게 되며, 이때에는 미분할로 인하여 SGM을 수행하지 못하는 문제가 발생된다. 따라서 본 논문에서는 이러한 문제를 피하기 위해 $R_fP(x) \neq 0$ 이고, $R_vP(x) \neq 0$ 인 세그먼트 즉 $R_fP(x) > 0$ 이고, $R_vP(x) > 0$ 인 $[(Er/Bw)]^{\text{th}}$ 번째 세그먼트들에 대해서 분할을 수행하여 SGM을 수행한다.

3.2 SGA(Segment Group Aware) 수행

서버로부터 블록 세그먼트 분할 패치가 이루어지면 SGM을 위한 SGA 과정을 수행한다. SGA의 목적은 하나의 그룹 안에 있는 세그먼트들은 서로 관련성을 가지고 있기 때문에 스트리밍 QoS를 보다 효율적으로 제공할 수 있도록 하기 위해서이다. 이때 스트리밍을 위한 라우팅 오버헤드는 QoS에 중요한 영향을 미치게 되며 분산 서비스 환경에서 세그먼트가 랜덤하게 구성되면 네트워크에서 이웃한 호스트들은 실제로 관련성이 떨어지게 된다. 이와 같은 기능은 네트워크 자원을 너무 낭비하는 요소가 되며 데이터 전송 성능 또한 감소시키는 기능을 수행하게 된다. 예를 들어 P2P 기반의 분산 환경에서 임의의 주어진 위치 P에 대하여 만일 위치 P와 호스트 A 사이의 거리가 P와 호스트 B 사이의 거리가 같다면 호스트 A와 B는 같은 그룹 배치가 가능해진다. SGA가 수행되면 스트리밍 서비스 관리를 위해 그룹 관리가 수행되어야 한다. 특히 새로운 그룹이 입력되면 입력된 그룹과 그룹화를 수행하기 위하여 가장 가까운 그룹을 탐색할 수 있도록 해야 한다. 그룹 A와 이웃한 그룹들은 그룹 A와 관련 있는 그룹으로 정의되고 참조블록 R_fP 와 R_vP 를 수행한다. 이때 R_fP 와 R_vP 는 이웃한 그룹들과 메시지를 교환할 수 있으며, 그룹의 이웃들은 R_fP 와 R_vP 인지를 인식하게 된다. SGA를 위한 구조는 그룹들이 상위계층과 하위계층구조로 구성되어 있으며, 하위 계층은 하나의 그룹 안에 세그먼트들로 구성되어 있다. 그룹 안의 세그먼트들은 서로 정보를 공유하게 되며, 두 개의 서로 다른 그룹들은 최단 거리를 통하여 서로 통신을 수행하게 된다. 이처럼 SGA는 같은 네트워크상에 구성된 블록 세그먼트들의 그룹들을 인식하여 메시지 관리 및 응용 전송 시간을 줄이

는 기능을 수행한다. SGA를 수행할 때 각각의 그룹은 세그먼트들의 구성 여부를 판단하게 된다. 새로운 세그먼트가 탐색되어 인식되면 이 세그먼트는 위치 탐색기법을 이용하여 가장 가까운 그룹과 연결되거나 그룹화 기준에 따라 그 자신의 그룹을 형성하게 된다. 그룹이 구성되면 참조 포인터(Reference Pointer)를 이용하여 이웃한 그룹에 접속한다. RP는 가장 가까운 그룹을 인식하는 안내 기능 역할을 하며 이를 위해 새로운 인식할 그룹A와 먼저 통신을 수행한다. 이후 RP는 다음에 캐시될 그룹 B를 선택하고, 그룹 A는 그룹 B를 접촉하게 되고 그 자체에서 그룹 B까지의 거리를 측정하여 캐시가 수행될 그룹들을 반복해서 인식하게 된다.

3.3 SGM(Segment Group Manager) 관리

SGM 관리는 SGA 과정을 통하여 수행된다. SGA과정에서 그룹들을 인식한 후에는 이웃한 다른 그룹들이 관리 되도록 세그먼트 참조블록 사본을 만들어 둘 필요성이 있다. 세그먼트 참조 블록 사본은 특정 세그먼트에 대한 미디어 전송 품질은 좋게 한다. 그러나 전송 품질을 위한 참조 블록 사본 관리는 중요한 기능이며 이 장에서는 SGM 관리를 통한 시스템 성능 향상 기법에 대해서 살펴본다.

3.3.1 거리기반 관리

거리 기반 관리 구조는 그룹들 간의 세그먼트의 전송 효율성을 향상시키기 위한 방법이며, 그룹들 간의 거리척도 기법[3]에 기반을 둔다. 그룹G_i와 그룹G_j 사이의 거리를 d(G_i, G_j)라고 하자. 본 논문에서는 평균거리가 최소화 되도록 하기 위해 서로 다른 그룹들 간의 세그먼트 참조블록 사본이 어디에 있는지를 탐색한다. 세그먼트 참조블록 사본은 X를 이용하며, 사본 변수 x_{ij}는 다음과 같이 정의된다.

(정의5) 사본 변수 x_{i,j}의 정의

$$x_{i,j} = \begin{cases} 1, & \text{그룹 } G_i \text{에 세그먼트 } S_i \text{에 대한 참조블록 사본이 있을 때} \\ 0, & \text{otherwise} \end{cases}$$

참조블록 사본 변수가 결정되면 그룹 G_i에 대한 접근행렬 Y={y_{ij}}를 다음과 같이 정의한다.

(정의6) 접근행렬 Y={y_{ij}}

$$y_{i,j} = \begin{cases} i & \text{if } i \neq j, \\ d(G_i, G_j) = \min_s \{d(G_i, G_s), x_{i,s} = 1\} \end{cases}$$

그룹 G_j에서 세그먼트 s_i를 캐싱하는 클라이언트 경우 접근행렬 Y의 원소 y_{ij}는 세그먼트 s_i를 가지고 있는 이웃한 그룹이다. 참조 블록 사본 변수와 접근행렬 변수 Y가 결정되면 거리 기반 관리를 위한 최소화는 다음과 같이 정의된다.

$$(정의7) \text{ 최소화 : } \sum_{i=1}^n \sum_{j=1}^n \{r_i \times d(G_j, G_{y_{i,j}})\}$$

$$\text{즉 } \sum_{i=1}^n x_{i,j} \leq S_j, x_{i,j} \in \{0, 1\}, j = 1, 2, \dots, N \text{이다.}$$

위의 정의는 복제 행렬 X의 값에 대한 평균 최소 거리이며, 각 그룹의 세그먼트 참조는 각 그룹의 세그먼트 용량을 초과해서는 안 된다. 따라서 각 그룹의 세그먼트 용량을 무시하고 전체 P2P 네트워크상에서의 세그먼트 용량만을 고려한다고 가정할 때 세그먼트 수를 SN_i(Segment Number)라 하고 T는 네트워크상에서 전체 용량이라 하자. 그러면 SN_i와 T와의 관계는 다음과 같이 정의된다.

$$(정의8) \sum_{i=1}^M SN_i \leq \sum_{j=1}^N T_j = S \text{이다.}$$

그리고 그룹에서 세그먼트 블록과 세그먼트s_i 간의 평균 접근거리는 acc_d(s_i)로 표현하며, 다음과 같이 정의된다.

$$(정의9) acc_d(s_i) \propto avg_{dis}(s_i) = \frac{\delta}{\sqrt{SG_i}} \text{이다.}$$

여기서 δ는 거리상수이며, SG는 가장 가까운 거리에 위치한 세그먼트 그룹이다.

위의 정의는 캐시가 수행되는 그룹과 스트리밍 사이의 가장 가까운 거리에 있는 세그먼트 s_i 간의 평균 접근거리를 의미하며 이때 네트워크가 수행되는 모든 피어들은 분산 응용 도메인 상에서 일정하게 분산되어 있다고 가정한다. 따라서 피어 기반 환경에서 클라이언트가 요구하는 세그먼트 블록들을 효율적으로 관리하기 위한 거리 최소화화 블록 세그먼트 선택과정은 다음과 같이 정의된다.

$$(정의10) \text{ 거리 최소화 : } \sum_{i=1}^M r_i \times acc_d(C_i)$$

$$= \sum_{i=1}^M r_i \times \alpha \times BS \times SG^{-1/2} \text{이며,}$$

$$\text{블록 세그먼트 선택 : } \sum_{i=1}^M SN_i \leq S. (SN_i > 0) \text{이다.}$$

여기서 r_n 는 거리 최소화가 되도록 그룹을 선택하기 위한 radius이며, BS는 블록 세그먼트이다.

3.3.2 관계성 기반 관리

관계성 기반 관리 구조는 분산 서비스율을 향상시키기 위한 관리 기법으로서, 알파-레벨 관계, 유사관계, 호환관계 기반으로 구성된다.

1) 알파-레벨 관계

알파-레벨 관계는 $[0, 1]$ 사이의 퍼지 값에서 임의의 α ($0 \leq \alpha \leq 1$)이 되는 구간을 이용하여 관계성을 표시하며, 퍼지 집합의 원소들에 대한 소속 기준을 정의하여 사용한다. 알파 레벨-관계성은 $R=(G, s)$ 에 대하여 $\mu_{s:s} \rightarrow [0,1]$ 로 정의되며, μ_s 는 그룹 세그먼트들에 대한 퍼지 유사도 관계가 수행되는 membership 함수이다. 만일 블록 세그먼트 BS에서 임의의 세그먼트 s 를 원소로 하는 퍼지 집합에 대해서 $\alpha \in [0,1]$ 이라면 세그먼트들에 대한 퍼지 집합 s 는 다음과 같이 정의한다.

(정의11) $G=\{s \mid bs(s) \geq \mu\}$ 이다.

예를 들어 $bs(s) \geq \mu$ 를 구성하는 bs 와 s 사이의 퍼지관계가 표 1과 같다고 가정하자.

[표 1] bs 와 s 의 퍼지관계

$bs \setminus s$	s_1	s_2	s_3	s_4	s_5
bs_1	0.43	0.87	0.32	0.97	0.75

$bs \setminus s$	s_6	s_7	s_8	s_9	s_{10}
bs_1	0.12	0.06	0.49	0.54	0.65

표 1에서 $bs_1(s) \geq 0.5$ 를 만족하는 알파-레벨은 $\{s_2, s_4, s_5, s_9, s_{10}\}$ 이 되며, $bs_1(s) \geq 0.5$ 를 만족하지 않는 $\{s_1, s_3, s_6, s_7, s_8\}$ 관리 대상에서 제외되며 $bs_1(s) \geq 0.5$ 를 만족하는 알파-레벨 $\{s_2, s_4, s_5, s_9, s_{10}\}$ 에 대해서 세그먼트 관리가 수행된다. 결과적으로 이러한 세그먼트 관리는 주문형 스트리밍 서비스가 수행되어 서비스율을 향상시켜주게 된다.

2) 유사관계

유사관계는 bs 의 세그먼트 원소 s 에 대해서 퍼지 관계성을 수행하는 관계성으로서 이들은 의미적으로 내포하는 관계성이 서로 다르다. s 와 s , s 와 bs 사이의 관련 정도를 파악하기 위한 관계성은 반사관계, 대칭관계, 전이관계를 파악하여 수행되며, 임의의 세그먼트 x, y, z 에 대해

$(s_x, s_y, s_z) \in bs$ 의 조건이 만족될 때 유사관계를 수행한다.

이때 s 에 대한 퍼지집합이 만족되는 $(s_x, s_y, s_z) \in bs$ 에 대하여 $s=\{s \in S, <bs, s> \in FR(\text{Fuzzy Relation})$ 일 경우 $\exists bs$ 에 대해서 $s \mid s \in S$ 이고, $bs=\{s \in S, <s, bs>$ 일 경우 $\exists s$ 에 대해서 $s \mid bs(s) \in FR$ 이면 반사관계, 대칭관계, 전이관계를 다음과 같이 표현한다.

- ① 반사관계 : $\mu \cong (s_x, s_x)=1$
- ② 대칭관계 : $\mu \cong (s_x, s_y)=\mu \cong (s_y, s_x)$
- ③ 전이관계 : $\mu \cong (s_y, s_z) \geq \min\{\mu \cong (s_x, s_y), \mu \cong (s_y, s_x)\}$

이와 같은 관계가 성립될 때 s 와 s 사이의 퍼지 유사관계는 다음과 같이 정의한다.

(정의12) $\mu_{s_1 \cong s_2} = \max\{\min\{\mu_{s_1}(S), \mu_{s_2}(S)\}, \min\{1 - \mu_{s_1}(S), 1 - \mu_{s_2}(S)\}\}$ 이다.

여기서 $\mu_{s_1}(S)$ 는 bs_1 에서 퍼지 유사관계를 만족하는 세그먼트들이며, $\mu_{s_2}(S)$ 는 bs_2 에서 퍼지 유사관계를 만족하는 세그먼트들이다.

bs 에서 s 들 사이의 평균 퍼지 유사 관계는 다음과 같이 정의된다.

(정의13) $\mu_{i,j} = \mu_{w_i \cong w_j} = \frac{1}{N} \sum_{n=1}^{bs} \mu_{w_i \cong w_j}(bs_n)$ 이다.

여기서 $\mu_{i,j}$ 는 s_i 와 s_j 는 퍼지 평균 유사도 행렬변수이며, N 은 전체 bs 의 수를 의미한다. 그리고 $\mu_{w_i \cong w_j}$ 는 임의의 bs 에서 s_i 와 s_j 간의 퍼지 유사도이다.

예를 들어 bs_1 과 bs_2 에서의 위의 정의를 이용한 퍼지 유사도가 표 2와 같다고 가정하자.

[표 2] bs_1 과 bs_2 의 s 들 사이의 퍼지 유사도($\mu_{w_i \cong w_j}$)

$bs_1 \setminus bs_2$	s_1	s_2	s_3	s_4	s_5	s_6	s_7
s_1	1	0.32	0.78	0.45	0.84	0.55	0.92
s_2	0.32	1	0.57	0.68	0.95	0.47	0.26
s_3	0.78	0.57	1	0.62	0.35	0.48	0.81
s_4	0.45	0.68	0.62	1	0.52	0.24	0.85
s_5	0.84	0.95	0.35	0.52	1	0.66	0.02
s_6	0.55	0.47	0.48	0.24	0.66	1	0.72
s_7	0.92	0.26	0.81	0.85	0.02	0.72	1

표 2에서 bs₂에 알파-레벨을 0.5-cut로 했을 때 관리되는 세그먼트들은 s₁={s₁, s₃, s₅, s₆, s₇}, s₂={s₂, s₃, s₄, s₅}, s₃={s₁, s₂, s₃, s₄, s₇}, s₄={s₂, s₃, s₄, s₅, s₇}, s₅={s₁, s₂, s₄, s₅, s₆}, s₆={s₁, s₅, s₆, s₇}이고, s₇={s₁, s₃, s₄, s₆, s₇}이 된다. 따라서 반사관계, 대칭관계, 전이관계를 만족하는 임의의 세그먼트들은 퍼지 유사정도에 대한 관련정도 값을 부여 받고 알파-레벨에 따라 관련 세그먼트들을 생성하여 그룹을 관리하게 된다.

3) 호환관계

호환관계는 bs에서의 세그먼트 s의 발생빈도를 이용하여 퍼지 유사관계를 설정하며, 발생빈도가 높을수록 퍼지 관계의 유사도가 높게 나타난다. 호환관계는 반사관계와 대칭관계에 기반을 두며, α-cut={s | bs(s) ≥ μ}를 만족하는 s들을 퍼지 유사 관계로 간주한다. 본 논문에서는 호환관계에 의한 퍼지 유사 관계 척도로 FAS(Fuzzy Average Similarity)[10]를 이용하며, FAS는 해당 세그먼트 s가 bs와 일치되는 정도를 결정하는 퍼지 유사 관계이다. 이 기법은 하나의 bs에서 많은 s들로 구성된 유사도 중복성 문제를 해결하기 위해 사용되며, 다음과 같이 정의된다.

$$\begin{aligned}
 \text{(정의14) } FAS &= \sum_{G_j \in \text{부모}(G_i) \text{의 자식}} (\alpha\text{-cut}) \\
 &= (\sum \text{집합 } \{s \mid bs(s) \geq \mu\} \text{의 원소}) / k
 \end{aligned}$$

여기서 k는 bs에서의 전체 세그먼트 수이다.

예를 들어 s와 bs간의 퍼지 유사 관계가 [표3]과 같다고 가정하자.

[표 3] s와 bs간의 퍼지 유사 관계

s \ bs	bs ₁	bs ₂	bs ₃	bs ₄	bs ₅	bs ₆
s ₁	0.75	0.53	0.32	0.31	0.54	0.84
s ₂	0.21	0.65	0.40	0.96	0.49	0.79
s ₃	0.06	0.76	0.61	0.57	0.60	0.92
FAS	0.34	0.65	0.44	0.61	0.54	0.85

표 3에서 α-cut={s | bs(s) ≥ μ}를 0.6-cut으로 적용하면 FAS ≥ 0.6을 만족하는 bs에 대한 퍼지 유사 관계는 {bs₂, bs₄, bs₆}가 된다. 이는 FAS를 만족하는 s들이 bs상에 반복적으로 존재함에 따라 유사관계가 높은 s들의 그룹 관리가 용이함을 의미한다.

4. 시뮬레이션 평가

이 절에서는 시뮬레이션 평가를 위해 Matlab을 이용한 다. 그리고 성능평가를 위해 네트워크 토폴로지, 세그먼트 분포, 성능 척도 및 시뮬레이션 결과 등에 대해서 살펴본다.

4.1 네트워크 토폴로지

시뮬레이션 평가를 위한 네트워크 토폴로지는 유클리드 공간 모델에 기반을 두며 유클리드 공간은 D-차원의 하이퍼큐브이다. 노드들은 하이퍼큐브에 무작위로 존재하며 거리척도는 노드들 사이의 유클리드 거리 척도에 대응된다. 본 논문에서는 실험 평가를 위해 하이퍼큐브의 에지 경로를 수행하는 총 시간을 3,000ms로 제한하고 제안된 기법을 적용 한 후 10개의 그룹을 생성하여 시뮬레이션을 수행한다. 그룹 내의 피어들의 경우 하나의 피어에서 다른 피어들에게 세그먼트를 전송하는 데 걸리는 시간은 매우 적다고 가정한다. 다른 그룹 피어들의 경우에 피어들 사이의 전송 소요시간은 그룹 들 사이의 거리에 의해 결정된다. 그리고 피어들 사이의 홉(hop) 수는 피어들 사이의 거리에 비례하며, 최대 홉 수는 10으로 제한한다. 그리고 각 링크 대역폭은 1Mbps에서 2Mbps로 하고 평균 링크 대역폭은 약 1.5Mbps로 한다.

4.2 세그먼트 분포

시뮬레이션을 위해 적용된 세그먼트는 비디오 클립과 인터넷상에서 캡처한 이미지를 사용하며, 10개의 그룹과 20,000개의 세그먼트가 있다고 가정한다. 각 세그먼트 비트율은 평균 1.5Mbps로 인코드 되며, 세그먼트의 파일당 상연시간은 1분 이내이며, 크기는 10MB 이내로 제한한다. M개의 세그먼트가 내림차순으로 구성되어 있다고

가정하며, 서버에서의 세그먼트의 적용 분포범위는 $\sum_{i=1}^n r_i=1$ 이다.

4.3 성능척도

그룹관리 전략은 서비스의 성능을 향상시켜, P2P기반의 분산 환경에서 세그먼트들을 사용자에게 효율적으로 스트리밍 서비스하기 위한 것이다. 본 논문에서는 세그먼트들에 대한 평균 전송효율, 세그먼트에 대한 미디어 품질, 서비스율, 처리 시간 지연율을 성능척도로 사용하였다.

세그먼트들에 대한 평균미디어 전송효율이란 서비스 사용자가 세그먼트를 요구한 후의 서비스하는데 걸리는 평균 접근 시간을 의미하며, 평균접근 시간이 작을수록 전송효율이 높게 된다.

세그먼트에 대한 미디어 품질 척도로 클라이언트에서

서비스되는 세그먼트에 대한 오류율을 사용한다. 세그먼트 오류율(SEG_{error})은 N×M개의 세그먼트가 서비스되는 세그먼트 SEG_{image}와 세그먼트가 서비스 되지 않을 오류 \overline{SEG}_{image} 에 의해 수행되며, 오류율은 다음과 같다.

$$SEG_{error} = \sqrt{\frac{1}{NM} \sum_{x=0}^N \sum_{y=0}^M [SEG_{image} - \overline{SEG}_{image}]^2}$$

서비스율은 관리된 그룹 내에서 주문형 세그먼트들을 서비스하기 위한 척도이며 서비스율(λ)는 다음과 같다. $\lambda = \sum_{c_i} [r_i \{1 - \prod_{\pi, j} (1 - \tau_{i,j})\}]$ 이다. 여기서 τ 는 그룹 내에서 관련된 세그먼트가 서비스될 퍼지값이다.

처리시간 지연율은 스트리밍을 요구한 서비스 요청응답이 클라이언트에 제때에 스트림 되지 않아 발생하는 지연시간으로서 스트림이 지연될 경우 클라이언트 측에서는 잠재적으로 지터가 발생한다. 이때 처리 시간 지연은 스트리밍을 수행하는 캐시에 의해 발생되며, 블록 세그먼트를 인코딩하는 E_r과 R_pP(x), R_vP(x) 및 평균 네트워크 대역폭 B_w에 의해 시간 지터가 결정된다.

4.5 시뮬레이션 결과

이 절에서는 제안된 기법을 평가하기 위해 서버환경에서 무작위로 분포된 세그먼트들에 대한 무작위 기법과 TGD 기법, 그리고 TPD기법에 대해서 평균 전송효율과 서비스율을 시뮬레이션한다.

1) TGD(Truncated Geometric Distribution) 기법

TGD 기법[3]은 P2P 기반의 세그먼트 성능을 알아보기 위한 기법이며 적용되는 식은 $r_i = SN^i/B$, $B = \sum_{i=1}^n SN_i = (SN - SN^{n+1})/(1 - SN)$ 이다. SN은 성능을 제어하는 제어 인수이다.

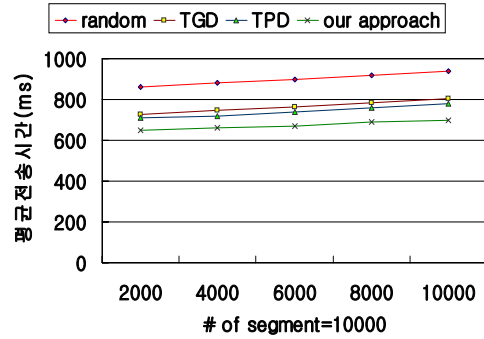
2) TPD(Truncated Pareto Distribution) 기법

TPD[3]기법은 P2P 기반의 세그먼트 성능을 알아보기 위한 기법이며 적용되는 식은 $r_i = i^{-\beta-1}/B_{\beta+1, N}$ 이며, 여기서 $B_{y, N} = \sum_{i=1}^n i^{-y}$ 는 정규화 인자이고 β 는 성능을 제어하는 제어인수이다.

3) 평균 전송 품질

전송 효율 평가를 위해 10,000개의 세그먼트를 2,000

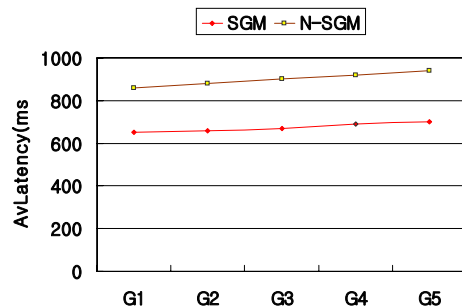
개 단위로 하여 평균 전송 시간을 평가한다. 그림 2에서 보듯이 세그먼트의 수가 증가하면 평균 전송시간은 증가하게 된다. 또한 저장용량이 고정된 경우에 세그먼트를 추가하게 되면 각 세그먼트에 대한 관리될 세그먼트의 수가 증가되어 전송효율이 감소하게 된다.



[그림 2] 평균 전송 효율

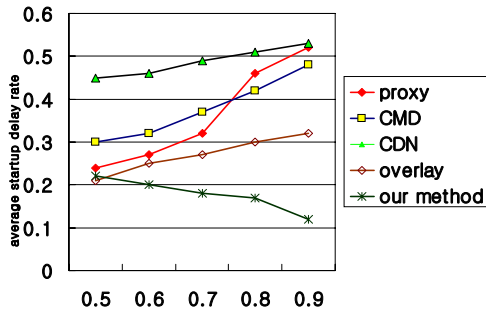
그림에서 보듯이 제안된 기법은 비교적 전송 품질이 낮은 TPD기법과 비교해 볼 때 8% 정도 전송 효율이 개선됨을 알 수 있으며, Random 기법과 비교해 볼 때는 30% 정도의 효율이 개선됨을 알 수 있다.

그림 3은 그림 2와 달리 k=20,000으로 설정하여 시뮬레이션을 수행한 결과이다. 즉 전체 세그먼트의 수 k에서 시뮬레이션을 위해 G1=4,000, G2=8,000, G3=12,000, G4=16,000, G5=20,000의 5개의 세그먼트 그룹으로 분류하여 평균 전송시간을 수행하였다. 그 결과 우리의 기법이 그룹화 관리가 수행되지 않았을 때(N-SGM)보다 30% 정도의 성능개선이 수행됨을 알 수 있다. 따라서 SGM 기법을 이용한 경우 세그먼트들의 관리 성능이 향상됨을 알 수 있으며 그렇지 않는 경우에는 성능이 감소됨을 알 수 있다.



[그림 3] k=20,000 일때

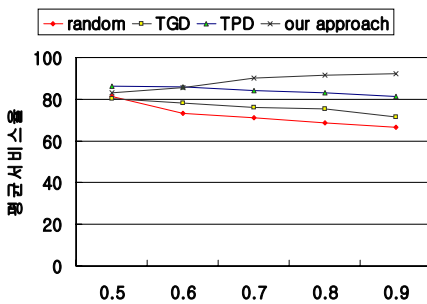
그림 4에서는 알파-레벨을 $0.5 \leq \text{알파-레벨} \leq 0.9$ 일 때로 구분하여 기존의 기법들 즉 proxy기법, CMD기법, CDN기법, overlay기법들과 제안된 기법의 평균 시간 지연율을 수행하였다. 그 결과 알파-레벨이 0.9일 때 CDN 및 proxy기법에 비해 성능이 약 4배정도 향상됨을 알 수 있다.



[그림 4] $0.5 \leq \text{알파-레벨} \leq 0.9$ 일 때 평균 시간지연율

4) 서비스율

서비스율은 관계성 기반 관리에 기반을 두며, 퍼지 제어 인수를 이용하여 성능을 측정하게 된다. 그림 5는 퍼지 제어 인수를 0.5 이상으로 했을 때의 평균 서비스율이며, 세그먼트의 수를 1,000개에서 10,000개까지 2,000개 단위로 변화시킨 결과이다. 이 그림에서 퍼지 제어 인수가 0.5일 때는 서비스율 면에서 TPD 성능이 우수하였으나 퍼지 제어 인수가 0.7 이상일 때는 본 논문에서 제안된 관계 기반 그룹 관리가 적용됨으로서 서비스율이 증가됨을 알 수 있으며, TGD 기법과 TPD 기법 또한 평균 서비스율이 약간 개선됨을 알 수 있다. 그러나 제안된 기법에서는 퍼지 제어 인수가 증가되고 세그먼트의 수가 증가될 때 평균 서비스율이 TGD, TPD기법에 비해 개선됨을 알 수 있다. 따라서 평균 효율은 TPD 기법보다 약 10%, 랜덤기법보다 약 30%정도 개선됨을 알 수 있다.



[그림 5] 평균 서비스율

5. 결론

최근에 P2P기반의 분산 환경에서 동적인 분산 콘텐츠를 효율적으로 스트리밍하기 위한 많은 연구가 수행되고 있다. 본 논문에서는 동적인 미디어 콘텐츠를 사용자 주문형 타입으로 서비스하기 위한 세그먼트 그룹 관리 기법을 제안하였다. 제안된 기법은 서버로부터 블록 세그먼트들을 분할하여 인식하고 인식된 세그먼트들에 대해서 SGA과정을 수행한다. SGA과정이 인식된 후에는 SGM이 수행되도록 하였으며, SGM은 거리기반 관리와 관계성 기반 관리로 SGM이 수행된다. 거리 기반 관리 기법은 유클리드 공간 모델에 기반을 두며, 거리 최소화에 의해 블록 세그먼트들이 관리된다. 그리고 관계성 기반 관리 기법은 알파-레벨관계, 유사관계, 호환관계를 통해서 세그먼트들이 관리된다. 거리기반 관리는 검색 효율성을 제공하기 위한 관리기법이며, 관계성 기반 관리 기법은 서비스율을 향상시키기 위한 기법이다.

제안된 기법의 효율성을 알아보기 위해 검색효율과 평균 서비스율을 비교 평가하였으며, 비교 대상은 랜덤기법, TGD, TPD기법이다. 그 결과 제안된 기법이 비교 대상 기법에 비하여 검색 효율은 8% ~ 30%, 평균 서비스율은 10% ~ 30% 정도 개선되었음을 알 수 있다.

참고문헌

- [1] Songqing Chen, Bo Shen, Susie Wee, and Xiaodong Zhang, "Segment-Based Streaming Media Proxy: Modeling and Optimization," IEEE Trans. on MULTIMEDIA, VOL. 8, NO. 2, pp. 243-256, 2006.
- [2] K. Wu, P.S. Yu, and J. Wolf, "Segment-based Proxy Caching of Multimedia Streams," Proc. WWW, Hong Kong, 2001.
- [3] Zhe Xiang, Qian Zhang, Wenwu Zhu, Zhensheng, and Yu-Qin Zhang, "Peer-to-Peer Based Multimedia Distribution Services," IEEE Trans. on MULTIMEDIA, VOL. 6, NO. 2, pp. 343-354. 2004.
- [4] Charu C. Aggarwal, Jiawei Han, Jianyong Wang, and Philip S. Yu, "A Framework for On-Demand Classification of Evolving Data Streams," IEEE Trans. on KNOWLEDGE AND DATA ENGINEERING, VOL. 18, NO. 5, pp. 577-589, 2006.
- [5] D. A. Tran, K. A. Hua, T. T. Do, "A peer-to-peer architecture for media streaming," IEEE J. Select. Areas Comm., Special Issue on Recent Advances in Service Overlay Networks, vol. 22, pp. 121-133,

2004.

- [6] S. Chen, B. Shen, S. Wee, and X. Zhang, "Investigating Performance Insights of Segment-based Proxy Caching of Streaming Media Strategies," Proc. ACM/SPIE Conf. Multimedia Computing and Networking, San Jose, CA, 2004.
- [7] Songqing Chen, Bo Shen, Susie Wee, and Xiaodong Zhang, "Segment-Based Streaming Media Proxy: Modeling and Optimization," IEEE Trans. on MULTIMEDIA, vol. 8, NO. 2, pp. 243-256, 2006.
- [8] S.Chen, B.Shen, S.We, and X. Zhang, " Adaptive and Lazy Segmentation based Proxy Caching for Streaming Media Delivery," Proc. ACM NOSSDAV, Monterey, CA, pp. 429-441, 2003.
- [9] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu, "On Demand Classification of Data Streams," Proc. ACM SIGKDD, pp. 503-508, 2004.
- [10] 이종득, 김대경, "퍼지 필터링 구조를 이용한 멀티 미디어 통계 사서함 시스템," 한국정보처리학회논문지 제11-B권 제6호, pp. 709-716, 2004.
- [11] [online] Available: <http://www.akamai.com/>
- [12] [online] Available: <http://www.napster.com/>
- [13] [online] Available: <http://www.gnutella.com/>
- [14] [online] Available: <http://freenet.sourceforge.com/>
- [15] [online] Available: <http://www.centerspan.com/>

이 종 득(Chong Deuk Lee)

[정회원]



- 1989년 2월 : 전북대학교 대학원 전산통계학과 (공학석사)
- 1998년 2월 : 전북대학교 대학원 전산통계학과 (이학박사)
- 1992년 3월 ~ 2002년 2월 : 서남대학교 컴퓨터정보통신학과 교수
- 2002년 3월 ~ 현재 : 전북대학교 응용시스템 공학부 교수

<관심분야>

정보통신, 멀티미디어 통신, 무선통신, 모바일 성능평가

정 택 원(Taegwon Jeong)

[정회원]



- 1981년 2월 : 서울대학교 대학원 전기공학과 (공학석사)
- 1991년 8월 : Univ. of Florida Dept of EE (Ph.D)
- 1983년 9월 ~ 1998년 8월 : ETRI 책임연구원
- 1998년 9월 ~ 현재 : 전북대학교 응용시스템 공학부 교수

<관심분야>

정보통신, 이동통신