

멀티미디어 서버에서 동적 복제 방식을 이용한 계층적 load balancing 기법

박성주, 임민규, 한승현, 이동만
한국정보통신대학원대학교 공학부
e-mail : lazz@icu.ac.kr

Hierarchical load balancing with dynamic replication in multimedia server

Sungju Park, Mingyu Lim, Seunghyun Han, Dongman Lee
School of Engineering, Information and Communication University

요 약

전형적인 멀티미디어 시스템은 클라이언트-서버 구조로 이루어지고 있다. 사용자의 요청에 따라 요청된 데이터를 갖고 있는 적절한 서버를 찾는 방법을 확장성 있게 지원하기 위해 계층적 관리 구조가 제시되었다. 이 접근 방법에서는 중간 계층에서 영역 관리자가 사용자의 요청을 전달해 줌으로 인해서, 사용자 요청에 대한 거부 확률을 효과적으로 줄일 수 있다. 그러나 load balancing 을 이루기 위한 계층적 관리 구조에서는 정적 복제 방식을 사용함으로써 인해서 사용자 요청에 대한 거부 확률이 최소화되지 못하였다. 따라서, 본 연구에서는 계층적 관리 구조에 멀티미디어 데이터가 요청되는 패턴에 따라 동적으로 변화하는 동적 복제 방식을 도입할 것을 제안하고, 실험을 통하여 제안된 방법을 기존의 방법과 비교하였다.

1. 서론

전형적인 멀티미디어 시스템은 클라이언트 - 서버(client - server) 구조로 이루어지고 있다. 클라이언트측의 사용자가 멀티미디어 데이터를 요구하고, 멀티미디어 데이터를 관리하는 서버는 그에 맞는 데이터를 사용자에게 제공하게 된다. 멀티미디어 시스템이 사용자수에 비례해서 증가하게 될 때, 반드시 고려해야 할 사항으로 확장성(scalability)이 있다. 단일 서버로는 다수의 사용자를 지원할 수 없기 때문에 기존의 시스템 [3, 4]에서는 멀티미디어 객체가 복제되는 다중 서버 방식을 채택하고 있다. 사용자의 요청에 따라 요청된 데이터를 갖고 있는 적절한 서버가 채택되어 사용자에게 서비스를 해주게 된다. 적절한 서버를 찾는 방법으로 middle platform 접근방법(SAP ; Scalable Access Platform) [1] 이 제시되었다. SAP에서는 사용자의 요청에 대한 확장성 있는 제어를 지원하기 위해, 클라이언트와 서버들간에 다음과 같이 계층적으로 구성된 중간 계층을 삽입하였다. - 최하위 노드(leaf node)는 지역적 또는 조직적 영역(domain)에 따라 몇 개의 멀티미디어 서버를 그룹화하고 관리한다. 각각의 영역은 영역 관리자(domain manager)에 의해 관리되고, 영역 관리자는 하위 관리자(children manager)에 대한 정보만 유지하게 된다. 사용자가 멀티미디어 객체를 요청하게 되면, 최상위 관리자(root manager)는 요청된 데이터가 접근되어 질 수 있는 적절한 하위 관리자에게 사용자의 요청을 전달해 주게 된다.

SAP 은 관리 영역을 그룹화함으로써 확장성을 지원하고자 했다. 각각의 영역에는 영역 관리자가 관리하는 몇 개의 서버가 있고, 이 영역은 영역 관리자에 의해 관리되어지며 계층적으로 구성된다. 서버들이 지역, 조직, 서버의 능력에 따라 영역별로 분류되어 그룹화되었기 때문에, 이러한 계층적 영역 구조는 관리 복잡도(management complexity) - 확장성을 지원하기 위한 평가 기준 중의 하나 - 측면에서 확장성을 달성하게 된다. 또한 중간 계층에서 영역 관리자가 사용자의 요청을 전달해 줌으로 인해서, 사용자 요청에 대한 거부 확률(rejection probabilities)을 최소화할 수 있다.

SAP 은 서버의 통계적 load 정보에 기반한 정적 복제 방식을 채택하고 있다. 서버의 load 를 주기적으로 확인해서, 어떤 멀티미디어 데이터가 자주 요청되면 그 데이터를 다른 서버로 복제한다. 어떤 서버에 복제할 것인가는 서버의 대역폭(bandwidth)과 공간(space) 비율에 기준하여 적절한 서버를 선택한다[2]. SAP 은 동적 load balancing 을 이루기 위한 계층적 구조로 인해 향상된 자원 관리 능력과 함께 사용자 요청 거부 확률을 줄였지만, 정적 복제 정책을 채택함으로써 여러 서버들간의 load imbalance 를 야기하여 시스템의 성능을 떨어뜨릴 수도 있다. 만약 어떤 서버의 멀티미디어 데이터가 사용자에게 인기가 높아진다고 가정하면, 그 서버는 급속히 동시 사용자 접속 한계에 다다를 것이다. 이런 정적 복제 방식은 계층적 구조의 동적 load balancing 기법에도 불구하고 사용자 요청 거부 확률을

높이게 된다. 따라서 본 연구에서는 계층적으로 관리되는 middle platform 구조에 멀티미디어 데이터가 요청되는 패턴에 따라 동적으로 변화하는 동적 복제 방식을 도입할 것을 제안한다. 제안된 방식의 주된 초점은 사용자 요청 거부 확률을 보다 더 동적인 방법으로 줄이고자 하는데 있다.

본 연구의 동기는 서비스 중에도 좀 더 빈번하게 요청되는 데이터는 다른 서버로 복제되어지고, 상대적으로 덜 요청되는 데이터가 그를 위해 제거되어질 수 있다는 것에 기인한다. 각각의 멀티미디어 서버가 사용자의 요청 패턴을 감시하다가, 어떤 멀티미디어 데이터가 정해진 역치값(threshold)을 넘어설 때, 그 데이터를 다른 서버에 복제한다. 이런 방식으로 멀티미디어 데이터가 각 서버에서 더 적응성 있는 균형을 유지할 수 있게 되고, 따라서 성능 향상이 이루어지게 된다.

동적 복제 방식은 이미 기존의 시스템에서 연구되어져왔다[3, 5]. 본 논문에서는 계층적 관리 구조와 동적 복제 방식을 결합시킴으로써, 멀티미디어 시스템의 응답성(responsibility)을 향상시키고자 한다. 비록 동적 복제 방식을 채택함으로써 각 서버가 부가적인 처리량이 요구되고, 같은 서버의 정보 변경이 필요하다 할지라도, 이로 인해 얻는 성능향상이 더 높음을 실험을 통하여 증명하였다.

시뮬레이션 결과에 의하면 동적 복제 방식의 사용자 요청 거부 확률이 전체 요청의 0~4% 인데 반해, 같은 환경의 실험에서 정적 복제 방식은 23.9%~44.6%의 수치를 보였다. 그럼에도 불구하고 둘 간의 사용자 요청당 복제 회수 비율은 1~2% 차이밖에는 보이지 않았다. 따라서 같은 시뮬레이션 환경에서 동적 복제 방식이 부과하는 부가적인 처리량과 서버의 정보 변경 회수는 무시할 만큼 적다고 말할 수 있다.

2. 관련 연구

2.1 계층적 구조

멀티미디어 데이터는 방대한 저장 공간과 I/O 대역폭을 요구하기 때문에, 확장성과 고장감내 측면에서 볼 때 중앙 집중 방식의 서버 구조를 고려하는 것은 바람직하지 못하다. 사용자로부터 요청되는 데이터를 분산시켜서 처리하기 위해서 계층적 구조[1, 4, 6] 접근 방식이 제시되어 왔다. 계층적 구조는 크게 두 가지 접근 방식으로 나뉘어진다. 첫 번째는 계층적 관리 방법[1]으로, 여기에서는 중간 노드가 실제 서비스를 제공하지 않고 단지 하위 노드를 관리하는 역할만 한다. 두 번째는 계층적 서비스 방법[4]으로 최하위 구조뿐만 아니라 중간 노드도 서비스를 제공하게 된다.

2.2 복제 방법 [3]

복제는 분산 멀티미디어 시스템이나 웹 캐쉬 기술에서 사용자에게 균일한 저장 구조를 제공하고 트래픽 감소, 정보 전달 시간을 줄이기 위해서 사용되어지고 있다. 복제 방법은 크게 정적 복제 방법과 동적 복제 방법으로 나뉘어진다.

정적 복제 방법은 기록 파일을 사용하여 주기적으로 데이터를 분산시키는 반면, 동적 복제는 사용자의 요청이나 선호도에 따라 실제 서비스 시간에 동적으로 데이터를 분산시킨다. 이 두 가지 방법은 서로간에 trade-off 가 있다. 정적 복제 방식은 기록파일을 사용하여 정확한 예측을 할 수 있다면 효과적으로 데이터를 분산시킬 수 있는 장점이 있다. 반면, 사용자의 변화하는 선호도에 신속하게 대응하지 못한다는 단점이 있다. 동적 복제 방식은 변화하는 사용자의 선호도에 잘 대응할 수 있는 반면, 그에 따르는 비용을 항상 고려해야 한다.

3. 설계

본 연구에서 제안하는 설계는 Boutaba 가 제안한 SAP 과 유사한 계층적 관리 방식을 사용한다. 서버는 논리적인 계층 구조로 구성되어 있다. 논리적인 계층 구조는 중간 노드(intermediate node)가 실제 서비스를 제공하지 않고 단지 최하위 노드만이 고객에게 서비스를 제공하는 것을 의미한다. 서버는 영역에 의해 그룹되어지고, 영역은 주로 지형적 지역으로 구분되어진다. 중간 노드는 영역을 관리하며, 이는 어떤 최하위 노드에 포함될 수도 있고 분리된 서버일 수도 있다. 중간 노드는 최하위 노드의 workload 를 관리하고 사용자 요청을 스케줄하는 역할을 수행한다. 이를 위해 중간 노드는 각 멀티미디어 서버의 상태 - workload, 현재 서버의 사용자 요청 수용 능력, 멀티미디어 데이터(비디오)의 종류 등- 를 유지한다.

본 연구에서는 역치근거(threshold-based) 동적 복제 방식을 채택한다. 만약 요청된 데이터를 갖고 있는 서버가 정해진 역치값을 넘어선 과부하(heavy-loaded) 상태이면, 멀티미디어 데이터는 동적으로 저부하(lighty-loaded) 상태인 서버로 복제된다. 그림 1 에 본 연구에서 제안한 방식의 기본적인 구조가 있다.

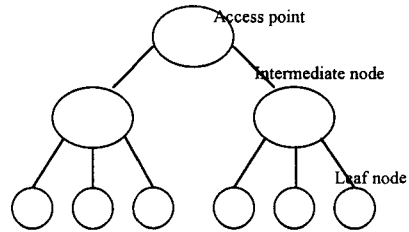


그림 1. 기본 구조

4. 시뮬레이션 모델

이 장에서는 시뮬레이션에 사용된 가정과 한정 요소(parameter), 사용자 행동 양식과 패턴을 기술하고, 시뮬레이션 모델이 제시된다.

4.1 사용자 행동 양식

사용자가 VoD (Video-on-Demand) 시스템에 영향을 미치는 요소 중에는 여러가지가 있지만, 여기서는 두 가지 요소 만을 고려하였다. 하나는 사용자의 요청 패턴이고 다른 하나는 비디오에 대한 사용자의 선호도이다. 현재까지 개발된 시스템 중 '진정한 VoD 서비스'를 제공하는 시스템은 아직 없다. 그래서, 이에 대한 요청 패턴과 시청 주기 패턴을 얻지는 못 하였고, 공중파 TV 방영과 인터넷 TV 에 대한 자료를 사용하였다.

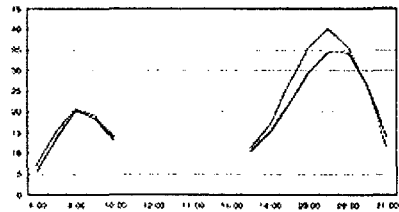


그림 2. TV 시청 주기. 1998 한국

그림 2 는 TV 시청 주기를 보여준다. 이 그래프에서 볼 수 있듯이, 시청률은 균등 분포(uniform distribution)되지 않고 오전 8:00 와 오후 9:00 사이에 최대 시청률을 보여주고 있다. VoD 서비스도 TV 방영

과 마찬가지로 분포를 보여준다고 생각되어질 수 있다. 그러나 본 실험에서는 사용자가 아침보다는 밤에 VoD 서비스를 더 많이 요청한다고 가정한다. 그림 1의 통계량을 보여주는 곡선을 면밀히 살펴보면, 이 곡선은 표준분포(normal distribution)를 따름을 알 수 있다. 따라서 사용자 행동 양식에 따라 다양한 분포가 생길 수 있지만, 본 연구에서는 많은 수의 사용자들 가정하여 표준 분포를 실험에 사용하였다.

또 하나 고려해야 할 점은 멀티미디어 데이터의 인기도와 관련이 있다. 실제 VoD 서비스에서는 최신 유행되는 비디오가 다른 비디오들보다 더 많은 요청 빈도가 생길 수 있다. 따라서 시물레이션에 쓰이는 비디오 객체들에게 Zipf's distribution에 따르는 각각의 인기도를 지정하였다. Zipf's law의 원래 정의는 다음과 같다. "장문의 문장에서 주어진 단어의 발생횟수는 발생되는 빈도수 순서의 역수라는 경험적 법칙"

Zipf's law의 간단한 형태로는 사람들의 행동양식이나 비디오의 인기도를 완전히 반영할 수 없을 것이지만, 본 실험에서는 비결정적인 실제 세계를 결정적인 시물레이션 모델에 반영하기 위해 Zipf's distribution을 실험에 사용하였다.

4.2 시물레이션 구성요소

본 연구의 시물레이션 모델을 위해 다음과 같은 시물레이션 요소들이 사용되었다.

- 일정 기간에 각각의 사용자가 시청하는 영화(또는 멀티미디어 데이터)의 평균 편수
- 일정 기간에서의 총 요청 회수
- 사용자 요청이 도착하는 비율 : 요청이 도착하는 비율은 하루 동안에도 계속 변하기 때문에, 본 실험에서는 영역안에서의 사용자 요청은 비동질적인 $\lambda g(t)$ 를 한정요인으로 가지는 Poisson process에 따라 도착된다고 가정하였다. 또한 시간 t에서 영역안에서의 총 도착 비율은, $\lambda g(t)$ 는 다음 분포 함수에 의한 표준 분포를 따른다고 가정하였다.

$$\lambda g(t) = N * (\lambda / 7) * (e^{-(t-\mu) / 2\sigma^2})$$

μ : the peak hour
 N : users
 λ : session per week per user
 σ : standard deviation of the user profile
 $\lambda g(t)$: aggregate request rate

μ 는 사용자 요청이 가장 많은 시간이고, $\lambda g(t)$ 는 한정 요소 σ 에 의해 제어되는 총 요청 비율이다. σ 값을 크게 하면 그래프에서 시간에 따라 널리 퍼지기보다는 곡선을 그릴 것이다.

- 멀티미디어 데이터 선택 패턴 : 각 멀티미디어 데이터는 사용자 요구에 따라 각기 다른 인기도가 있다. 어떤 특정 멀티미디어 객체에 대한 사용자 요청 패턴을 예상 할 수는 없지만, Zipf's distribution에 의해 VoD 서버의 객체들의 선호도를 제공할 수는 있다.
- 요청되는 멀티미디어 객체의 길이 : 멀티미디어 객체의 길이는 다양하지만, 평균적으로 비디오의 상영 시간은 60분에서 120분 사이이다. 따라서 이 실험에서는 멀티미디어 객체의 길이를 60분에서 120분 사이의 분단위로 결정하였다.
- 영역에 포함되는 멀티미디어 서버의 수
- 멀티미디어 서버의 용량 : 서버 용량은 서버가 동시에 서비스할 수 있는 비디오의 최대 개수로 정의된다

4.3 비교 요소

요청 거부 확률은 VoD 시스템에서 양 극단 서비스의 질(End-to-end Quality of Service)에 밀접하게 연관되어 있다. 정적 복제 방식은 사용자 요청 패턴과 여러 변화에 잘 적응하기 힘들기 때문에 높은 요청 거부 확률을 가질 수 있다. 그에 반해, 동적 복제 방식은 서버의 workload를 감시하는데 시스템 자원을 더 많이 쓰고 그로 인해 시스템에서 더 많은 비용을 사용하게 할 수도 있다.

복제로 인한 비용은 I/O 비용, 네트워크 대역폭 소비, 여러 시스템 자원 소모 등을 포함한다. 동적 복제에 위에 언급한 여러 비용들을 부담하더라도, 전체 시스템의 구성을 좀 더 사용자 요청에 정확하게 변화할 수 있도록 함으로써 서버의 사용가능한 서비스 능력을 향상시키고자 하는 바람에서 행해진다. 따라서 복제의 비용과 그로 인해 얻는 성능향상은 다음과 같은 요소들에 의해 비교되어야 한다 : 복제의 빈도가 높아질 수록, 즉, 더 동적일수록, 복제로 인한 비용은 높아진다. 그러나 복제가 어떤 범위안의 빈도만큼 행해질 때는 그 비용에도 불구하고 멀티미디어 시스템의 성능을 향상시킬 것이다. 이 역치값 - 동적 복제로 인한 성능향상의 잇점이 동적 복제의 비용보다 많아지는 때의 -은 실험에 의해서 결정되어질 수 있다.

4.4 가정

시물레이션 모델을 단순화하기 위한 작업으로 몇 가지 시물레이션에 쓰이는 값을 상수로 정하였다.

- 멀티미디어 서버의 용량과 영역안에서의 멀티미디어 서버의 수 : 본 실험에서는 서버가 저장할 수 있는 멀티미디어 개수와 서버 용량을 모두 동일하다고 가정하였다.

5. 시물레이션 환경

제한된 계층적 구조에서의 동적 복제 방식은 Java를 사용하여 Linux 환경에서 시물레이션 되었다. SAM, DM(Domain Manager), 비디오 서버는 객체로 구현되었다. 각 객체는 서버 load, 영역 관리자 load 등에 관한 정보를 유지하는 테이블을 가지고 있다.

시물레이션은 사용자 패턴에 따라 총 1000개의 사용자 요청에 의해 행해졌다. 시물레이션 시행 도중, 요청된 객체에 대한 load가 정해진 threshold를 넘어갈 때, 복제를 하게 된다.

그림 3은 기본적인 시물레이션 구조와 access node, intermediate node, leaf node가 갖고 있는 데이터 구조를 보여준다.

AN_MMOobject 구조체는 최하위 노드에서 복제된 모든 멀티미디어 객체에 대한 참조를 갖고 있다. AN_Load는 Access Node가 관리하는 영역의 총 load 합을 가지고 있다. MMO_Load는 영역안에 있는 멀티미디어 객체의 인기도(요청회수)를 관리한다.

IN_MMOobject 구조체는 하위 서버에 있는 멀티미디어 객체에 대한 참조를 가지고 있다. IN_Load는 intermediate node가 관리하는 최하위 노드에 대한 load를 가지고 있다. MMO_Load 구조체는 고객에게 현재 서비스되고 있는 객체와 인기도를 가지고 있다.

6. 시물레이션 결과

시물레이션을 통하여 계층적 관리 구조에서의 동적 복제 방법과 정적 복제 방법을 실험하고, 사용자 요청 거부 확률과 동적 복제 방법에 뒤따르는 비용을 비교해보았다.

한 주동안 사용자가 시청하는 비디오의 평균 숫자는 3.5라고 가정하였고, 정적 복제는 하루마다 일어난다고 가정하였다. 따라서 시물레이션은 정해진 시간 간격을 두고 24시간을 기준으로 행해진다.

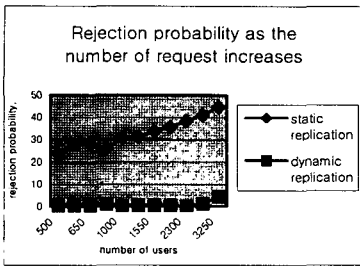


그림 4. Rejection probabilities

그림 4은 정적 복제 방법과 동적 복제 방법의 요청 거부 확률을 비교하여 나타내어준다. 사용자 수가 증가함에 따라, 정적 복제 방법의 요청 거부 확률은 높아지지만(총 요청 횟수의 23.9-44.6%), 동적 복제 방법에서는 매우 낮은 수치를 유지함을 알 수 있다. (총 요청 횟수의 0.17-4.42%) 이 결과는 동적 복제 방법이 정적 복제 방법에 비해서 시스템 자원을 최대한 활용할 뿐만 아니라 사용자에게 더 좋은 QoS를 제공해 줄을 보여준다.

복제로 인한 비용을 측정하기 위해, 초기의, 그리고 서비스 시간 동안의 복제 횟수를 측정하였다. 그림 5는 하루 동안의 복제 횟수를 보여주고 있다.

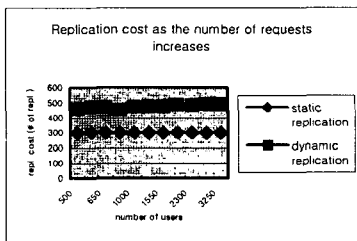


그림 5. The number of replications

정적 복제 방법의 경우에는 복제 횟수가 항상 일정하다. 이는 서버의 부하와 상관없이 정적 복제 방법에서는 주기적으로 복제를 행하기 때문이다. 상수는 초기의 멀티미디어 객체의 복제를 의미한다. 반면, 동적 복제 방법에서는 사용자 수가 증가함에 따라 조금씩 복제 횟수가 늘어나고 있다. 그러나 이것이 동적 복제 방법의 비용이 정적 복제 방법에 비해 두 배로 증가함을 의미하지는 않는다.

7. 결론

본 연구에서는 사용자 요청 패턴에 기인하여 요청 거부 확률을 적응성 있게 줄이기 위해, 동적 복제 방법을 이용한 확장성 있는 계층적 구조의 load balancing 방법을 제안하였다. 본 논문의 중요한 아이디어는 서비스 시간 중에 각 멀티미디어 데이터의 load를 감시하고 필요할 경우에 다른 서버로 복제할 수 있게 하는 것이다. 이는 사용자 요청 패턴이 변화에 따라 적절히 적응할 수 없는 정적 복제 방법에 비하여, 사용자 요청 확률을 줄일 수 있는 방법을 제공하므로 시스템의 확장성에 공헌을 하게 된다. 실험의 결과로부터 동적 복제 방법이 시스템의 자원을 잘 활용하면서도 그로 인한 비용은 비교적 적다는 것을 알 수 있다.

그러나 여전히 복제비용과 연관이 되는 상업적인 측면을 고려해야 하는 문제가 남아 있다.

참고문헌

- [1] Boutaba, R., and Hafid, A., "A generic platform for scalable access to multimedia-on-demand systems," *Selected Areas in Communications, IEEE Journal*, Vol 17, September 1999, pp. 1599-1613.
- [2] Dan, A., and Sitarum, D., "An online video placement policy based on bandwidth to space ratio," *ACM SIGMOD'95*, San Jose, CA, 1995, pp. 376-385.
- [3] Lie, P., Lui, J., and Golubchik, L., "Threshold-Based Dynamic Replication in Large-Scale Video-on-Demand Systems," *8th International Workshop on Research Issues In Data Engineering*, 1998, pp. 52-59.
- [4] Ren-Hung Hwang and Youn-Chen Sun, "Optimal Video Placement for Hierarchical Video-on-Demand Systems," *7th International Conference on Computer Communications and Networks*, 1998, pp. 454-461.
- [5] Bestavros, A., "Demand-based Document Dissemination to Reduce Traffic and Balance Load in Distributed Information Systems," *7th IEEE Symposium on Parallel Distributed Processing*, 1995, pp. 338-345.
- [6] Mundur, P., Simon, R., and Sood, A., "Integrated Admission Control in Hierarchical Video-on-Demand Systems", *IEEE International Conference on Multimedia Computing and Systems*, Volume: 1, 1999, Page(s): 220 -225
- [7] Chou, C., Golubchik, L., Lui, J., "A Performance Study of Dynamic Replication Techniques in Continuous Media Servers", *Proceedings of the international conference on Measurement and modeling of computer systems*, 1999, Pages 202 - 203
- [8] Anastasiadi, A., Kapidakis, S., "A Computational Economy for Dynamic Load Balancing and Data Replication", *Proceedings of the first international conference on Information and computation economies*, 1998, Pages 166 - 180
- [9] Venkatasubramanian, N., and Ramanathan, S., "Load Management in Distributed Video Servers", *Proceedings of the 17th International Conference on Distributed Computing Systems*, 1997, Page(s): 528 -535
- [10] Dan, A., and Sitarum, D., "An Online Video Placement Policy based on Bandwidth to Space Ratio (BSR)", *Proceedings of the 1995 ACM SIGMOD international conference on Management of data*, 1995, Page(s) 376 - 385
- [11] Nussbaumer, J. P., Patel, B.V., Schaffa, F., "Multimedia Delivery on Demand: Capacity Analysis and Implications", *the 19th Conference on Local Computer Networks*, 1994, Page(s): 380 -386
- [12] Jones, J.R., "Baseband and passband Transport Systems for Interactive", *IEEE Communications Magazine* Volume: 32 5, May 1994, Page(s): 90 -101