논문 2008-4-6

비디오 서버 클러스터 상에서의 서버 및 네트워크 부하를 고려한 비디오 배치 방법

A Method for Video Placement on a Cluster of Video Servers Using Server and Network Loads

김상철*

SangChul Kim*

요 약 본 논문은 비디오 서버 클러스터 기반의 VOD 시스템에서의 비디오 배치 문제를 정의하고 효율적인 비디오 배치 방법을 제안한다. 비디오 배치란 비디오 데이터를 위치와 개수가 정해진 비디오 서버들에 할당하는 것을 말한다. 제안된 방법의 목적은 서버들 간의 부하 균형을 유지하면서 네트워크 부하를 최소화하도록 비디오를 서버에 배치하는 것이다. 사용자 요구 기각률은 사용자가 지각하는 VOD 시스템 성능 지수들 중 특히 중요한 것으로 기존 연구들에서 VOD 시스템 성능 비교의 기준으로 삼고 있다. 우리의 방법에 대한 실험 결과를 보면, 서버 부하 균형만을 고려하거나 네트워크 부하만을 고려한 비디오 배치보다는 이 둘을 모두 고려한 경우가 사용자 요구 기각률을 낮추는 특징을 보였다. 또한 비디오 배치에서 서버 부하 균형에 대한 고려가 특히 기각률에 중요한 영향을 미친다는 것을 알 수 있었다. 우리의 조사에 따르면 비디오 서버 클러스터 환경에서 서버와 네트워크 부하를 동시에 고려하는 효과적인 비디오 배치 연구는 거의 발표되지 않았다.

Abstract The paper presents the problem definition of video placement and efficient methods for placing video data on a cluster of video servers. The video placement is to place each of video replicas on one of the servers where the number and location of the servers are already determined. The rejection ratio of user requests is one of most important user-perceive performance measures, so it has been used as a performance criteria in many researches on the VOD system. The objective of our video placement is to achieve the load balancing among servers and the minimization of total network loads. To our experiment, the presented methods show better performance in terms of the rejection ratio of user requests than the methods for video placements in which only either server load balancing or network load minimization is considered. Also, it is observed that considerations on server load balancing is especially important in video placement. To our survey, little research has been published on video placement in which server and network load are considered together in a video server cluster environment.

Keywords : VOD, video placement, server cluster.

Ⅰ. 서 론

최근 동영상을 기반으로 하는 서비스에 대한 사용자들의 요구가 급증하고 있다. 인터넷 상에서 가장 일반적으로 제공되는 동영상 서비스는 VOD (Video-On-Demand) 형태의 서비스라 할 수 있다.

비디오 데이터의 대용량 크기로 인해서 저장과 전송을 위해서는 큰 저장 공간, 고성능 서버와 큰 네트워크 대역폭이 필요하다. 이런 이유로 사용자(클라이언트)를 위한 VOD 서비스의 주된 목표는 제한된 서버 능력과 네

*정회원, 한국외국어대학교 컴퓨터공학과
접수일자: 2008.7.15, 최종수정일자: 2008.8.2

트워크 자원 하에서, 초기 지연 없이 클라이언트의 비디오 요구에 대한 기각률(rejection rate)을 낮추는 것이다. 사용자 요구 기각률은 사용자 지각 성능 (user-perceived performance) 중에서 가장 중요한 것으로서, VOD 관련 연구의 결과 분석에서 가장 많이 사용되는 지표이다 [1].

VOD 시스템에 대한 주된 연구 주제는 비디오 스트리밍(video streaming)[2], 디스크 저장(disk storage)[3], 비디오 복제(video replication)[4, 5] 및 비디오 배치(video placement)[5, 6, 7, 8]이다. 특히 비디오 복제 및 배치는 비디오 서버 클러스터 구조를 갖는 VOD 시스템에서 다루어지는 연구 주제이다. VOD 서버 클러스터의 경우 각 비디오 서버는 각자의 저장 장치를 가지고 백본 네트워크로 연결되어 있다. 이것은 다른 구조에 비해서 저장 공간이나 스트리밍 처리 능력 면에서 확장성과 신뢰성이 뛰어나는 장점이 있다.

비디오 복제 연구는 각 비디오를 몇 개로 복제하는가에 관한 것이고, 비디오 배치 연구란 이들 복제(replica)들을 어떤 서버에 배치하는가에 관한 것이다. 비디오 복제와 배치에 관한 기존 방법들 간의 비교는 [1]에 기술되어 있다. 비디오 복제의 기본 원리는 인기가 높은 비디오를 그렇지 못한 비디오에 비해서 많은 수를 복제하는 것이다. 대표적인 복제 방법으로 Zipf-like 분산 기반 방법 [5]와 한계적 애덤스 단순 제수 (Bounded Adams' Monotone Divisor) 기반 방법 [4]를 들 수 있다. 비디오 서버 클러스터 환경에서의 비디오 배치에 대한 연구들이 다수 발표되어 있다. 이들 연구의 한 방향은 서버들 간의 부하 불균형을 최소화하도록 복제들을 배치함으로써 서버들의 자료 처리 능력을 골고루 활용하도록 하는 것이다 [5, 7]. 다른 방향은 네트워크상 트래픽 부하를 줄이기 위해서, 클라이언트와 서버간의 거리를 고려하여 복제들을 배치하는 것이다 [6, 8].

클라이언트의 요구는 원하는 비디오를 보유한 서버의 출력 밴드위스가 부족하거나 클라이언트와 서버간의 네트워크 밴드위스 부족으로 경로가 존재하지 않으면 기각된다. 그러므로 사용자 요구의 기각률을 최소화하기 위해서는 서버간의 부하 균형과 네트워크상의 트래픽 부하를 함께 고려되어야 되어 한다. 즉, 서버간의 부하가 균형을 이루게 하여 클라이언트의 요구가 특정 서버로 몰리는 현상을 방지하고, 클라이언트가 원하는 비디오를 가진 서버가 가능한 가까운 거리에 위치하도록 하여 네트워크상 트래픽 양을 줄이는 것이 필요하다.

본 논문은 비디오 서버 클러스터 환경에서 서버 간의 부하 불균형과 네트워크 트래픽의 최소화를 함께 고려하기 위한 비디오 배치 문제를 정의하고, 새로운 비디오 배치 방법 두개를 제안한다. 본 방법의 주된 특징은 기존 연구와 달리 비디오 인기도와 복제 개수를 감안하여 서버들 간의 부하 불균형을 줄이면서, 클라이언트와 가까운 서버에 인기도 합이 큰 비디오 집합을 배치함으로써 네트워크상의 부하를 감소시키는 것이다. 제안된 배치 방법을 실험해 본 결과, 서버 부하 균형만을 고려하거나 네트워크 부하 최소화만을 고려한 비디오 배치보다는 이들을 동시에 고려한 비디오 배치가 사용자 요구 기각률 면에서 우수하였다. 서버 부하 균형만을 비디오 배치와의 비교에서는 기존 연구를 사용하였다. 또한 비디오 배치에서 서버 부하 균형에 대한 고려가 네트워크 부하에 대한 고려보다 사용자 요구 기각률 면에서 중요한 영향을 미친다는 것을 알 수 있었다.

우리의 조사에 의하면 서버 부하 불균형과 네트워크 부하를 함께 최소화하는 효과적인 비디오 배치에 관한 연구 발표는 거의 없었다. [9]에서는 부하가 적은 서버 순으로 비디오를 배치하는 과정에서 부하가 같은 서버들에 대해서는 서버 위치를 고려하는 비디오 배치 방법을 제안하였다. 이것은 상대적으로 인기도가 높은 비디오들을 배치할 때에만 서버 위치를 고려하는 한계가 있다.

II. 문제의 정의

1. 가정 및 용어

전체 네트워크 G는 노드 집합 G_n과 링크 집합 G_e으로 구성된다. 즉, $G = \{G_n, G_e\}$이다. 클라이언트 노드의 집합은 G_c로 나타내고 $G_c \subset G_n$이다. 클라이언트 노드들은 같은 비디오 전송 요구의 패턴과 도착률을 가진다.

서버 집합 S는 N개 서버들로 구성되며, $S = \{s_1, s_2, ..., s_N\}$이고 $S \subset G_c$이다. 각 서버 s_k는 동일한 저장 용량 C_{video}와 동일한 출력 대역폭 (outgoing bandwidth) B_{video}를 가진다.

비디오 집합 V는 M개 비디오로 구성되며, $V = \{v_1, v_2, ..., v_M\}$이다. 비디오 v_i의 인기도는 p_i로 정

의하고, 0과 1사이 값으로서 $\sum_{1}^{M} p_i = 1$이다. p_i는 사용자가 v_i를 요구할 확률이다. 비디오 v_i의 복제 개수를 r_i라고 하고, 비디오 복제별 인기도 q_i는 $q_i = \frac{p_i}{r_i}$로 계산한다.

V_k는 s_k에 배치된 모든 비디오의 집합이다. b_{video}는 비디오의 코딩률, d_{video}는 비디오의 길이, c_{video} ($c_{video} = b_{video} * d_{video}$)는 비디오의 크기로서 모든 비디오에 동일한 것으로 가정하였다. λ는 클라이언트 노드들에 발생되는 단위 시간당 비디오 요구의 평균 도착률로 정의한다.

2. 비디오 배치 문제의 정의

본 논문에서는 v_i의 복제 개수 (즉, r_i)는 기존 방법 [4, 5]로 미리 결정되어 있다고 본다.

비디오 배치 문제는 [제한 1], [제한 2], [제한 3] 및 [제한 4]를 만족하면서 [목적 1]과 [목적 2]를 달성하는 $V_k (1 \le k \le N)$를 구하는 것이다.

[목적 1] Minimize $\mathcal{L}$

$$\mathcal{L} = \max_{1 \le k \le N} |l_k - \bar{l}| \qquad \text{[정의 1]}$$

[목적 2] Minimize $\mathcal{D}$

$$\mathcal{D} = \sum_{k=1}^{N} D(k) \qquad \text{[정의 2]}$$

[제한 1] $x_{k,i} = 0$ 또는 1, $k = 1, 2 \ldots N$, $i = 1, 2 \ldots M$

[제한 2] $\sum_{i=1}^{M} x_{k,i} \cdot c_{video} \le C_{video}$, $k = 1, 2 \ldots N$

[제한 3] $\sum_{k=1}^{N} x_{k,i} = r_i$, $i = 1, 2 \ldots M$

[제한 4] $l_k \le B_{video}$, $k = 1, 2 \ldots N$

그림 1. 비디오 배치 문제의 정의
Fig 1. Problem Definition of Video Placement

위에서 사용된 용어의 정의는 다음과 같다:

$$l_k = Q_k \cdot \lambda \cdot c_{video} \qquad k = 1, 2 \ldots N$$

$$Q_k = \sum_{v_i \in V_k} q_i, \qquad k = 1, 2 \ldots N$$

$$\bar{l} = \sum_{k=1}^{N} l_k / N$$

$$D(k) = Q_k \cdot d(k), \qquad k = 1, 2 \ldots N$$

$$d(k) = \sum_{n \in G_c} Dist(n, s_k), \qquad k = 1, 2 \ldots N$$

$$V_k = \{v_i | x_{k,i} = 1, 1 \le k \le N, 1 \le i \le M\}, \quad k = 1, 2 \ldots N$$

$x_{k,i}$는 v_i의 복제가 s_k에 배치된 지 여부를 나타내는 변수이다. v_i의 복제가 s_k에 배치되면 $x_{k,i} = 1$이고, 아니면 $x_{k,i} = 0$이다. l_k는 서버 s_k의 평균 부하이고 $\bar{l}$는 l_k들의 평균이다. L은 l_k와 $\bar{l}$ 간의 최대 차이로서 서버들 간의 서버 부하 불균형 정도를 나타내고 있기 때문에[8], 본 논문에서는 "서버 부하 불균형 지수"라고 부른다. 노드 n1 $\in G_n$과 n2 $\in G_n$에 대해서, $Dist$(n1, n2)는 이들 간의 네트워크상 거리로서 홉(hop) 수로 표현된다. Q_k는 s_k에 배치된 비디오의 인기도 합이고, $d(k)$는 s_k와 모든 클라이언트 노드들 사이의 거리 합이다. Q_k는 임의의 비디오 요구를 처리하고자 s_k가 선택될 확률이기도 하다. $D(k)$는 클라이언트들과 서버 간의 비디오 스트리밍이 발생할 확률을 감안한 s_k와 모든 클라이언트들 간의 거리 합이다. 그러므로 D는 사용될 확률을 감안한 전체 서버와 클라이언트 간의 거리 합이다. D가 작으면 큰 경우에 비해서 비디오 서비스를 위한 네트워크의 부하가 적을 가능성이 높다. 그런 의미에서 D는 네트워크 부하의 총량을 예측하는 지수라고 볼 수 있다. 본 논문에서는 D를 "네트워크 부하 지수"라고 부른다.

클라이언트가 요구하는 비디오를 가진 서버가 비디오 스트림을 전송해 주지 못하는 경우는 두 가지이다. 하나는 해당 서버가 과부하 되어 서버의 출력 대역폭이 충분하지 못한 경우이다. 다른 하나는 비디오 전송을 위한 충분한 네트워크 대역폭을 가지는 경로가 해당 서버와 클라이언트 간에 존재하지 못하는 경우이다. [목적 1]과 [목적 2]는 이 두 가지 경우를 최소화하고자 설정되었다.

[제한 2]는 각 서버에 배치된 비디오 데이터들의 총 크기는 저장 용량 C_{video}를 넘지 않는다는 점을 나타낸다. [제한 3]은 v_i의 복제 수는 r_i개임을 나타낸다. [제한 4]는 각 서버의 평균 부하는 최대 전송 대역폭 B_{video}를 넘지 않는다는 점을 나타낸다.

[목적 1]에서는 L을 최소화함으로써 사용자 요구가 특정 서버들에 집중되어 서버 과부하가 발생하는 현상을 피하고자 한다. 또한, [목적 1]은 네트워크상의 특정 지역에 부하가 편중되는 현상도 줄여 준다. 그 이유는 서버들은 네트워크상에서 일반적으로 지역적으로 균등하게 분산되어 있기 때문에, 서버들 간의 부하가 균등하면 네트워크 지역들 간의 부하도 어느 정도 균등하게 되는 효과가 발생한다. [목적 2]는 $D(k)$들의 합인 D를 최소화함으로써 네트워크상의 부하를 최소화하는 것이다. 네트워크상의 부하가 많아지면 서버와 클라이언트 간에 충분한 밴드위스를 가진 경로가 존재하지 않을 가능성이 높아진다.

III. 비디오 배치 방법 및 분석

1. 비디오 배치 방법

[목적 1] 만을 고려하면 비디오 배치는 빈 패킹 문제(bin packing) 와 유사하고, [목적 2]만을 고려하면 편의시설 배치 문제(facility location problem)와 유사하다. 이들 문제가 NP-Complete인 점을 감안하면 최적의 비디오 배치를 구하는 폴리노미알 타임(polynomial-time) 알고리즘은 존재하지 않는다. 본 논문에서는 서버 간의 부하 균형과 네트워크 부하 최소화라는 두 가지 목적을 함께 고려하는 휴리스틱 알고리즘 두개를 제안한다.

방법 1 (M1):

Step 1) $V_k \leftarrow \varnothing$, k = 1, 2 ... N

$$S_{cap} \leftarrow \frac{C_{video}}{c_{video}}$$

Step 2) q_i의 내림차순으로 v_i들을 정렬하고, 정렬된 순서대로 Step 3을 수행한다.

Step 3) 각 $v_i \in V$에 대해서,

Step 3-1) s_1, s_2 ... s_N에 대해서, l_1, l_2 ... l_N을 계산한다.

Step 3-2) 배치된 비디오 수가 S_{cap} 보다 작은 서버들 중에서, l_k가 가장 작은 서버 $(1 + \Delta) \cdot r_i$개를 선택한다. 선택된 서버 집합을 Z_1라고 하자.

Step 3-3) 각 $s_k \in Z_1$에 대해서, $D(k)$를 계산한다.

Step 3-4) Z_1의 서버들 중에서, $D(k)$가 가장 작은 서버 r_i개를 선택한다. 선택된 서버 집합을 Z_2라고 하자.

Step 3-5) 각 $s_k \in Z_2$ 에 대해서, 다음을 수행한다.

$V_k \leftarrow V_k \cup \{v_i\}$

그림 2. 비디오 배치 방법 1
Fig 2. Video Placement Method 1

Step 1에서는 각 V_k를 공집합으로 초기화한다. Scap은 각 서버에 배치할 수 있는 총 비디오 개수로서 저장 용량을 비디오 개수로 표현한 것이다. Step 2에서는 비디오 복제별 인기도 q_i의 내림차순으로 비디오들을 정렬한다. 그 정렬이 끝나면, 첫 번째 비디오부터 차례로 Step 3을 수행한다.

Step 3-1에서는 각 서버의 평균 부하를 계산한다. 서버의 평균부하는 현재 어떤 비디오들이 해당 서버에 배치되어 있는지에 따라 결정된다. 예를 들어 V_k = 0이면 l_k = 0이 될 것이다.

Step 3-2는 현재 평균 부하가 가장 작은 서버들에 비디오 v_i를 배치한다. 어떤 서버의 평균부하가 적다는 점은 그 서버의 출력 대역폭에 여유가 많다는 것을 의미한다. Step 3이 복제별 인기도가 높은 비디오부터 수행되는 점을 함께 감안하면, Step 3-2의 목적은 서버들이 결과적으로 균등한 평균 부하를 갖도록 유도하는 것이다. Δ는 0이상 값으로서, 실험적으로 정해지는 파라미터이다.

Step 3-3은 Z1의 각 서버에 대해서 $D(k)$를 계산한다. Step 3-4는 v_i를 Z1의 서버들 중에서 클라이언트들과의 거리가 짧은 서버들에게 배치한다. Step 3-4의 목적은 서버와 클라이언트간의 트래픽을 감소시켜 네트워크 대역폭 부족으로 인한 비디오 요구 기각률을 줄이는 것이다.

만약 파라미터 Δ가 충분이 큰 값이면 Z1은 모든 서버들의 집합이 될 것이다. 이런 경우 Step 3-4는 $D(k)$만을 고려해서 서버를 선택함으로써 Step 3-4의 목적을 최대한 달성하게 된다. 반면에 서버 부하 균형을 달성하

려는 Step 3-2의 목적 달성은 어려워진다. $D(k)$ 대신에 $d(k)$를 사용하도록 수정하면 서버 선택 시 서버 부하가 미치는 영향을 최소화할 수 있을 것이다. 만약 Δ = 0이면, 이와는 반대 경우가 된다. 이 경우 Z1은 부하가 가장 작은 서버 r_i개의 집합이 되어, Step 3-4의 Z2는 Z1과 동일하게 된다. 따라서 Step 3-4의 목적은 최대한 달성할 수 있지만, Step 3-2의 목적을 달성할 가능성은 낮아진다. Δ의 값은 사용자 요구의 기각률을 최소화하도록 실험적으로 정해진다.

Step 3의 동작을 예를 들어 설명한다. 현재 v_2를 고려하고 있고, N=5, Δ=0.5, r_2=2라고 가정하자. 따라서 (1 + Δ)r_2 = 3이다. Step 3-1에서 계산한 결과가 l_1=100, l_2=130, l_3= 300, l_4=150, l_5=200라고 가정하자. Step 3-2에서 평균부하가 작은 순으로 서버 3개를 선택하면 s_1, s_2, s_4가 선택되므로 이들이 Z1을 구성한다. Step 3-3에서 계산한 결과가 $D(1)$=300, $D(2)$=250, $D(4)$=280라고 가정하자. Step 3-4에서 $D(k)$가 작은 순으로 서버 2개를 선택하면 s_2와 s_4가 선택되므로 이들이 Z2를 구성한다. Step 3-5에서는 v_2가 V_2와 V_4에 각각 추가된다.

방법 2 (M2):

Step 1) $V_k \leftarrow \Phi$, k = 1, 2 ... N

$$S_{cap} \leftarrow \frac{C_{video}}{c_{video}}$$

Step 2) q_i의 내림차순으로 비디오 v_i들을 정렬하고, 정렬된 순서대로 Step 3을 수행한다.

Step 3) 각 $v_i \in V$에 대해서,

Step 3-1) s_1, s_2 ... s_N에 대해서, l_1, l_2 ... l_N을 계산한다.

Step 3-2) 배치된 비디오 수가 S_{cap} 보다 작은 서버들 중에서, l_k가 가장 작은 서버 r_i개를 선택한다. 선택된 서버 집합을 Z라고 하자.

Step 3-3) 각 $s_k \in$ Z에 대해서, 다음을 수행한다.

$V_k \leftarrow V_k \cup \{v_i\}$

Step 4) Q_k의 내림차순으로 V_k들을 정렬하고, 그 순서를 $V_{\pi(1)}$, $V_{\pi(2)}$ ... $V_{\pi(N)}$이라 하자. 따라서, $Q_{\pi(1)} \geq Q_{\pi(2)} \geq ... \geq Q_{\pi(N)}$. d(k)의 오름차순으로 서버들을 정렬하고, 그 순서를 $S_{\psi(1)}$, $S_{\psi(2)}$ ... $S_{\psi(N)}$이라 하자. 따라서, $d(\psi(1)) \leq d(\psi(2)) \leq ... \leq d(\psi(N))$.

Step 5)

Step 5-1) for i = 1 to N { $T_{\pi(i)} \leftarrow V_{\pi(i)}$ }

Step 5-2) for i = 1 to N { $V_{\psi(i)} \leftarrow T_{\pi(i)}$ }

그림 3. 비디오 배치 방법 2
Fig 3. Video Placement Method 2

비디오 배치 방법 2는 앞선 방법 1과는 달리 두 단계로 동작한다. Step 1 ~ 3은 첫 번째 단계로서 서버 부하 불균형 지수를 고려해서 비디오 배치를 수행한다. Step 4 ~ 5는 두 번째 단계로서 네트워크 부하 지수를 고려해서 비디오 재배치를 한다.

Step 1 ~ 2는 방법 1의 Step 1 ~ 2와 동일하다. Step 3에서는 각 v_i에 대해서 부하가 가장 작은 서버 r_i개를 선택하여 v_i를 배치한다. 비디오 배치 방법 1의 Step 3에서 Δ=0으로 하면 방법 2의 Step 3과 동일하게 된다.

Step 4 ~ 5의 목적은 각 s_k에 배치된 비디오 집합 V_k를 다른 서버에 재배치하여 네트워크 부하 지수인 D 를 현재보다 감소시키는 것이다. 이를 위해서 평균부하가 큰 비디오 집합은 클라이언트와의 거리가 작은 서버에 배치하게 된다. 예를 들어 설명하면 다음과 같다. N=5, Q1=0.1, Q2=0.2, Q3=0.25, Q4=0.4, Q5=0.05, d(1) = 550, d(2) = 600, d(3) = 530, d(4) = 650, d(5) = 500이라 가정하자. Vk를 Qk의 내림차순으로 정리하면, 그 순서는 V4, V3, V2, V1, V5이다. 따라서 π(1) = 4, π(2) = 3, π(3) = 2, π(4) = 1, π(5) = 5이다. 서버들을 d(k)의 오름차순으로 정리하면, 그 순서는 s5, s3, s_1, s2, s4이다. 따라서 ψ(1) = 5, ψ(2) = 3, ψ(3) = 1, ψ(4) = 2, ψ(5) = 4이다. 그래서 s5에는 V4, s3에는 V3, s1에는 V2, s2에는 V1, s4에는 V5를 재배치한다. 비디오 재배치 전후의 D를 비교해 보면 다음과 같다.

	재배치 전	재배치 후
D	592.5	535

Theorem 1: 비디오 배치 방법 2의 Step 4에서 구한 순열 π 와 순열 ψ이 주어져 있을 때, 비디오 재배치들 중에

서 Step 5의 비디오 재배치가 [정의 2]의 네트워크 부하 지수를 최소화한다.

Proof) 비디오 배치 방법 2의 Step 5에서 기술한 비디오 재배치를 $\mathcal{G}0$라고 부르자. $\mathcal{G}0$이 아니면서 네트워크 부하 지수를 최소화하는 비디오 재배치 $\mathcal{G}$가 존재한다고 가정하자. 그러면 $\mathcal{G}$에 대해서는 다음 ① ~ ③이 성립하는 자연수 i, j, m, n이 존재하게 된다.

① $1 \le i, j, m, n \le N$

② $i < j$, $m < n$

③ $V\pi(i)$가 $s\psi(n)$에 배치되고, $V\pi(j)$가 $s\psi(m)$에 배치된다.

$V\pi(i)$를 $s\psi(m)$에 배치하고 $V\pi(j)$를 $s\psi(n)$에 배치되도록 $\mathcal{G}$를 변경하고, 그렇게 변경된 비디오 재배치를 $\mathcal{G}$'라고 하자. $\mathcal{G}$의 네트워크 부하 지수를 $D\mathcal{G}$라고 하고, $\mathcal{G}$'의 네트워크 부하 지수를 $D\mathcal{G}$'라고 하자. 그러면,

$$\begin{aligned} D\mathcal{G} - D\mathcal{G}' &= Q\pi(i)d(\psi(n)) + Q\pi(j)d(\psi(m)) \\ &\quad - Q\pi(i)d(\psi(m)) - Q\pi(j)d(\psi(n)) \\ &= (Q\pi(i) - Q\pi(j))d(\psi(n)) \\ &\quad + (Q\pi(j) - Q\pi(i))d(\psi(m)) \\ &= (Q\pi(i) - Q\pi(j))(d(\psi(n)) - d(\psi(m))) >= 0. \end{aligned}$$

만약 $D\mathcal{G} - D\mathcal{G}' > 0$라면 $\mathcal{G}$가 네트워크 부하 지수를 최소화하는 비디오 재배치라는 가정에 대한 모순이 발생한다. 만약 $D\mathcal{G} = D\mathcal{G}'$일 때, $\mathcal{G}' = \mathcal{G}0$라면 역시 가정에 대한 모순이 발생한다. $D\mathcal{G} = D\mathcal{G}'$이고 $\mathcal{G}' \neq \mathcal{G}0$라면, $\mathcal{G}$'는 네트워크 부하 지수를 최소화하는 또 하나의 비디오 재배치가 된다. 따라서 $\mathcal{G}$가 네트워크 부하 지수를 최소화하는 비디오 재배치라는 가정하의 증명 과정이 그대로 $\mathcal{G}$'에도 적용된다. 끝.

2. 서버 과부하 예측

$\eta_{i,t}$를 시간구간 [0, t] 동안에 발생하는 v_i에 대한 총 사용자 요구 수라고 하자. 집합 S_i $(1 \le i \le M)$을 비디오 v_i가 배치된 서버 집합이라 하자. 앞에서 정의한 $x_{k,i}$로 이용해서 S_i를 정의하면, $S_i = \{s_k | x_{k,i} = 1\}$이다. 시간구간 [0, t] $(0 < t < d_{video})$ 동안에 서버 과부하로 인해서 사용자 요구가 기각되지 않을 확률 $P_{normal}(t)$ 은 다음과 같다. 확률계산의 단순화를 위해서 네트워크 과부하로 인한 사용자 요구 기각은 발생하지 않는다고 가정한다.

$$P_{normal}(t) = \sum_{(n_1,n_2,\ldots,n_M)\in\Omega} P(n_1,n_2,\ldots,n_M)$$

$$\Omega = \{(\eta_{1,t}, \eta_{2,t}, \ldots, \eta_{M,t}) |$$

$$\sum_{i=1}^{M}\eta_{i,t} \le \frac{B_{video}N}{b_{video}},\ \eta_{1,t} \le \frac{B_{video}}{b_{video}}|S_1|,$$

$$\eta_{2,t} \le \frac{B_{video}}{b_{video}}|S_2|, \ldots,\ \eta_{M,t} \le \frac{B_{video}}{b_{video}}|S_M|\}$$

$$P(n_1,n_2,\ldots,n_M) = P_{\text{포이송}}(n_1+n_2+\ldots+n_M)\cdot P_{\text{다항분포}}(n_1,n_2,\ldots,n_M)$$

$$P_{\text{다항분포}}(n_1,n_2,\ldots,n_M) = \frac{(n_1+n_2+\ldots+n_M)!}{n_1!n_2!\ldots n_M!}p_1^{n_1}p_2^{n_2}\ldots p_M^{n_M}$$

$$P_{\text{포이송}}(n_1+n_2+\ldots+n_M) = \frac{e^{-\lambda t}(\lambda t)^{n_1+n_2+\ldots+n_M}}{(n_1+n_2+\ldots+n_M)!}$$

집합 Ω는 [0, t] 동안에 사용자 요구가 기각되지 않을 사용자 요구 패턴의 집합이다. 확률 $P(n_1,n_2,\ldots,n_M)$는 사용자 요구 패턴 $(n_1,n_2,\ldots,n_M) \in \Omega$이 발생할 확률로서, 확률 $P_{\text{포이송}}(n_1+n_2+\ldots+n_M)$와 확률 $P_{\text{다항분포}}(n_1,n_2,\ldots,n_M)$의 곱으로 계산할 수 있다. $P_{\text{포이송}}(n_1+n_2+\ldots+n_M)$는 [0, t] 시간구간 내에 사용자 요구가 총 $n_1+n_2+\ldots+n_M$ 개 발생할 포이송 분포 확률이고, $P_{\text{다항분포}}(n_1,n_2,\ldots,n_M)$는 v_1에 대한 요구가 n_1개, v_2에 대한 요구가 n_2개, ..., v_M에 대한 요구가 n_M개로 구성될 다항분포 확률이다.

$P_{normal}(t)$을 이용해서 비디오 배치와 서버 출력 밴드위스가 서버 과부하에 어떤 영향을 미치는 지를 미리 예측해 봄으로써, 본 절의 확률모델은 비디오 배치들의 성능을 비교하거나 서버 구성에 참고할 수 있을 것이다.

Ⅳ. 실험과 분석

1. 시뮬레이션 환경

실험에 사용된 네트워크는 GT-ITM [10]을 이용하여 구성하였으며 트랜짓-스텁 (transit-stub) 모델을 가진다. 24개의 스텁 도메인이 존재하고 트랜짓 서브네트워크와 스텁 도메인의 노드들은 총 240개로 하였다. 트랜짓 서브네트워크의 각 링크(link)는 평균 450개의 비디오 스트림을 동시에 전송할 수 있고, 스텁 도메인의 각 링크는 평균 150개의 비디오 스트림을 동시에 전송할 수 있다.

비디오 서버의 개수는 8개이고 트랜짓 서브네트워크에 존재하고, 각 서버는 동시에 600개 비디오 스트림을 전송할 수 있다. 비디오의 개수는 80개로 하고, 각 비디오의 길이는 80분이다. 각 서버의 저장 공간을 비디오 10개, 15개 및 30개를 저장할 수 있는 세 경우로 나누어 실험했다. 저장 공간의 크기에 따라서, 비디오의 평균 복제 개수 $\bar{\gamma}$ 는 1.0, 1.5와 3.0 중 한 값이 된다. 비디오들의 인기도는 θ = 0.75인 Zipf 분포를 따르고, 비디오 서비스를 요청하는 사용자 요구의 빈도는 포이송 분포를 따른다.

2. 전송방법

우리는 본 논문의 비디오 배치 방법을 유니캐스트와 P2P 기반의 멀티캐스트 방식인 P2Cast[2] 하에서 실험하였다. 사용자 요구의 기각률은 비디오 배치뿐만이 아니라 비디오 전송 방법에 따라 달라진다. P2Cast는 지금까지 제안된 P2P 기반의 대표적인 멀티캐스트 방식 중 하나이다. [2]에 기술된 P2Cast는 단일 서버 환경에서 동작하도록 제안되었다. 따라서 다수 서버를 가진 VOD 환경을 위해서는 다음과 같은 확장이 필요하다: 더미(dummy) 노드를 루트로 하고 모든 비디오 서버들을 자식으로 하여 초기 멀티캐스팅 트리를 구축한다. 그러면 클라이언트에게 비디오를 전송할 부모 노드나 패칭 서버 노드를 선택을 위해서 기존 알고리즘을 그대로 사용할 수 있다.

3. 실험 결과

본 논문의 비디오 배치 방법과 기존연구[5]를 사용자 요구의 기각률 관점에서 서로 비교하였다. 표 1, 2 및 3은 유니캐스트 전송 하에서의 사용자 요구 기각률을 보여주고, 표 4, 5 및 6은 멀티캐스트 전송 하에서의 사용자 요구 기각률을 보여 준다. 실제 기각률은 네트워크 형상이나 서버 배치 등에 따라서 다소 차이가 날 수 있을 것이다. 표에서 M1과 M2는 각각 비디오 배치 방법 1과 비디오 배치 방법 2를 각각 나타낸다.

M1은 파라미터 Δ=0.5와 Δ=L의 두 경우에 대해서 실험하였다. Δ=L 경우는 Δ에 충분히 큰 값을 지정하고 서버 부하가 미치는 영향을 줄여 네트워크 부하의 최소화에 초점을 맞춘 비디오 배치를 나타낸다. Δ=0.5 경우는 서버 부하 불균형과 네트워크 부하를 모두 고려하게 된다. 사전 실험에 의하면 Δ가 대략 0.5일 때가 사용자 요구의 기각률 면에서 상대적으로 우수하였다. 기존 연구 [5]는 서버의 부하균형에 초점을 맞춘 비디오 배치 방법이다.

표 1. 기각률 비교 (유니캐스트, $\bar{\gamma}$ =1.0)

Table 1. Rejection Ratio Comparison (Unicast, $\bar{\gamma}$ =1.0)

λ	M1, Δ=L	[5]	M1, Δ=0.5	M2
35	1.9	1.2	1.1	1.0
40	9.8	9.5	9.4	9.1
45	13.7	12.2	11.5	11.0
50	18.5	16.8	15.2	14.5
55	22.5	20.8	19.0	18.2

표 2. 기각률 비교 (유니캐스트, $\bar{\gamma}$ =1.5)

Table 2. Rejection Ratio Comparison (Unicast, $\bar{\gamma}$ =1.5)

λ	M1, Δ=L	[5]	M1, Δ=0.5	M2
35	0.1	0	0	0
40	1.8	1.2	0.8	0.6
45	3.5	2.9	2	1.8
50	9.1	7.5	6.8	6.0
55	26.1	23.9	23.7	22.8

표 3. 기각률 비교(유니캐스트, $\bar{\gamma}$ = 3.0)

Table 3. Rejection Ratio Comparison (Unicast, $\bar{\gamma}$ =3.0)

λ	M1, Δ=L	[5]	M1, Δ=0.5	M2
35	0	0	0	0
40	0.9	0.6	0.5	0.3
45	2.5	2.1	1.8	1.6
50	8.5	7.6	7.3	6.5
55	24.2	22.8	22.5	21.8

표 1, 2와 3은 $\bar{\gamma}$ = 1.0, $\bar{\gamma}$ = 1.5와 $\bar{\gamma}$ = 3.0에서의 사용자 요구의 기각률을 각각 보여준다. 클라이언트가 요구

하는 비디오를 가진 서버가 여러 개인 경우에는 가장 가까운 서버가 선택되어 비디오 스트림을 전송한다. 유니캐스트가 가장 초보적인 비디오 전송 방식인 점을 감안하면, 이들 표는 전송 방법이 기각률 개선에 미치는 영향을 최소화한 상태 하에서, 비디오 배치 방법과 기각률간의 관계를 보다 분명하게 보여 주다고 할 수 있다.

사용자 요구의 기각은 서버 과부하로 인해서 서버가 스트림을 전송하지 못하거나 네트워크 과부하로 인해서 서버와 클라이언트 사이에 경로가 확보되지 못하면 발생한다. 실험 결과를 분석해 보면, 다음 같은 몇 가지 주목할 만한 사항을 알 수 있다.

첫째, $\bar{\gamma}$이 높아짐에 따라서 사용자 요구 기각률이 전반적으로 낮아진다. 클라이언트 입장에서 보면 자신이 원하는 비디오를 가진 서버를 보다 가까운 곳에서 찾을 수 있기 때문에 결과적으로 네트워크 부하가 준다.

둘째, 모든 $\bar{\gamma}$에 대해서 M2가 기존연구나 M1보다 대체적으로 우수함을 알 수 있다. 또한, 기존연구나 Δ=L인 M1보다는 Δ=0.5인 M1이 대체적으로 우수하다. 기존연구는 서버 부하의 균형에만 치중하고 Δ=L인 M1은 네트워크 부하 감소에만 치중을 하게 된다. 기존연구와 Δ=L인 M1에 비해서 M2와 Δ=0.5인 M1이 우수한 이유는 서버 부하를 균등하게 분산시키면서 네트워크 부하 감소를 함께 고려하기 때문이라 판단된다.

셋째, M2는 서버 부하 균형을 먼저 고려하여 비디오를 배치한 후 네트워크 부하를 고려하여 비디오를 재배치한다. Δ=0.5인 M1은 각 비디오를 배치함에 있어서 서버 부하 균형과 동시에 네트워크 부하를 고려하도록 한다. M2와 Δ=0.5인 M1의 서버 불균형 지수 및 네트워크 부하 지수를 계산해 보면, 전자는 후자에 비해서 서버 불균형 지수는 낮고 네트워크 부하 지수는 다소 높다. M2가 Δ=0.5인 M1보다 낮은 기각률을 보이는 것은 비디오 배치에서 서버 부하 불균형 지수 L에 대한 고려가 네트워크 부하 지수 D에 대한 고려보다 중요하다는 것을 나타낸다. [정의 2]에 의하면, 서버와 클라이언트들 간의 거리 합 (즉, $d(k)$)이 서버 사이에 차이가 없다면 네트워크 부하 지수는 평균 부하 (즉, l_k)로부터 주된 영향을 받는다. 본 실험에서는 서버들을 네트워크상 지역적으로 편중되게 설치되어 서버와 클라이언트간의 거리 합이 서버 사이에 큰 차이는 나지 않았다.

넷째, 기존연구와 Δ=0.5인 M1을 서로 비교하면, 후자의 경우 서버 부하 불균형은 다소 희생하더라도 클라이언트들과의 거리가 가까운 서버에 인기도가 높은 비디오를 배치함으로써 네트워크 과부하로 인한 사용자 요구 기각률을 낮추게 된다. 클라이언트들과의 거리가 서버 사이에 큰 차이가 없는 경우, 두 방법의 사용자 요구 기각률 차이는 크지 않고 후자가 전자보다 큰 경우도 발생하였다.

다섯째, 기존연구와 Δ=L인 M1의 공통점은 서버 부하나 네트워크 부하 중에 하나만 고려한다는 것이다. 하지만 전자가 후자보다 사용자 요구 기각률 면에서 대체적으로 우수한 이유는 서버들이 대체적으로 네트워크 지역들 간에는 균등하게 설치되어 있기 때문이다. 서버 평균 부하가 균등하도록 비디오 배치가 되면, 사용자의 비디오 요구들이 여러 서버들에 균등하게 나누어서 도착하게 되어 서버 과부하 확률이 낮아진다. 또한 서버들로부터 출발하는 비디오 스트림들도 네트워크 지역 간에 균등하게 나누어져 특정 네트워크 지역의 과부하로 인해서 서버와 클라이언트사이의 경로가 존재하지 않게 되는 네트워크 과부하의 확률도 낮아지기 때문이다. 전자가 후자보다 우수하다는 것은 앞에서 설명한 M2와 Δ=0.5인 M1의 비교에서와 같이, 비디오 배치에서 서버 부하 불균형 지수에 대한 고려가 네트워크 부하 지수에 대한 고려보다 기각률 면에서 중요하다는 것을 보여주고 있다.

여섯째, 도착률이 낮아지거나 높아지면, 기각률 차이가 줄어들거나 기각률 대비 차이의 비율이 줄어드는 경향을 보인다. 도착률이 낮을 때에는 서버와 네트워크 부하가 상대적으로 높지 않아 기각률이 낮아지므로 기각률 차이가 줄어든 것이고, 도착률이 높을 때는 서버와 네트워크의 과부하로 인해 기각률이 높아져서 비디오 배치 방법이 기각률에 미치는 영향이 크지 않기 때문이다.

표 4. 기각률 비교(멀티캐스트, $\bar{\gamma}$ = 1.0)

Table 4. Rejection Ratio Comparison (Multicast, $\bar{\gamma}$=1.0)

λ	M1, Δ=L	[5]	M1, Δ=0.5	M2
50	2.5	2.1	2	1.8
60	14.3	13.9	11.6	11.5
70	22.3	21.3	20.1	19.5
80	29.2	28.3	29	26.5

표 5. 기각률 비교(멀티캐스트, $\bar{\gamma}$ = 1.5)

Table 5. Rejection Ratio Comparison (Multicast,$\bar{\gamma}$=1.5)

λ	M1, Δ=L	[5]	M1, Δ=0.5	M2
50	0	0	0	0
60	6.6	6.4	4.3	3.4
70	16.6	15.5	15.2	14.5
80	25.6	22.7	22	21.2

표 6. 기각률 비교(멀티캐스트, $\bar{\gamma}$ = 3.0)

Table 6. Rejection Ratio Comparison (Multicast,$\bar{\gamma}$=3.0)

λ	M1, Δ=L	[5]	M1, Δ=0.5	M2
50	0	0	0	0
60	4.2	4	2.6	2
70	18.2	17.5	16.5	16.4
80	23.1	22.8	20.1	20.1

표 4, 5와 6은 멀티캐스트 전송 하에서 사용자 요구의 기각률을 보여준다. 멀티캐스트 환경에서는 비디오 스트림을 클라이언트 노드로부터 전송받을 수 있다. 따라서 비디오를 요구하는 클라이언트와 비디오를 전송하는 클라이언트가 같은 스텁 도메인에 존재하는 경우 서버가 비디오를 전송하는 경우보다 네트워크상의 트래픽을 현저하게 감소한다. 유니캐스트 전송에서 지적한 사항들이 멀티캐스트 전송에서의 실험에서도 나타난다.

V. 결 론

비디오 서버 클러스터 환경하의 VOD 시스템에서는 여러 비디오 서버를 네트워크상에 분산 배치하고 적절한 개수로 복제된 비디오들을 분산 저장한다. 본 논문에서는 비디오 서버 클러스터 기반의 VOD 시스템에서의 비디오 배치 문제를 정의하였고, 정의된 문제에 대해서 효과적인 해결 방법 두 개를 제안하였다. 제안된 방법의 특징은 비디오 서버들 간의 부하 균형을 최대한 유지하면서 클라이언트와 가까운 서버에 인기도 합이 큰 비디오 집합을 배치함으로써 네트워크상의 부하를 최소화하는 것이다.

제안된 방법들을 다양한 비디오 복제률에 대해서 유니캐스트와 P2P 기반의 멀티캐스팅 전송 하에서 실험하였다. 실험 결과, 서버 부하 균형만을 고려하거나 네트워크 부하 최소화만을 고려한 비디오 배치보다는 이들을 모두 고려한 비디오 배치가 사용자 요구 기각률 면에서 비디오 복제률과 전송방법에 무관하게 우수하였다. 사용자 요구 기각률은 사용자가 지각하는 VOD 시스템 성능 지수들 중 특히 중요한 지수이다. 서버 부하 균형만을 고려한 비디오 배치의 실험에는 기존 연구를 활용하였다. 또한, 서버 불균형 지수만을 고려한 비디오 배치가 네트워크 부하 지수만을 고려한 사용자 요구 기각률 면에서 우수하였다. 이런 실험 결과는 비디오 배치에서 일반적으로 서버 부하 불균형에 대한 고려가 네트워크 부하에 대한 고려보다 중요한 역할을 한다는 것을 나타낸다. 추후 연구로 제안된 방법을 애니캐스트 전송을 포함하는 다양한 VOD 시스템 환경 하에서 실험하여 발전시켜 나가고자 한다.

참 고 문 헌

[1] M. Karlsson, C. Karamanolis, and M. Mahalingam, A framework for evaluating replica placement algorithms, *TR HPL-2002, HP Laboratories*, July 2002.

[2] Y. Guo, K. Suh, J. Kurose, "P2Cast : Peer-to-peer patching scheme for VOD service," *Proceedings of Conference of ACM WWW*, pp.301-309, Budapest, Hungary, May 2003.

[3] W-C. Feng, B. Krishnaswami, A. Prabhudev, "Proactive buffer management for the streamed delivery of stored video," *Proceedings of Conference of ACM Multimedia*, pp. 285-290, Bristol, UK, September 1998.

[4] T. Ibarkai, N. Katoh, *Resource allocation problem - Algorithmic approaches*, The MIT Press, 1988.

[5] C. Z. Xu, X. Zhou, "Optimal video replication and placement on a cluster of video-on-demand servers", *Proceedings of the Int'l Conference on Parallel Processing*, pp. 547 - 555, British Columbia, Canada, August 2002.

[6] Y. Chen, R. H. Katz, J. D. Kubiatowicz, "Dynamic

replica placement for scalable content delivery, " *LNCS 2429*, pp.306 - 318, Springer-Verlag, March 2002.

[7] A. Dan, D. Sitaram, "A online video placement policy based on bandwidth to space ratio(BSR)," *Proceedings of Conference of ACM SIGMOD*, pp.376-385, San Jose, United States, May 1995.

[8] B.-J. Ko, D. Rubenstein, "Distributed, self-stabilizing placement of replicated resources in emerging networks," *IEEE/ACM Trans. on Networking*, 13(3), pp.476 - 487, 2005.

[9] 윤수미, 김상철, "VOD 환경에서 다수 서버상의 비디오 배치 방법," 정보산업공학연구소 논문집, 한국외국어대학교, 제 12호, pp.100-110, 2007.

[10] Ellen W. Zegura, Ken Calvent, S. Bhattacharjee, "How to model an Internet," *Proceedings of IEEE INFOCOM*, pp.594-602, San Francisco, United Sates, April 1996.

Acknowledgement: 본 논문은 2008년도 한국외국어대학교 학술연구비 지원에 의하여 연구되었음

저자 소개

김 상 철(정회원)

- 1983년 서울대학교 컴퓨터공학과 학사 졸업
- 1994년 미시간주립대학교 컴퓨터공학과 박사학위
- 1994년 ~ 현재 한국외국어대학교 컴퓨터공학과 교수

<주관심분야 : 멀티미디어 시스템, 멀티미디어 콘텐츠, 임베디드 소프트웨어>