

차세대 집중형 항행 서비스를 위한 이동체 웹 서버 클러스터 설계

Vehicular Web Server Cluster Design for Next Generation Centralized Navigation Services

김용호*, 김영용*

Ronny Yongho Kim*, Young Yong Kim*

요 약

HTTP나 오디오/비디오 스트리밍 서비스는 미래 집중형 항행 시스템에서 사용될 수 있는 가장 유력한 서비스들로 이러한 서비스들을 제공하기 위해서 서비스 사업자들은 웹 서버들의 묶음인 웹 서버 클러스터를 사용하는 것이 권장된다. 본 논문에서는 사용자의 등급에 따라 우선순위를 고려한 차등 접속을 제공할 수 있는 웹 서버 클러스터에 대한 수학적 모델링과 성능 분석 그리고 설계 방법을 제공한다. 몇 가지 가능한 시나리오를 통해 사용자 등급에 따라 차등 서비스를 제공해 줄 수 있는 효과적인 웹 서버 클러스터 방법을 대기행렬 이론을 이용한 성능 모델링을 통하여 분석해 보았다. 또한 효과적인 웹 서버 클러스터 방법을 제공해 줄 수 있는 트래픽 부하 조절기 설계에 대한 효율적인 기준도 제시하였다. 마지막으로 제안된 수학적 모델의 정확도를 컴퓨터 모의실험을 통하여 검증하였다.

Abstract

HTTP or audio/video streaming services are good candidates for future centralized navigation system and in order to provide stability for such services, service providers use a cluster of web servers. In this paper, we provide the criteria for web server cluster design of vehicular users with consideration of differentiated access per different user classes. Several feasible scenarios are examined and their performance analysis using queueing theory is presented to provide the foundation for web server cluster design using traffic load balancer. Through the thorough analysis, efficient criteria for traffic load balancer design is derived. In order to satisfy users' service requirements, priority services controlled by traffic load balancer are considered and analyzed. We also provide the evaluation of the accuracy of the analytical model through simulation.

Key words : HTTP, Multimedia, Navigation Server, Queueing, Streaming, Web Server, Web Traffic

I. 서 론

World Wide Web (WWW)은 이제 우리 일상생활에

서 없어서는 안 되는 필수 요소가 되었고 Hypertext Transfer Protocol(HTTP)은 웹상의 트래픽 중 많은 부분을 차지하게 되었다. 또한 이러한 인터넷의 폭발적

* 연세대학교 전기전자공학(Department of Electrical and Electronics Eng., Yonsei University)

· 제1저자 (First Author) : 김용호

· 투고일자 : 2009년 8월 29일

· 심사(수정)일자 : 2009년 8월 31일 (수정일자 : 2009년 9월 28일)

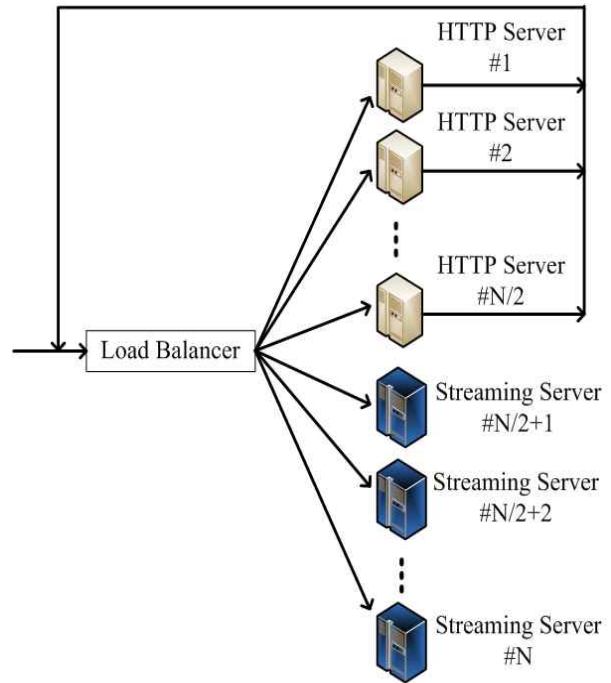
· 게재일자 : 2009년 10월 30일

인 사용량 증대는 오디오/비디오 스트리밍 (Audio/Video Streaming) 서비스 요구 증대를 동반했다. 더 많은 이동체 사용자들이 이동 중에 다양한 무선 이동 통신 접속 방법을 이용하여 인터넷을 접속할 수 있게 되었다. 이러한 다양한 무선 접속 방식을 통한 이동체 사용자들의 인터넷 접속 환경을 이용하면 별도의 항행 서비스망의 구성없이 최신 교통 상황 정보를 실시간으로 제공할 수 있는 미래 집중형 항행 시스템에 응용 가능할 것으로 예상된다. 또한 이러한 항행 시스템은 HTTP를 통해 사용자와의 양방향 통신 환경을 제공할 수 있고 오디오/비디오 스트리밍 서비스를 통해 음성 안내를 동반한 영상 정보도 제공할 수 있다.

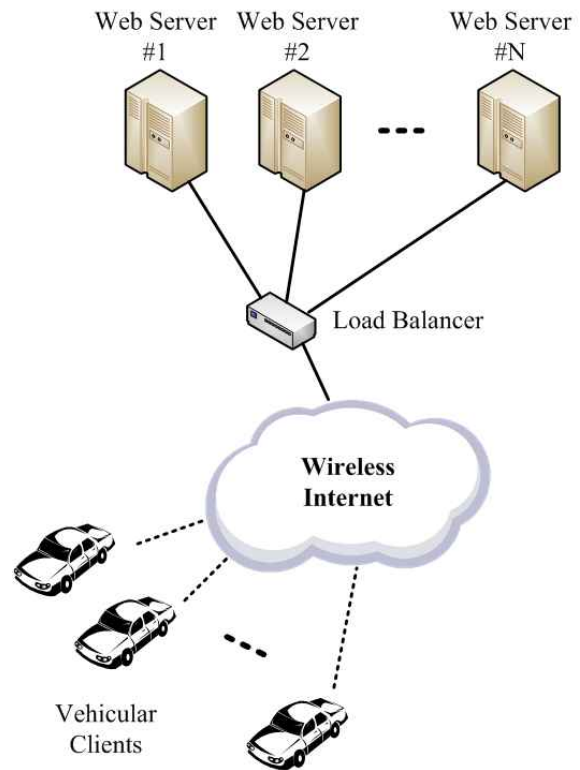
많은 웹서비스 제공자들이 웹 서버 클러스터 방식으로 서비스를 제공하고 있다 [1]. 이러한 클러스터 방식의 웹 서버 환경에서는 클러스터 전단에 위치한 트래픽 부하 조절기(traffic load balancer)가 사용자로부터의 요청을 서버들로 적절히 분산해 줌으로써 모든 서버가 동일한 양의 부하를 담당하도록 할 수 있다 [2]. 만약에 트래픽 부하 조절기가 HTTP트래픽 또는 스트리밍 트래픽 등과 같은 트래픽의 종류를 구분할 수 있는 경우에는 요청을 해당 트래픽 종류를 처리할 수 있는 서버로 전달할 수 있고 [3] 동적인 부하 분산 처리와 쉽게 확장 가능한 웹 서버 설계가 가능하다.

본 논문에서는 실용적인 시나리오들의 분석을 통해 이동체 사용자들이 HTTP나 오디오/비디오 스트리밍과 같은 서비스들을 이용할 때 사용자 등급에 따라 차등 접속을 제공할 수 있는 웹 서버 클러스터에 대한 모델링과 성능 분석 그리고 설계 방법을 제안한다. 본 연구는 트래픽 부하 조절기를 채용한 웹 서버 클러스터를 설계하는데 기초를 제공할 것으로 기대한다.

본 논문은 다음과 같이 구성되어 있다. II장에서는 확장 가능한 웹 서버 클러스터의 수학적 분석 모델을 제공하고, III장에서는 제공된 수학적 분석 모델을 이용하여 우선순위를 고려한 차등 서비스 시나리오를 포함한 다양한 시나리오들에 대한 성능 분석을 하였다. IV장에서는 유도된 수식들을 이용한 성능 평가를 제공했다. V장을 통해 논문의 결론에 대해 논의 한다.



(a) 확장 가능한 웹 서버 클러스터



(b) 분석 모델

그림 1. 확장 가능한 웹 서버 클러스터와 분석 모델
Fig. 1. Scalable Web Server Cluster and Analytical Model

II. 시스템 모델

확장 가능한 웹 서버들을 실현하기 위해 그림 1 (a)에 보여 지는 바와 같이 웹 서버들의 전단에 트래픽 부하 조절기가 위치하는 망 구성을 고려한다. 동적인 부하 분산과 확장 가능한 웹 서버 설계를 위해서 트래픽 부하 조절기는 트래픽 종류와 부하 상태에 따라서 트래픽을 분산 할 수 있다고 가정하기로 한다. 분석을 하고자 하는 웹 서버들로 구성된 클러스터는 N개의 서버들로 구성되어 있으며 스트리밍 서버들은 사용자들이 요청하는 스트리밍 트래픽 량에 따라 동적으로 구성 가능하도록 설정한다. 트래픽 부하 조절기는 트래픽의 종류에 따라서 라운드 로빈(Round Robin), 대기열 기반 방법 등의 다양한 분배 정책을 사용할 수 있다. 서비스를 요청하는 트래픽은 사용자 등급에 따라서 유료 요청과 무료 요청으로 나눈다.

다음은 본 논문의 분석 모델에서 사용되는 파라미터들이다.

- N : 서버의 수
- N_{nf} : 유료 사용자만을 위한 전용 서버의 수
- P_f : 무료 서비스 요청률
- λ_h : HTTP 서비스를 요청하는 비율의 합
- λ_s : 스트리밍 서비스를 요청하는 비율의 합
- μ_h : HTTP 서비스의 평균 서비스율
- μ_s : 스트리밍 서비스의 평균 서비스율
- P_h : HTTP 서비스를 통해 부가적으로 스트리밍 서비스를 요청하는 확률

본 논문의 분석 모델에서는 분석을 간단히 하기 위해서 웹 서버들 중의 1/2이 스트리밍 트래픽을 위한 서버로 구성되어 있다고 가정한다. 또한 제안된 모델에서는 트래픽 부하 조절기는 서비스를 요청하는 트래픽을 동일한 확률로 배분한다고 가정한다. 본 논문에서 사용하는 분산 분석 모델은 그림 1 (b)과 같고 이와 관련된 HTTP 서비스 요청을 위한 도착률과 스트리밍 서비스 요청을 위한 도착률은 다음과 같이 나타낼 수 있다.

$$\text{HTTP: } 2\lambda_h/N$$

$$\text{- 무료 요청: } 2(\lambda_h \times P_f)/N$$

$$\text{- 유료 요청: } 2(\lambda_h \times (1 - P_f))/N$$

$$\text{스트리밍: } 2(\lambda_s + P_h \times \lambda_h)/N$$

$$\text{- 무료 요청: } 2((\lambda_s + P_h \times \lambda_h) \times P_f)/N$$

$$\text{- 유료 요청: } 2((\lambda_s + P_h \times \lambda_h) \times (1 - P_f))/N$$

HTTP서비스는 보통 고정된 서비스 시간으로 처리되고 스트리밍 서비스에 소요되는 평균 서비스율은 지수분포로 분석 될 수 있다. HTTP서비스와 스트리밍 서비스에 필요한 평균 서비스 시간은 각각 $1/\mu_h$ 와 $1/\mu_s$ 로 표기하기로 한다.

III. 성능분석

우선 HTTP 서비스에 관해 분석을 해 보기로 한다. 분석에서는 유료 사용자에게 대한 우선순위가 고려되었다. 분석에 사용되는 모델은 II장에서 제시된 도착률과 서비스율을 가지는 M/G/n 대기행렬 모델로 분석될 수 있다. 평균 지연 시간은 M/G/n 대기행렬 시스템에서 수식 (1)의 Pollaczek-Khinchine(P-K) 평균값식을 이용하여 구할 수 있다 [4], [5].

$$\overline{w_d} = \overline{\chi} + \overline{\chi} \frac{\rho(C_b^2 + 1)}{2(1 - \rho)} \quad (1)$$

HTTP 서비스 요청에 대해 평균 서비스 시간 $\overline{\chi}$ 는 $1/\mu_h$ 이고, 사용률 ρ 는 $2\lambda_h/N\mu_h$ 그리고 분산계수 C_b^2 는 0인 값을 사용할 수 있다. 또한 스트리밍 서비스 요청에 대해서 평균 서비스 시간 $\overline{\chi}$ 는 $1/\mu_s$, 사용률 ρ 는 $2(\lambda_s + \lambda_h \times P_h)/N\mu_s$ 그리고 분산계수 C_b^2 는 1인 값을 사용할 수 있다. 따라서 HTTP 서비스 요청이 HTTP서버에서 처리되는데 걸리는 HTTP 지연 시간 $\overline{w_h}$ 와 스트리밍 서비스 요청이 스트리밍 서버에서 처리 되는데 걸리는 스트리밍 지연 시간 $\overline{w_s}$ 는 각각 다음과 같다.

표 1. 각 서버에서의 평균지연, 도착률, 사용률
Table 1. Average delay, arrival rate and utilization in each type of servers

평균 지연 시간			
HTTP ($\bar{\chi} = 1/\mu_h, C_b^2 = 0$)			
무료 사용자 ($\rho_f = \frac{2(\lambda_h \times P_h)}{(N - N_{nf})\mu_h}$)		유료 사용자 ($\rho_{nf} = \frac{2(\lambda_h \times (1 - P_h))}{N_{nf} \times \mu_h}$)	
$\frac{\mu_h \times (N - N_{nf}) - (\lambda_h \times P_f)}{\mu_h (\mu_h \times (N - N_{nf}) - 2(\lambda_h \times P_f))}$		$\frac{\mu_h \times N_{nf} - (\lambda_h \times (1 - P_f))}{\mu_h (\mu_h \times N_{nf} - 2(\lambda_h \times (1 - P_f)))}$	
스트리밍 ($\bar{\chi} = 1/\mu_s, C_b^2 = 1$)			
무료 사용자 ($\rho_f = \frac{2(\lambda_s + \lambda_h \times P_h) \times P_f}{(N - N_{nf})\mu_s}$)		유료 사용자 ($\rho_{nf} = \frac{2(\lambda_s + \lambda_h \times P_h) \times (1 - P_f)}{N_{nf} \times \mu_s}$)	
$\frac{\mu_s \times (N - N_{nf})}{\mu_s (\mu_s \times (N - N_{nf}) - 2(\lambda_s + \lambda_h \times P_h) \times P_f)}$		$\frac{\mu_s \times N_{nf}}{\mu_s (\mu_s \times N_{nf} - 2(\lambda_s + \lambda_h \times P_h) \times (1 - P_f))}$	
도착률과 사용률			
HTTP			
무료 사용자		유료 사용자	
도착률	$\frac{2\lambda_h}{N} P_f$	도착률	$\frac{2\lambda_h}{N} (1 - P_f)$
사용률	$\frac{2\lambda_h}{N\mu_h} P_f$	사용률	$\frac{2\lambda_h}{N\mu_h} (1 - P_f)$
스트리밍			
무료 사용자		유료 사용자	
도착률	$\frac{2(\lambda_s + \lambda_h \times P_h)}{N} P_f$	도착률	$\frac{2(\lambda_s + \lambda_h \times P_h)}{N} (1 - P_f)$
사용률	$\frac{2(\lambda_s + \lambda_h \times P_h)}{N\mu_s} P_f$	사용률	$\frac{2(\lambda_s + \lambda_h \times P_h)}{N\mu_s} (1 - P_f)$

$$w_h = \frac{N\mu_h - \lambda_h}{\mu_h (N\mu_h - 2\lambda_h)} \quad (2)$$

그리고

$$w_s = \frac{N}{N\mu_h - 2(\lambda_s + \lambda_h \times P_h)} \quad (3)$$

이제 유료 사용자에게 대한 우선 서비스를 고려해 보도록 하자. 무료 사용자보다 유료 사용자에게 우선 서비스를 제공하는 방법에는 여러 가지가 있을 수 있

다. 그 중 한 가지 방법은 유료 사용자들만을 처리하는 전용 서버들을 두는 것이다. 분석의 간략화를 위해서 $N/2$ 개의 서버가 HTTP용으로 그리고 $N_{nf}/2$ 개의 서버가 유료 사용자를 위한 전용 HTTP용, 스트리밍용 서버로 할당되었다고 가정한다. 유료 사용자를 위한 전용 서버가 할당이 되었기 때문에 각 서버는 한 가지 종류의 우선순위만 가진다. 식 (1)을 사용하여 각 서버 종류에 대한 평균 지연 시간을 구할 수 있다. 그 결과는 표 1에 요약되어 있다.

우선 서비스 분석에는 잘 알려진 대기 행렬 이론

표 2. 비선점 서비스 방식에서 서비스 우선순위를 고려한 클래스별 평균지연시간
 Table 2. Average delay per service class with consideration of priority for non preemptive policy

HTTP	
유료 사용자	$\bar{w}_{h_nf} = \frac{\lambda_h}{\mu_h(N\mu_h - 2\lambda_h(1 - P_f))}$
무료 사용자	$\bar{w}_{h_f} = \frac{N \times \lambda_h}{(N\mu_h - 2\lambda_h(1 - P_f)) \times (N\mu_h - 2\lambda_h)}$
스트리밍	
유료 사용자	$\bar{w}_{s_nf} = \frac{2(\lambda_s + \lambda_h \times P_h)}{\mu_s(N\mu_h - 2(\lambda_s + \lambda_h \times P_h)(1 - P_f))}$
무료 사용자	$\bar{w}_{s_f} = \frac{2N(\lambda_s + \lambda_h \times P_h)}{(N\mu_h - 2(\lambda_s + \lambda_h \times P_h)(1 - P_f))(N\mu_s - 2(\lambda_s + \lambda_h \times P_h))}$

이 사용되었다. 두 클래스를(유료 사용자, 무료 사용자) 가지는 비선점(non preemptive) 우선순위 방식에서는 잔여 서비스 시간 R과 각 클래스의 평균대기 지연 (queueing delay) -- \bar{w}_f : 낮은 우선순위 (무료 가입자), \bar{w}_{nf} : 높은 우선순위 (유료 가입자) -- 은 다음과 같이 표현 가능하다 [6].

$$R_{h_nf} = \frac{\lambda_h \times (1 - P_f)}{N \times \mu_h^2} \tag{8}$$

$$R_{s_f} = \frac{2(\lambda_s + \lambda_h \times P_h)}{N \times \mu_s^2} \tag{9}$$

$$R_{s_nf} = \frac{2(\lambda_s + \lambda_h \times P_h)(1 - P_f)}{N \times \mu_s^2} \tag{10}$$

$$R = \frac{1}{2} \sum_{i=1}^2 \lambda_i \chi_i^2 \tag{4}$$

$$\bar{w}_{nf} = \frac{R}{1 - \rho_2} \tag{5}$$

$$\bar{w}_f = \frac{R}{(1 - \rho_2) \times (1 - \rho_2 - \rho_1)} \tag{6}$$

식 (4), (5), (6)을 사용하여 각 클래스의 HTTP서비스와 스트리밍 서비스에 대한 평균 지연 시간을 계산할 수 있다. 표 2에 계산한 식들을 정리해 놓았다.

선점(preemptive) 우선순위 방식에서는 각 클래스의 잔여 서비스 시간 -- R_f : 낮은 우선순위 (무료 가입자), R_{nf} : 높은 우선순위 (유료 가입자) -- 이 HTTP와 스트리밍 서비스들에 대해 다음과 같이 각각 다르다.

$$R_{h_f} = \frac{\lambda_h}{N \times \mu_h^2} \tag{7}$$

식 (7), (8), (9), (10)을 이용하여 선점 방식을 이용할 때 평균 지연 시간을 계산한 식들을 표 3에 정리해 놓았다.

IV. 성능 평가

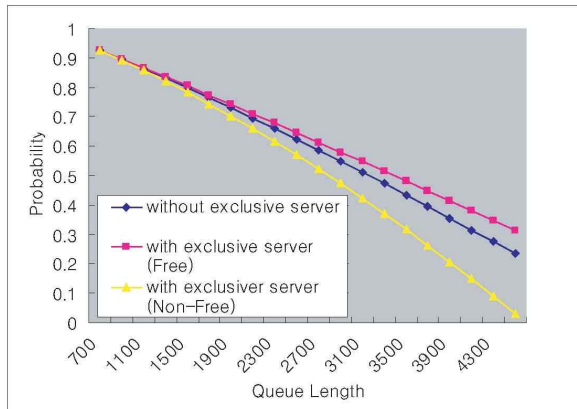
HTTP서비스의 호 블로킹 확률은 대기 큐의 길이에 영향을 받고 스트리밍 서비스의 호 블로킹 확률은 도착률에 영향을 받는다. 따라서 본 논문에서HTTP 서비스는 도착률을 고정하고 대기 큐의 길이를 조정하며 스트리밍 서비스는 대기 큐의 길이를 고정하고 도착률을 변경하면서 성능 분석을 실시하였다. 호 블로킹에 대한 결과는 그림 2에 도시하였다. 기대한 바와 같이 유료 사용자들이 무료 사용자들보다 낮은 호 블로킹 확률을 가짐을 확인할 수 있다. 각 서비스별 전용 서버를 사용하지 않고 서버들을 공용으로 사

표 3. 선점 서비스 방식에서 서비스 우선순위를 고려한 클래스별 평균지연시간

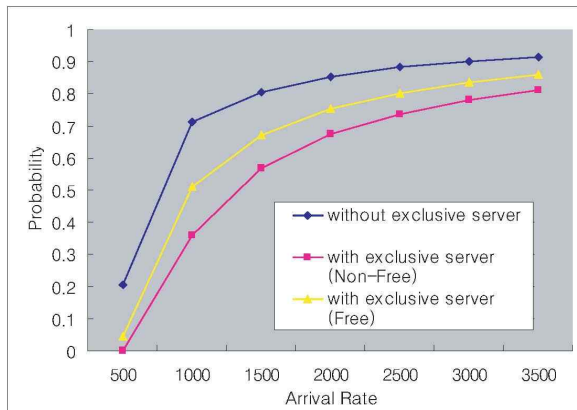
Table 3. Average delay per service class with consideration of priority for preemptive policy

HTTP	
유료 사용자	$\bar{w}_{h_nf-p} = \frac{N\mu_h - \lambda_h(1 - P_f)}{\mu_h(N\mu_h - 2\lambda_h(1 - P_f))}$
무료 사용자	$\bar{w}_{h_f-p} = \frac{N \times (N\mu_h - \lambda_h)}{(N\mu_h - 2\lambda_h(1 - P_f)) \times (N\mu_h - 2\lambda_h)}$
스트리밍	
유료 사용자	$\bar{w}_{s_nf-p} = \frac{N}{N\mu_h - 2(\lambda_s + \lambda_h \times P_h)(1 - P_f)}$
무료 사용자	$\bar{w}_{s_f-p} = \frac{N^2 \mu_s}{(N\mu_h - 2(\lambda_s + \lambda_h \times P_h)(1 - P_f))(N\mu_s - 2(\lambda_s + \lambda_h \times P_h))}$

용하는 경우는 HTTP와 스트리밍의 서비스 종류에 따라 다른 결과를 보인다.

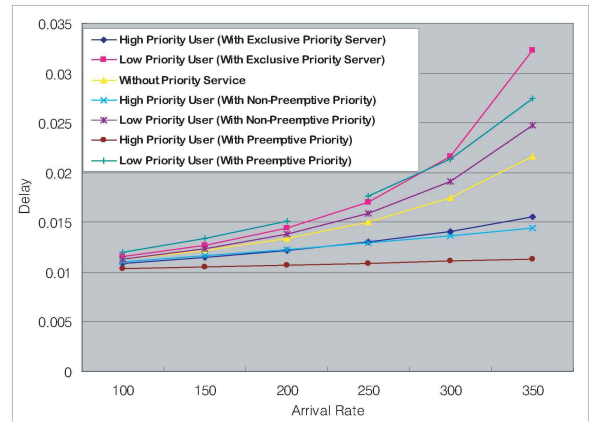


(a) 호 블로킹: HTTP 서비스

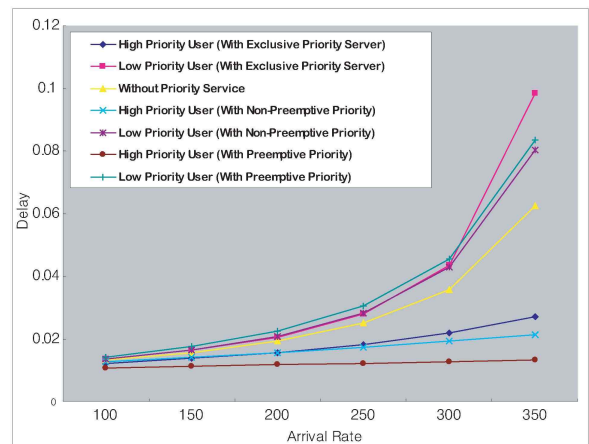


(b) 호 블로킹: 스트리밍 서비스

그림 2. 호 블로킹 확률 분석
Fig. 2. Call blocking probability analysis



(a) 지연 시간: HTTP 서비스



(b) 지연 시간: 스트리밍 서비스

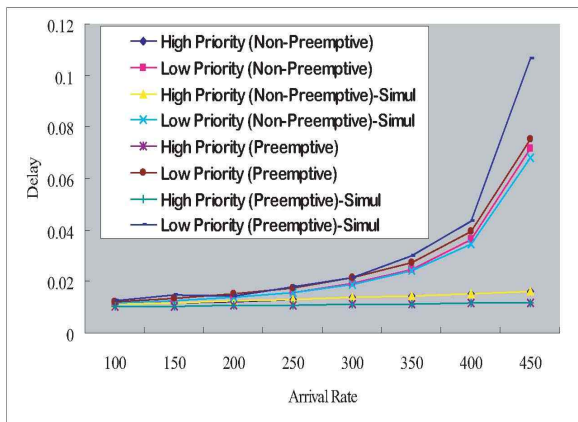
그림 3. 지연 성능 분석
Fig. 3. Delay performance analysis

다양한 시나리오에서 지연 시간에 대해 검토되고 분석 되었으며 결과들은 그림 3에 도시 하였다. 높은

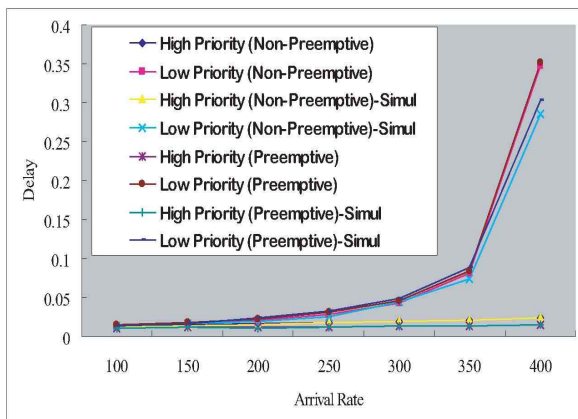
우선순위의 사용자들은 낮은 우선순위 사용자들에 비해서 좀 더 짧은 지연 시간을 가지는 것을 확인할 수 있다. 분석은 P_h 는 0.7, N 은 10 그리고 N_{nf} 는 5로 가정되어 이루어 졌다. 그림 4는 수학적 분석 모델과 컴퓨터 모의실험의 비교 결과를 보여주는 것으로 컴퓨터 모의실험 결과 값과 수학적 분석 모델 결과 값들이 거의 일치하는 것을 알 수 있다. 컴퓨터 모의실험은 그림 1과 같은 설정을 C언어로 구현하여 실험하였다. 모의실험에서 사용된 도착률과 서비스율은 분석모델에서 사용된 값과 동일한 값이 사용되었다. 다만, 선점 우선순위 대기행렬 결과가 수학적 분석 모델 결과 값보다 약간 큰데 이것은 컴퓨터 모의실험에서는 선점 재시작 (preemptive resume) 정책이 사용된 반면에 수학적 분석 모델에서는 선점 반복 (preemptive repeat) 정책이 사용되었기 때문이다.

V. 결 론

본 논문을 통해 오늘날의 다양한 무선 인터넷 접속 방법을 이용하여 구성할 수 있는 미래 집중형 항행 시스템의 항행 웹 서버 클러스터의 성능 분석이 제공 되었다. 다양한 웹 서비스 제공 시나리오들을 대기 행렬 이론을 이용한 수학적 분석 모델로 모델링하고 호 블로킹율과 지연 성능 관점에서 분석 하였다. 분석에서는 오늘날 인터넷 트래픽의 주를 이루는 HTTP와 스트리밍을 이용하여 분석을 실시하였다. 사용자 등급이 높은 사용자에게 차등서비스를 제공하기 위해서 두 가지 방법이 검토되었다. 하나는 우선 등급 사용자만이 접속할 수 있는 전용 서버를 두는 것이고 다른 하나는 우선순위 대기 정책을 적용하는 것이다. 우선순위 대기 정책에 대해서는 비선점 정책과 선점 정책 모두에 대해 성능 분석을 실시하였다. 컴퓨터 모의실험을 통해 제안된 성능 분석 모델들의 정확성을 검증하였다. 본 논문에서 제공된 분석 방법은 미래에 사용 가능한 집중형 항행망의 성능을 분석하고 동적으로 최적화하는데 초석이 될 수 있을 것이라 사료된다.



(a) 성능 검증: HTTP 서비스



(b) 성능 검증: 스트리밍 서비스

그림 4. 컴퓨터 모의실험을 통한 성능 검증

Fig. 4. Performance evaluation through computer simulation

감사의 글

본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (IITA-2008-C1090-0801-0038)

참 고 문 헌

- [1] D. A. Menasce, "Trade-offs in Designing Web Clusters", *IEEE Internet Computing*, vol 6, Issue 5, Sept.-Oct. 2002
- [2] V. Cardellini, M. Colajanni, P.S. Yu, "Dynamic Load Balancing on Web-Server Systems", *IEEE Internet Computing*, vol 3, Issue 3, May.-June. 1999
- [4] Dutta, K.; Anindya Datta; VanderMeer, D.; Thomas, H.; Ramamritham, K., "ReDAL: An Efficient and Practical Request Distribution Technique for Application Server Clusters," *Parallel and*

Distributed Systems, *IEEE Transactions on* , vol.18, no.11, pp.1516-1528, Nov. 2007

- [5] L. Kleinrock, "Queueing Systems Vol I: Theory", Wiley, 1975
- [6] Albert Leon-Garcia, "Probability and Random Processes for Electrical Engineering", Addison-Wesley, 1994
- [7] Schroeder, T.; Goddard, S.; Ramamurthy, B., "Scalable Web server clustering technologies," *Network, IEEE* , vol.14, no.3, pp.38-45, May/Jun 2000

김 용 호 (金龍浩)



1998년 : 인하대학교 전자공학과 (공학사)

2003년 : 연세대학교 전기전자공학과 (공학석사)

2005년 ~현재 : 연세대학교 전기전자공학과 박사과정

관심분야 : 차세대 이동통신 시스템, 차세대 인터넷, 멀티미디어 통신, 네트워크 코딩

김 영 용 (金泳龍)



1991년 : 서울대학교 전자공학과 (공학사)

1993년 : 서울대학교 전자공학과 (공학 석사)

2000년 : The University of Texas at Austin 전기 및 컴퓨터 공학과 (공학 박사)

1998년 ~ 2000년 : Telcordia Technologies 연구원

2000년 ~ 2005년 : 연세대학교 전기전자공학과 조교수

2005년 ~ 현재 : 연세대학교 전기전자공학과 부교수

관심분야 : 차세대 이동통신 시스템, 차세대인터넷, 애드혹 및 센서 네트워크