

이동통신망 관리용 운용시스템 설계에 관한 연구*

A study on the design of operations system for managing the mobile communication network

하기종** · 이천희***

Ki-Jong Hah · Cheon-Hee Yi

Abstract

In this paper, operations system was designed for the centralization of data processing of various state information from the facilities of mobile communication network. And the system performance experimental system module was measured and analyzed from the designed experimental system module.

The configuration of system design was presented with the centralized type to monite and control the facilities of mobile communication network in the central office. The communication process design of the internal system was implemented with the resource of message queue having a excellent transmission ability for processing of a great quantity of information in the inter-process communication among communication resources of UNIX system. The process with a server function from the internal communication processes was constructed with a single server or a double server according to the quantity of operations and implemented with the policy of the presented server. And then, we have measured performance elements in accordance with the change of input parameters from the designed experimental module: response time, waiting time, buffer length, the maximum quantity existing in message queue. And from these results, we have compared and analyzed the system state each server algorithm according to performance variations.

1. 서론

정보 발생원으로 부터 다량의 정보를 신속 정확하게

전달하기 위해 각종 산업 및 군사 분야에서 관련 정보들을 실시간으로 처리할 수 있도록 다양한 기술 개발이 활성화되고 있다.

* 본 연구는 청주대학교 안식년 기간동안 이루어진 것임

** 영동전문대학 정보통신과

*** 청주대학교 전자공학과

이와 같은 응용의 전형적인 분야는 실험 측정 및 감시, 통신장비 및 전자교환기, 프로세스 제어, 미사일 유도, 항공 트래픽 제어, 생산공정, 산업 로봇 등 다양하게 활용된다. 본 논문에서는 현재 국내에서 수요가 급증할 뿐만 아니라 서비스가 다양화되는 이동 무선통신망을 효율적으로 운용 및 관리하기 위해 국내 환경에 적합한 이동 무선통신망에 대한 통합 운용시스템 설계에 대해 연구하였다.

통합 운용시스템의 설계를 위한 시스템 구성은 지역별로 분산된 이동 무선통신망의 설비들을 중앙에서 감시 및 제어하는 집중형 시스템을 제안하였다.

시스템의 내부 설계는 감시 정보들을 실시간으로 처리하기 위해 UNIX를 사용하는 워크스테이션을 집중형 시스템의 성능모델로 채택 하여 시스템의 통신설비(Communication facility)들 중 프로세스간의 통신(IPC: Inter-Process Communication)에 적합한 메시지 전달 버퍼를 이용한 내부 통신 프로세스 설계를 시도하였으며, 설계될 실험모델로부터 입력 패킷정보들의 도착율(Arrival rate) 변화에 따른 패킷정보들의 서비스 효율을 높이기 위해 단일서버와 이중서버 프로세스의 구성을 제안하였다. 여기서 이중서버의 구성은 시스템의 부하를 가능한 최소화하면서 시스템의 성능 및 신뢰성을 제공하기 위해 하나의 서버는 항상 서비스를 제공하다가, 패킷정보가 버퍼 내에 일정한 비율이상을 점유할 경우에 다른 하나의 서버가 동시에 서비스를 제공할 수 있도록 하였다. 이와같이 구성된 서버들로부터 입력 패킷정보의 도착율 변화에 따른 서비스율(Service rate)에 의해 결정되는 메시지 전달 버퍼에 대한 평균 응답시간(Mean response time), 평균 버퍼길이(Mean buffer length) 및 최대 정보량(Quantity of maximum information) 등의 시스템 성능평가 요소들을 산출하여 비교 분석 하므로써 시스템 성능에 영향을 미치는 요소와 시스템 설계 및 구성에서 고려되어야 할 특성들을 알 수 있었다.

2. 이동 무선통신망 관리용 통합 운용시스템

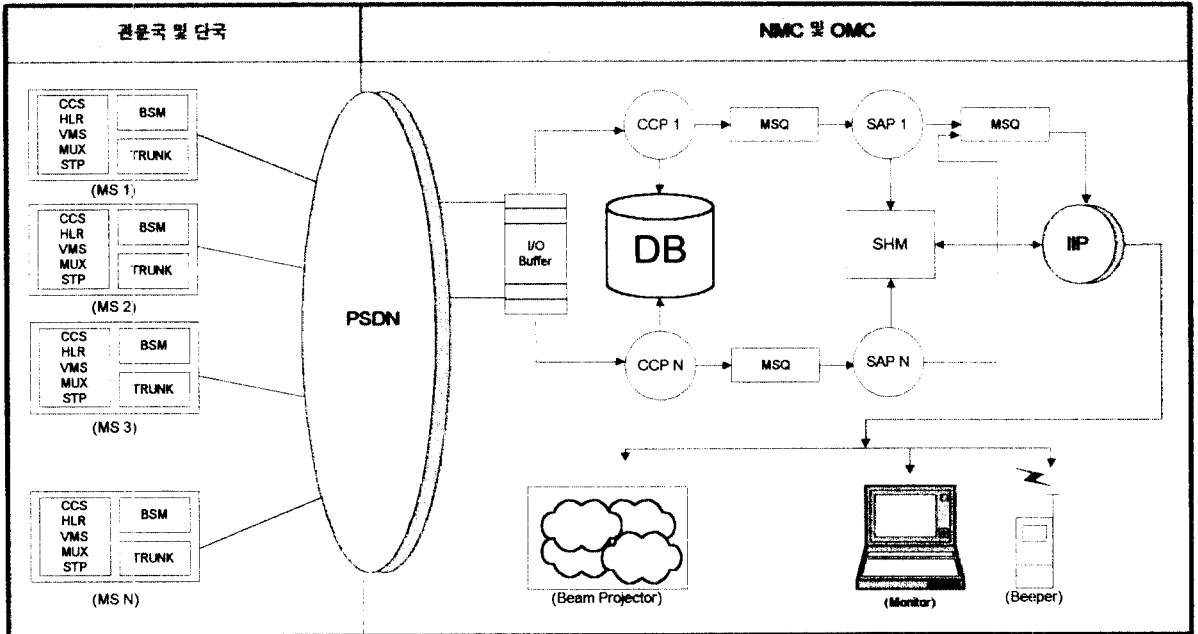
통합 운용시스템은 이동 무선통신망의 구성요소인 이동통신 시스템(MS: Mobile System)으로부터 발생하는 운용관리 및 상태정보들을 집중화하여 시스템 상태

를 실시간으로 원격 감시 및 분석할 수 있도록 구성되어야 한다. 따라서 지역적으로 분산된 MS 시스템들을 집중국에 수용하여 정보들을 통합 처리하기 위한 구성 및 특성은 다음과 같다.

통합 운용시스템의 전체 구성도는 <그림 1>과 같이 원격지의 기지국(BS: Base Station)과 이동통신 교환국(MSC: Mobile System Center)에 설치되는 다중 MS 시스템, 운용보전국(OMC: Operations Maintenance Center)에 설치될 통합 운용시스템간의 상호연결을 위한 통신 회선망으로 구성된다. 다수의 입력원에 해당하는 지역적으로 분산된 MS의 원격 채널로부터 발생하는 정보들을 신호 변환장치와 통신망을 통해 중앙의 통합 운용시스템으로 전달된다. 입력된 정보들은 시스템 내의 수집분류 프로세스(CCP: Collection & Classification Process)에 의해 수집 및 분류되어 중요한 상태정보는 수집분류 프로세스에서 상태분석 프로세스(SAP: State-Analytic Process)로 실시간 전송이 될 수 있도록 UNIX 시스템의 통신설비들 중 메시지 전달 버퍼를 사용하였다.^{[1][2]}

상태분석 프로세스는 시스템 상태정보를 분석 및 저장하고, 시스템의 상태를 실시간으로 감시 및 추적하기 위해 상태를 클래스(Class)별로 분류한 후 공유 메모리(Shared memory) 영역에 할당하여 시스템의 동적인(Dynamic) 상황 정보를 실시간으로 변경 관리할 수 있도록 하며, 상태 클래스 정보는 버퍼를 통해 정보 통합 프로세스(IIP: Information Integration Process)로 전달한다.

그리고 시스템 구성 특징은 첫째, 입력 정보원에 대해 수집분류 프로세스인 CCP를 일 대 일(One-to-one)로 매핑(Mapping) 시킨다. 둘째, 프로세스간의 다량의 정보 전송을 실시간으로 처리하기 위해 메시지 전달 버퍼를 시스템 모델링에 사용한다. 셋째, 버퍼의 넘침(Overflow)을 방지하면서 시스템 부하를 가능한 최소화하기 위한 방안으로 임계형 이중 프로세스(Dual process)를 사용한다. 따라서 이와같은 시스템 특징을 구비한 통합 운용시스템이 구성될 경우 기대되는 효과는 상태정보의 집중 감시 및 관리에 따른 시스템 장애에 대한 신속 정확한 판단 및 조치가 가능하다.

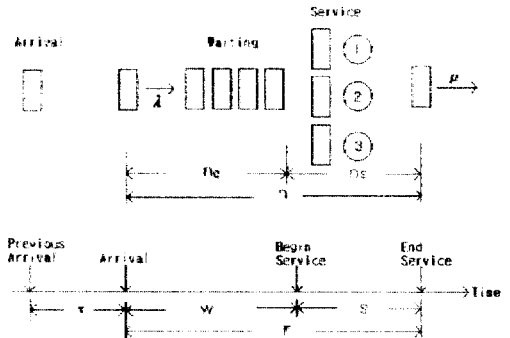


〈그림 1〉 이동통신망 관리용 운용시스템의 구성도

3. 시스템 모델링을 위한 버퍼의 동작

IPC 메시지 전달 버퍼에 대해 시간에 따라 전개되는 동작은 〈그림 2〉로부터 다음과 같이 버퍼의 분석에 사용되는 중요한 변수들로 정의할 수 있다.³⁾⁴⁾

- τ : 도착간격 시간(Interarrival time), 두 개의 연속적인 패킷정보의 도착시간 간격
 - λ : 평균 도착율(Mean arrival rate), $\lambda = 1 / E[\tau]$ (여기서, $E[\tau]$ 는 τ 의 평균값)
 - s : 정보당 서비스 시간(Service time)
 - μ : 서버당 평균 서비스율(Mean service rate), $\mu = 1 / E[s]$
 - n : 시스템내 패킷정보의 수로 버퍼길이를 나타냄
 - n_q : 서비스를 받기 위해 기다리는 패킷정보의 수
 - n_s : 서비스를 받고 있는 패킷정보의 수
 - γ : 응답시간(Response time)
 - ω : 대기시간(Waiting time), 즉 도착 시간과 서비스 시작 순간의 시간간격
- 여기서 λ 와 μ 를 제외한 모든 변수들은 랜덤(Random) 변수들이며, 이들의 관계는 다음과 같다.



〈그림 2〉 메시지 전달 버퍼의 분석에 사용되는 랜덤변수

만약 시스템 내에 정보의 수가 계속적으로 증가하여 무한 수가 된다면 시스템이 불안정해진다. 따라서 시스템의 안정된 동작을 위해 다음과 같이 평균 도착율은 평균 서비스율 보다 작아야만 한다.

$$\lambda < m\mu \quad (\text{여기서, } m \text{은 서버의 수}) \quad (1)$$

정보의 수는 버퍼 내의 정보의 수와 서비스 받는 정보의 수의 합과 항상 일치하며..

$$n = n_q + n_s \quad (2)$$

정보들이 버퍼의 부족으로 인해 손실(Loss) 되지 않는다면, 시스템내의 평균 정보의 수($E[n]$)는 다음과 같이 시스템의 평균 응답시간($E[\gamma]$)에, 그리고 버퍼내 평균 정보 수($E[n_q]$)는 평균 대기 시간($E[\omega]$)과 관련되어진다.

$$E[n] = \lambda \times E[\gamma] \quad (3)$$

$$E[n_q] = \lambda \times E[\omega] \quad (4)$$

메시지 전달 버퍼를 사용하는 시스템내 패킷정보에 의해 소요되는 시간은 다음과 같이 버퍼내 대기시간과 서비스를 받는 시간에 의해 결정된다.

$$E[\gamma] = E[\omega] \times E[s] \quad (5)$$

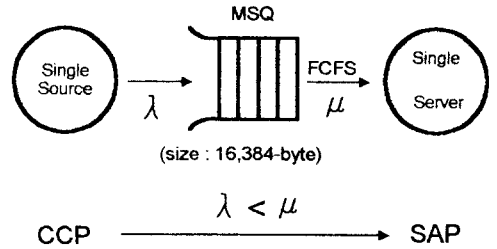
4. 시스템의 설계 및 성능분석

제안된 통합 운용시스템의 구성에서 각각의 프로세스는 메시지 전달 버퍼의 입력 정보량에 따라 크게 단일서버와 이중서버로 구성하였다. 본 논문에서는 이들 서버의 정책을 제안하여 서버 프로세서를 설계하여 그 성능을 비교 분석하였다.

(1) 서버 프로세스의 정책

시스템의 구성에서 첫째단의 버퍼(MSQ)는 <그림 3>과 같이 입력 정보원인 CCP와 정보서비스를 지원하는 SAP를 버퍼 중심으로 전후단에 각각 일대일(One-to-one)로 연결하여 실시간 정보전송을 지원한다. 여기서 버퍼에 대해 서버의 기능을 제공하는 SAP프로세스는 버퍼의 정보원인 CCP가 단일 프로세스로 구성되어 있기 때문에 (1)식의 $\lambda < m\mu$ 조건하에서 입력 패킷정보들의 손실(Loss)없이 서비스를 제공하기 위해서 단일 서버에 의해 만족될 수 있다. 따라서 단일서버 정책은 표 1에서 제시한 바와 같이 버퍼에 새로운 패킷정보가 도착했을 경우 현재 서버가 idle 상태이면 FCFS(First-Come First-Out) 규칙에 따라 서비스를 제공하고, 반면에 busy 상태이면 패킷정보를 서버에 의해 서비스 될 때까지 버퍼에 대기시키는 일련의 과정을 반복적으로 수행하도록 한다.

그리고 시스템의 구성에서 둘째단의 버퍼는 <그림 4>와 같이 전단에 다중 SAP가 연결되어 평균 도착율은 입력 패킷정보들의 포아송 스트림의 결합이 결국



<그림 3> 단일서버의 구성

포아송 스트림의 결과가 되어 $\lambda = \sum_{i=1}^k \lambda_i$ 가 된다.^{[8][9]} 이와같은 패킷정보의 도착율에 대해 서버의 기능을 담당하는 2개의 IIP프로세서의 서비스율은 아래의 식을 만족해야만 한다.

서버가 동종일 경우에는 평균 도착율과 서비스율의 관계는 (6)식과 같다.

$$\lambda = \sum_{i=1}^2 \lambda_i < m\mu, \quad (\mu_1 = \mu_2) \quad (6)$$

서버가 이종일 경우에는 (7)식과 같이 나타낼 수 있다.^[10]

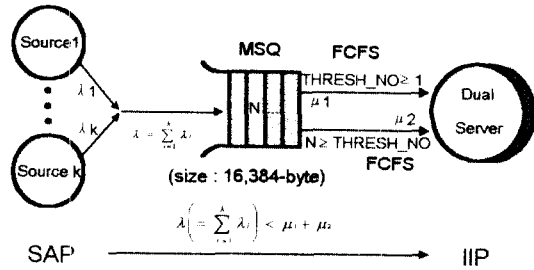
$$\lambda = \sum_{i=1}^2 \lambda_i < \mu_1 + \mu_2, \quad (\mu_2 \leq \mu_1) \quad (7)$$

본 연구에서는 버퍼의 입력원이 다중 프로세스일 경우 시스템 자체의 과부하(Overload)에 의한 버퍼내의 실시간 정보들의 손실(Loss)방지와 지연시간을 최소화하기 위한 대책으로 IIP 프로세스에 대해 이중의 이중서버 기법을 사용하여 서버 프로세서를 설계하였다.

이중서버로 구성된 IIP 프로세스는 표 2와 같이 하나의 서버(S_1)는 메시지 큐 버퍼내의 패킷정보의 수가 PACKET_NO \geq 1일 경우 서비스를 통해 정보를 빼가고 (Departure), 반면에 다른 한 서버(S_2)는 PACKET_NO \geq 임계값(THRESH_NO)일 경우 버퍼내의 패킷정보들이 서비스될 수 있도록 하였다.

(2) 실험 및 성능분석

앞에서 제시된 <그림 1>의 이동통신망의 구성으로부터 프로세스간 상호 정보통신시 실시간으로 정보를



〈그림 4〉 이중서버의 구성

〈표 1〉 단일서버 정책

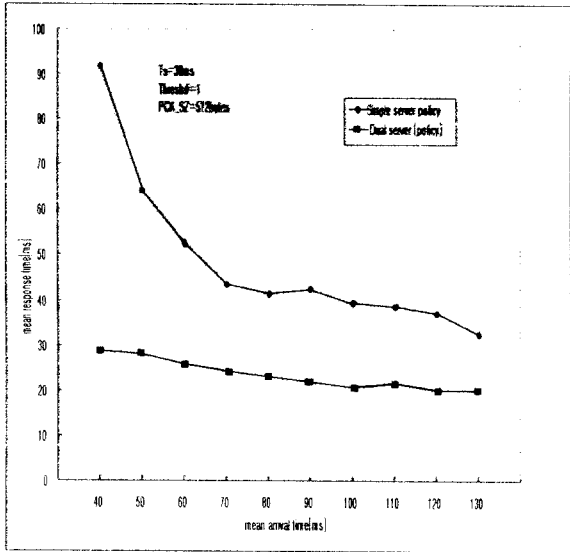
SINGLE_SERVER(N=0,packet_no)
1 THRESH_NO ← N 2 srvc_type ← FCFS 3 get msq_id 4 QSZ ← sizeof(msgbuf) 5 while rcv_msgcnt < packet_no 6 do if MSQ_num > THRESH_NO 7 then if S1=idle 8 then S1(msq_id, srvc_type) 9 else if S1=busy 10 then PACKET_WAIT(msq_id)
S1(msq_id, srvc_type)
1 receive packet with FCFS discipline 2 rcv_msgcnt ← rcv_msgcnt+1 3 rcv_tim[rcv_msgcnt] ← current_milisec 4 max_bytes ← max_cbytes from MSQ state 5 PCK_SND(msq_id, poisson_tim)
PACKET_WAIT(msq_id)
1 packet waiting in MSQ 2 get poisson_tim 3 Timer In Milisec(poisson_tim) 4 tot_tim ← tot_tim + poisson_tim 5 get MSQ state
PCK_SND(msq_id, poisson_tim)
1 classify state_MSQs into class 1-4 2 snd_msgcnt ← snd_msgcnt+1 3 snd_tion[snd_msgcnt] ← current_milisec 4 send packet by msq_id

〈표 2〉 이중서버 정책

DUAL_SERVER(N, packet_no)
1 THRESH_NO ← N 2 srvc_type ← FCFS 3 get msq_id 4 QSZ ← sizeof(msgbuf) 5 while rcv_msgcnt < packet_no 6 do if MSQ_num > THRESH_NO 7 then if S1=busy and S2=idle 8 then S2(msq_id, srvc_type) 9 else if S1=S2=busy 10 then PACKET_WAIT(msq_id) 11 else if (S1=S2=idle) or 12 (S1=idle and S2=busy) 13 then S1(msq_id, srvc_type) 14 else if S1=idle 15 then S1(msq_id, srvc_type) 16 else if S1=busy 17 then PACKET_WAIT(msq_id)
S1(msq_id, srvc_type), S2(msq_id, srvc_type)
1 receive packet with FCFS discipline 2 rcv_msgcnt ← rcv_msgcnt+1 3 rcv_tim[rcv_msgcnt] ← current_milisec 4 max_bytes ← max_cbyte from MSQ state
PACKET_WAIT(msq_id)
1 packet waiting in MSQ 2 get poisson_tim 3 Timer In Milisec(poisson_tim) 4 tot_tim ← tot_tim + poisson_tim 5 get MSQ state

처리하는 메시지 전달 버퍼는 실제 시스템에서 유한 크기의 N 값을 가지는 모델을 채택하였다. 실험모듈은 입력 패킷정보들을 충분히 수용할 수 있을 뿐만 아니라, 입력된 패킷정보들을 실시간으로 처리할 수 있도록 시스템 구현환경을 설정하기 위해 시스템 커널 파라미터(Kernel parameter)중 버퍼 크기 파라미터를 16,384 [bytes], 프로세스의 실시간 처리를 위해 프로세스 스케줄링(Process scheduling)에 관련되는 시간간격(Time

slice) 파라미터를 사용 시스템에서 최소 단위인 20[ms]로 각각 설정하여 커널을 튜닝(Tuning)해 재생성 하였으며^{[12][13]}, 이를 기반으로하여 실험모듈을 구현하였다. 실험모듈로 부터 각각의 정보원과 서버 프로세스에 대한 패킷정보의 도착 및 서비스 분포는 포아송 프로세스에 따라 발생되도록 하였고, 이들은 평균 입력시간에 의해 변경 설정될 수 있도록 하였다.

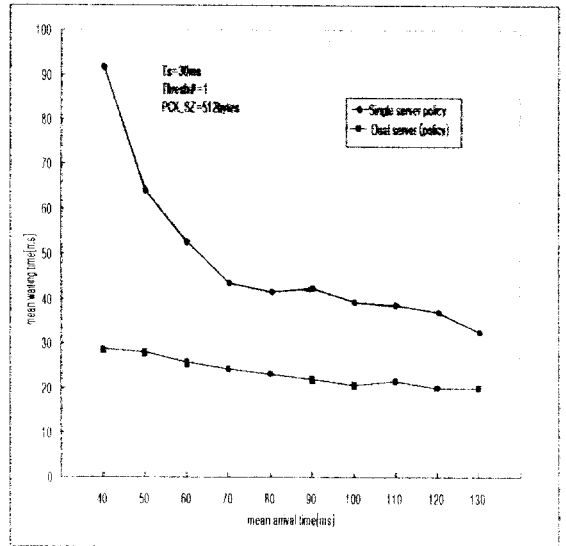


〈그림 5〉 서버정책의 응답시간 비교

〈그림 5〉는 단일서버와 이중서버 정책의 성능을 비교하기 위하여 버퍼에 입력되는 패킷정보의 사이즈 PCK_SZ=512bytes, 이중서버에서 버퍼의 임계값 Thresh#=1, 평균 서비스시간 Ts=30ms 으로 각각 설정하여 평균 도착시간 Ta = 40ms에서 10ms 단위로 변화시키면서 각각의 정책에 대해 측정된 응답시간을 나타낸다. 단일서버일 경우 패킷정보의 도착율이 증가함에 따라 응답시간에 대한 증가폭의 변화가 크고, 반면에 도착율이 점차 감소함에 따라 변화가 없이 거의 일정함을 볼 수 있다. 즉, 버퍼 이용율(Utilization rate)이 증가함에 따라 응답시간에 미치는 영향이 증가하고, 감소함에 따라 응답시간에 영향을 미치지 않음을 알 수 있다. 그리고 이중서버의 경우는 두 개의 서버가 입력 패킷정보에 대한 부하를 분담하므로 도착율의 변화에 따른

응답시간이 단일서버 정책에 비해서 거의 변화가 없이 안정된 값을 가짐을 볼 수 있다. 그림에서 도착율이 가장 높은 Ta=40ms 인 경우 주어진 입력조건에 대한 단일서버 정책에 의한 응답시간이 91.869ms, 이중서버는 28.129ms 이었다. 따라서 단일서버 정책에 비해 응답시간을 70% 정도 줄일 수 있음을 알 수 있다. 그러므로 실험모듈의 구성에서 IIP 프로세스와 같이 단일 버퍼에 입력되는 다중 패킷정보 처리 분야에서는 단일서버 정책보다는 이중서버 정책을 사용하므로써 이용율의 증가에 따른 시스템의 패킷정보 처리성능을 보상할 수 있음을 보여준다.

〈그림 6〉은 단일서버와 이중서버 정책에 대한 대기시간을 나타낸 것이다. 이때 시스템의 입력 파라미터는 패킷정보 사이즈 PCK_SZ = 512bytes, 버퍼의 임계값 Thresh#=1, 평균 서비스시간 Ts=30ms, 평균 도착시간 Ta=40ms에서 10ms단위로 변화시키면서 각각 설정하였다.

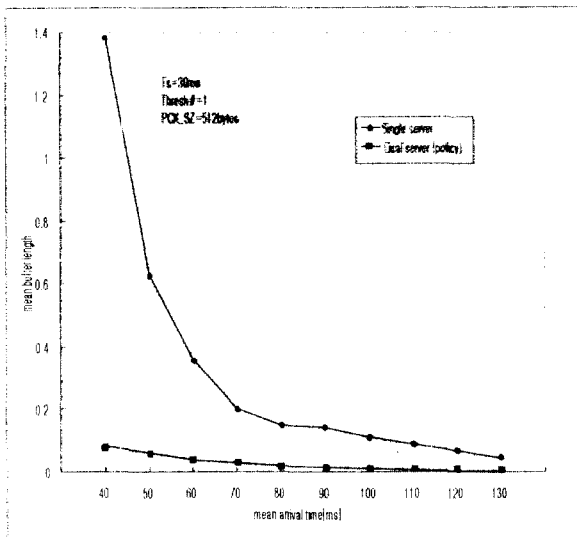


〈그림 6〉 서버정책의 대기시간 비교

단일서버 정책은 Ta=70ms를 기준으로 하여 도착율이 증가함에 따라 대기시간의 상승폭에 대한 변화가 크게 나타남을 볼 수 있고, 이것은 시스템의 동작속도의 척도가 되는 응답시간에 직접적인 영향을 초래하

로서 시스템의 성능을 저하시킬 수 있음을 알 수 있다. 이와같은 대기시간의 영향을 줄이기 위해서는 서비스율($\mu=1/T_s$)을 증가 시키므로서 해결할 수 있다. 그러나 사용된 시스템의 서비스율을 높이기 위해서는 앞에서 기술한 바와 같이 커널 파라미터중 프로세스 스케줄링에 관련되는 time slice 파라미터를 조정해야하나, 시스템에 설정할 수 있는 최소 단위가 한정되어 있으므로 제약을 받는다. 따라서 이와같은 문제의 해결점은 실험모듈의 구성에서 IIP 프로세스와 같이 이중서버 정책을 사용하여 버퍼에 대한 서비스율을 높임으로서 해결할 수 있다. 그림에서와 같이 $T_a=40ms$ 인 경우 이중서버 정책(8.825ms)을 사용하므로서 단일서버 정책의 대기시간(61.87ms)에 비해 88%정도 줄일 수 있다.

<그림 7>은 단일서버와 이중서버 정책에 대해 패킷 정보의 도착율 변화에 따른 버퍼길이를 나타낸다. 이때 시스템의 입력 파라미터는 패킷정보 사이즈 $PCK_SZ=512bytes$, 버퍼 임계값 $Thresh\#=1$, 평균 서비스시간 $T_s=30ms$, 평균 도착시간 $T_a=40ms$ 에서 10ms 단위로 변화시키면서 각각 설정하였다.



<그림 7> 서버정책의 버퍼길이 비교

그림에서 단일서버 정책은 패킷정보의 도착율이 높을수록 버퍼내에 패킷정보들의 점유율이 증가하여 버

퍼길이가 점차적으로 증가함을 볼 수 있다. 이것은 결과적으로 시스템의 블라킹에 직접적인 영향을 초래하여 수용된 시스템의 중요한 실시간 상태정보들을 유실하게 된다. 따라서 이와같은 버퍼길이의 증가에 따른 시스템의 블라킹율을 줄이기 위해서는 그림에서와 같이 이중서버 정책을 사용하여 버퍼에 대한 서비스율을 높임으로서 해결할 수 있다. 그림으로부터 $T_a=40ms$ 인 경우 이중서버 정책들의 버퍼길이(0.0844)는 단일서버 정책의 버퍼길이(1.3809)에 비해 94%정도 줄일 수 있음을 볼 수 있다.

그리고 시스템 실행동안 버퍼에 존재하는 최대 패킷 정보량은 단일서버 정책인 경우 표 3과 같이 패킷정보 사이즈의 증가와 서비스율 ($1/T_s$)의 감소에 따라 증가하고, 이중서버 정책인 경우는 표 4에서와 같이 임계값의 증가에 따라 증가함을 볼 수 있고, 이것은 시스템의 블라킹(Overflow)율을 증가 시킴을 의미한다. 시스템의 블라킹은 표에서와 같이 단일서버 정책의 경우 패킷크기의 증가에 따라 발생확률이 높으며, 이중서버 정책의 경우 패킷크기와 임계치가 높을수록 블라킹율이 증가함을 알 수 있다. 따라서 버퍼의 최대 패킷정보량은 통합 운용시스템 설계시 시스템의 버퍼 규모에 대한 효율적인 활용측면과 블라킹 대책에 중요한 파라미터가 된다. 예를 들면 시스템의 정보원으로부터 2048bytes 단위의 패킷정보가 평균 100ms 단위로 도착하고, 단일서버에 의해 입력된 패킷정보를 30ms 단위로 서비스할 경우 표 3으로 부터 버퍼의 사이즈는 최소한 10,240bytes 이상으로 튜닝해야 입력된 패킷정보를 손실(Loss)하지 않고 실시간으로 처리할 수 있음을 보이고 있다.

5. 결론

본 연구에서는 이동 무선통신망과 관련설비들에 대한 종합적인 운용 및 관리를 통해 상태정보를 파악하여 시스템의 성능개선, 데이터 관리 및 장애에 신속히 대처 할 수 있는 통합 정보처리 체제를 위한 제안된 시스템의 구성 및 입력 패킷정보의 서비스에 대한 서버정책들을 시스템 모델링에 적용하여 그 성능을 분석하였다.

시스템의 실험모듈은 입력 패킷정보들의 도착율 정

〈표 3〉 단일서버에 대한 메시지 큐 상의 최대 정보량

(단위 : byte)

서비스 시간(T_s)	패킷크기 [byte]	도착시간 (T_a) [ms]						
		40	50	60	70	80	90	100
30[ms]	512	7168	7168	4608	4608	4096	3584	3072
	1024	15360	13311	8192	8192	8192	8192	7168
	2048	OVFL	OVFL	14336	14336	12288	12288	10240
40[ms]	512		7680	5632	5120	5120	4096	3584
	1024		16384	12264	9216	8192	8192	7168
	2048		OVFL	OVFL	OVFL	14336	12288	12288

* OVFL : Overflow

〈표 4〉 이중서버 정책에 대한 최대 정보량

도착시간 임계값	40 ms	50 ms	60 ms	70 ms	80 ms	90 ms	100 ms
1	10240	10240	10240	10240	8192	8192	8192
2	10240	10240	10240	10240	8192	8192	8192
3	14336	12288	12288	10240	10240	10240	10240
4	16384	14336	12288	12288	12288	10240	10240
5	16384	16384	14336	12288	12288	12288	10240
6	OVFL	OVFL	14336	12288	12288	12288	10240
7	OVFL	OVFL	14336	14336	14336	12288	12288

도에 따라 서비스 효율을 높이기 위해 서버 프로세스를 단일서버와 이중서버로 구성하였고, 이중서버의 구성에서 서버정책은 임계형 서버정책과 우선권 서버정책을 사용하여 각각의 서버 프로세스에 의한 메시지 큐의 버퍼 성능을 측정하여 비교 분석하였다. 따라서 본 연구에서 관찰된 성능분석 특성은 다음과 같다.

· 실험모듈의 구성에서 IIP 프로세스와 같이

단일 버퍼에 입력되는 다중 패킷정보처리 분야에서는 서비스율을 높이기 위해 단일서버 정책보다는 제안된 이중서버의 정책을 사용함으로써 이용율의 증가에 따른 시스템의 응답속도 및 블락킹율을 개선할 수 있었다.

· 시스템 모델에서 다중 정보원 프로세스로부터 버퍼에 입력된 패킷정보를 서비스하는 이중서버의 정책에 대한 응답시간은 실시간 정보처리에 중요한 성능 파라미터로서 서비스율 $\mu_1 + \mu_2$ 를 증가시키거나 임계값을 감소시키므로서 응답시간에 대한 성능이 개선됨을 알 수 있었다. 또한, 버퍼 길이도 임계값의 증가와

서비스율의 감소에 따라 증가하여 버퍼 넘침에 의한 시스템의 성능에 영향을 미칠 수 있음을 예측할 수 있었다.

· 버퍼내에 존재하는 최대 패킷정보량은 메시지 큐의 버퍼 사이즈에 대한 시스템의 블락킹을 예측 및 개선하는데 기초자료로 활용할 수 있다.

· 시스템의 블락킹은 이중서버의 정책에는 패킷크기와 임계치가 높을수록 블락킹율이 증가함을 알 수 있었다.

본 연구에서는 이와같이 설계된 시스템 실험모듈로부터 실험적 입력 파라미터 값의 조정을 통해 정보처리 성능을 다양하게 실험하므로써 시스템의 성능변화에 대한 결과를 분석할 수 있었다.

참고문헌

- [1] Keith Haviland, Ben Salama, "UNIX system Programming", Addison- Wesley Publishing Company, 1987.
- [2] W. Richard Stevens, "UNIX Network Programming", Prentice Hall, 1991.
- [3] M. H. MacDougall, "Simulating Computer Systems", MIT Press, 1987.
- [4] Raj Jain, "The Art of Computer Systems Performance Analysis" Digital Equipment Corporation.
- [5] Gordon D. and Harris C., "Closed Queueing Systems with Exponential Servers", Operations Research, Vol. 15(2), pp254-265, 1985.
- [6] Thomas S. Heines, "Buffer Behavior in Computer Communication Systems", IEEE Trans. on Computers,

Vol C-28(8), pp573-576, 1979.

[7] John D. C. Little, "A Proof for the Queuing Formula : $L = \lambda W$ ", Operations Research, Vol.9(3), pp383-387, 1961.

[8] Nicolas D. Georganas, "Buffer Behavior with Poisson Arrivals and Bulk Geometric Service", IEEE Trans. on Communication, Vol. 35, 1987.

[9] J. Hsu, "Buffer Behavior with Poisson Arrival and Geometric Output Processes", IEEE Trans, on Communications, pp1940-1941, Dec. 1974.

[10] Jin-Fu Chang and Rong-Feng Chang, "The Behavior of a Finite Queue with Batch Poisson Inputs Resulting from Message Packetization and a synchronous Server", IEEE Trans. on Communications, Vol. COM-32(12), PP1277-1285, 1984.

[11] Woei Lin. and P. R. Kumar, "Optimal Control of a Queueing System with Two Heterogeneous Servers", IEEE Trans. on Automatic Control, Vol. AC-29(8), pp696-703, 1984.

[12] Hewlett Packard, "System Administration Tasks", Hewlett Packard Company, 1991.

[13] SunSoft, "System Administration Guide Volumn 2", 1996.

● 저자소개 ●



하기중

1981년 광운대학교 응용전자공학과 졸업(공학사)
 1986년 중앙대학교 대학원 전자공학과 졸업(공학석사)
 1997년 청주대학교 대학원 전자공학과 졸업(공학박사)
 1986년~1994년 한국통신 연구개발본부 전임연구원
 1994년~현재 영동전문대학 정보통신과 조교수
 관심분야 실시간 정보처리, 컴퓨터 네트워크, 이동통신망



이천희

1968년 한양대학교 전자공학과 졸업·동대학원 졸업
 1975년 성균관대학교 대학원 전자자료처리학과 졸업
 1986년 성균관대학교 전자공학과 공학박사학위취득
 1971년 한국마벨(주) 근무
 1977년 동양공업 전문대학 전자공학과 근무
 1979년~현재 청주대학교 전자공학과 교수
 1983년~1985년 캘리포니아 산호세주립대 객원교수
 관심분야 VLSI Layout, ASIC, DRAM, CAD Tool개발 등