

시뮬레이션을 이용한 (N+1) 지능망에서의 시스템 확장방안 분석

노용덕, 송상훈

세종대학교 컴퓨터공학과 교수

요 약

지능망은 소비자 또는 가입자의 다양한 욕구를 신속하고도 효과적으로 제공하며 동시에 새로운 서비스에 대한 수요가 있을 경우에 이에 쉽게 대처할 수 있는 차세대 통신망이다. 가입자의 수가 증가하고 서비스의 질적요구가 높아짐에 따라서 시스템을 확장하여야 하는데, 여기에는 시스템의 특성에 따라서 여러 가지 대안이 존재한다. 이 논문에서는 두가지 가능성 있는 대안에 대하여 시뮬레이션 기법을 통하여 분석하고 이에 대한 선택 기준을 제시한다.

Analysis For (N+1) Type IN System Upgrade Plan

Noh Yongdeok, Song Sanghoon

Sejong University, Dept. of Computer Science

Abstract

The main idea behind the Intelligent Networks (IN) concept is the separation of switching functionality from the service control, in order to meet various service requirements of subscribers and development of new services in time. On the increase of subscriber and for high quality of service, the IN upgrades its service systems, and there are a number of ways depending on the system architecture and the characteristics of services. This paper compares two alternatives, traces the performance of them by means of the simulation technique, and finally puts a rule to select one of them.

1. 서론.

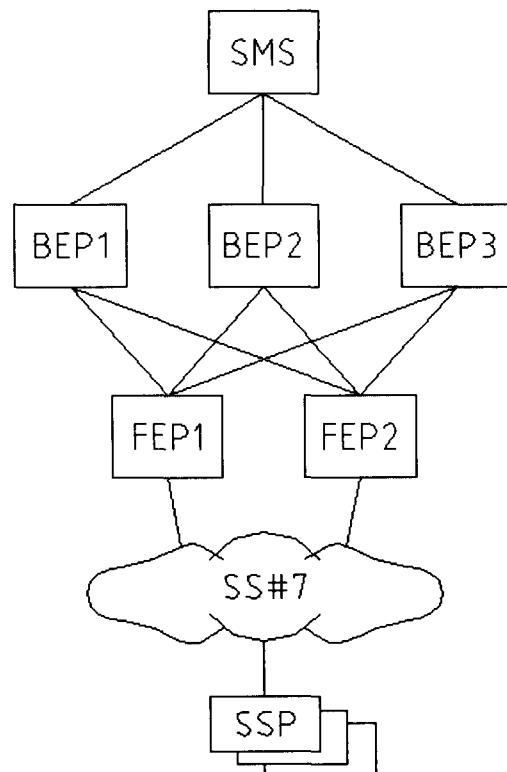
지능망은 소비자 또는 가입자의 다양한 욕구를 신속하고도 효과적으로 제공하며 동시에 새로운 서비스에 대한 수요가 있을 경우에 이에 쉽게 대처할 수 있는 통신망이다. 지능망은 SSP (Service Switching Point), SCP (Service Control Point), SMS (Service Management System), SCE (Service Creation Environment) 및 IP (Intelligent Peripheral)로 구성되어 있다[1].

본 논문에서의 지능망 서비스 시스템은 $(N+1)$ 형식으로 설계된 시스템으로, 서비스 제어장치인 SCP (Service Control Point) 가 SCP-FEP (SCP - Front End Processor) 와 SCP-BEP (SCP - Back End Processor)로 나뉘어져 있다. 여기서 SCP-FEP 는 SS#7 프로토콜 스택 등 신호망과의 인터페이스를 담당하는 기능과 지능망 응용 프로토콜 (INAP : Intelligent Network Application Protocol) 처리기능을 포함하는 지능망 인터페이스 처리부로 구성되며, SCP-BEP 는 지능망 호(Call)에 대한 서비스 제공을 위하여 서비스 처리기능을 수행하는 서비스 처리부로 이루어진다. SCP의 기능을 위와 같이 분리함으로써 인터페이스 처리기능과 서비스 처리기능을 독립적으로 운용할 수가 있으며, 따라서 서비스 처리 용량의 증가나 신규 서비스의 확대 실시에 따른 시스템의 확장을 각 기능별로 할 수가 있다[2]. 또한, 각각의 SCP-FEP 및 각각의 SCP-BEP 는 서로 독립적으로 동작하며, SCP-BEP의 경우에 동일한 종류 또는 다른 종류의 서비스를 하나의 SCP-BEP 별로 제공한다. 지능망에서는 다양한 서비스의 제공으로 트래픽에서의 오버로드가 생길 수가 있으며, 따라서 수요자를 만족시키는 QoS를 제공하는 것은 매우 중요한 일이다. 이를 위하여 SCP-FEP 와 SCP-BEP 간의 부하균형 (Load Balancing)이나 Call Gapping과 같은 방법을 사용한다[3]. 그러나 가입자의 수가 계속 늘어나고 이에 따라서 호의 수가 증가하는 경우에는 궁극적으로 지능망 시스템의 확장에 대하여 고려를 해야 한다. 여기서는 호의 수가 급격히 증가함에 따라서 시스템을 확장하는 경우에 고려할 수 있는 몇 가지 대안에 대하여 논하고 각 대안에 대하여 시뮬레이션 기법을 사용하

여 시험하고 그 결과를 정리하였다.

2. 대상 시스템과 시뮬레이션 모델의 검증

현재 사용중인 2개의 SCP-FEP 와 3개의 SCP-BEP, 그리고 1개의 SMP를 갖는 $(N+1)$ 형식의 지능망 서비스 시스템을 [그림 1]에 정리하였다. 시스템 설치시에 이 시스템이 갖는 성능, 즉 어느 정도 수의 호에 대한 서비스를 제공할 수 있는지에 대하여 시스템상에서 시험 호를 발생하고 그 결과를 수집하였다. 이때, SCP-BEP 3에서는 시험을 발생시키는 작업만을 수행하고 SCP-FEP는 1개만을 사용하였는데, 초당 40호, 50호, 및 60호의 경우에 대하여 자료를 수집하였다. 여기서는 이때 얻은 결과를 사용하여 이 시스템에 대한 시뮬레이션 모델을 검증하였다. 시뮬레이션 모델은 SLAM SYSTEM 4.1[4]을 사용하여 구현하였으며, 모델 구현시에 다음과 같은 사항을 가정하였다.



[그림 1] 지능망의 구조

- 각 호의 도착시간 및 처리시간은 지수분포를 따른다.
- FEP 내의 스택에서의 지연시간은 호가 들어오

고 나가는 시간 모두를 합쳐서 70ms 이다.

- 각 CPU에서의 기본부하는 FEP와 SMP(또는 SMS)는 20%이고, BEP의 경우는 25%이다.
- 각 프로세서내의 CPU 개수는 4개이며, 작업처리에 있어서 각 CPU는 균등한 기회를 갖는다.
- 하나의 호가 FEP에서 각 BEP로 갈 확률은 모두 같다. (표 2의 부하균형표 참조)

- 시스템내 프로세스 버퍼의 크기는 100 이다.

여기서는 그 당시에 얻은 SCP-FEP, SCP-BEP, SMS에 대한 CPU 사용량과 실행시간 (Turnaround Time = 처리시간 + 대기시간)을 조사하고, 각각의 얻은 값을 시뮬레이션 모델의 결과와 비교하였다[5]. [표 2] 및 [표 3]은 이때 사용한 SCP-FEP에서의 부하균형표와 모델 검증의 결과를 보여주고 있다. [표 2]에서 서비스내의 팔호안의 숫자는 도착한 하나의 호가 요구하는 서비스의 확률 값을 나타낸다.

[표. 2] 부하균형표

서비스	SCP-BEP1	SCP-BEP2
VPN (0.12)	0.5	0.5
AFS (0.10)	0.5	0.5
PN (0.35)	0.5	0.5
NP (0.43)	0.5	0.5

[표. 3] 모델의 동질성 검증

	SMS	BEP1	BEP2	FEP	Time
40 CPS	38	47	45	40	193
모델	37	46	46	39	205
Chi-Square 값 : 0.2737 < 11.14					

	SMS	BEP1	BEP2	FEP	Time
50 CPS	41	54	53	46	208
모델	40	50	52	44	226
Chi-Square 값 : 0.8440 < 11.14					

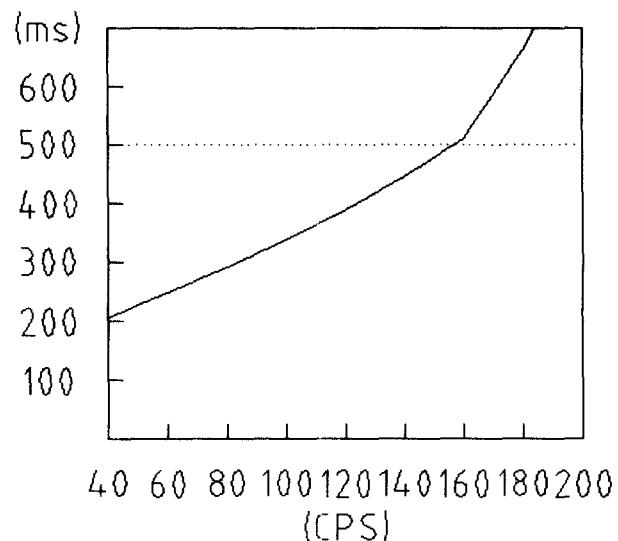
	SMS	BEP1	BEP2	FEP	Time
60 CPS	43	56	56	51	256
모델	44	55	55	49	249
Chi-Square 값 : 0.052 < 11.14					

[표 3]에서는 CPU의 Utilization과 실행시간에 대한 시뮬레이션 모델의 값과 실제 실험의 값을 유

의수준 0.05로 검정한 결과를 보이고 있다. [표 3]에서 보듯이, 40 CPS, 50 CPS와 60 CPS 모두, 시뮬레이션 모델의 값과 실험의 값이 서로 다르다고는 할 수 없다.

3. 호의 수와 시스템 자원의 사용량

시뮬레이션 모델을 사용하여 [그림 2]와 같은 지능망 시스템에 대한 성능추이를 분석하였다. 호의 수가 증가함에 따라서 각 호에 대한 실행시간 및 FEP, BEP, SMP CPU에 대한 사용량 (Utilization)을 조사하였으며, 그 결과를 [그림 3]에 정리하였다.

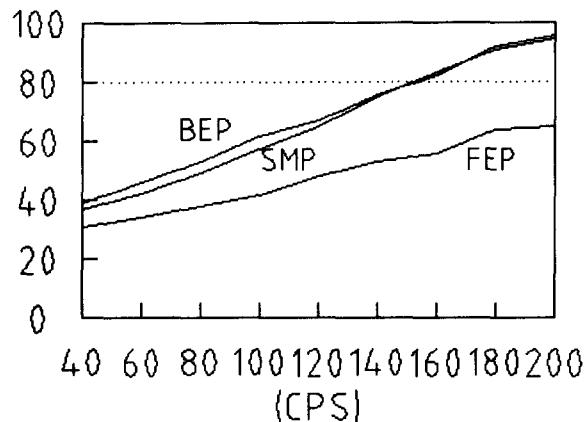


[그림 2] 호의 수와 실행시간 추이.

지능망에서는 지능망 시스템의 과부하에 대하여 여러 가지 기준을 정의하여 사용하고 있다. 여기서는 하나의 서비스 호가 SCP-FEP에 들어온 후에 다시 연결을 끊는 신호를 보낼 때까지의 시간, 즉, 실행시간이 500 ms를 넘거나 CPU의 사용량이 80%를 초과하기 시작하면 과부하가 걸린 것으로 판단하기로 한다.

[그림 2]에서 보면, 초당 160호 가까이에서 실행시간이 500ms를 초과하기 시작하고, CPU 사용량에 있어서는 BEP와 SMP 모두 초당 150호 근처에서 80%를 초과하기 시작하고 있다. 따라서, 이를 해소하기 위한 방법으로 새로운 부하균형값이나 Call Gapping 기법을 사용하기도 하나, 이러한 방법은 일시적인 해결책은 될 수 있어도, 호의 수가 항상

일정 범위를 넘어서는 경우에는 궁극적인 해결책이 뒷 못하므로 시스템 확장을 고려해야 한다.



[그림 3] FEP, BEP, SMP CPU 평균 사용량 추이.
여기서는 시스템의 확장 방안으로 다음의 2가지를 고려하고 각각의 방안에 대한 시뮬레이션을 실시하였다.

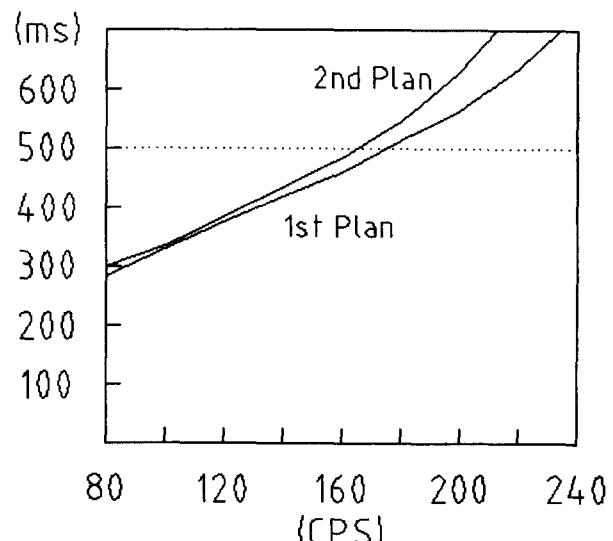
- 제1안 : SMP 와 SCP-BEP내의 CPU 수를 각각 4개 및 2개씩 모두 추가.
- 제2안 : SMP 내의 CPU 수를 4개 추가하고 SCP-BEP를 1개 증설

4. 시스템 확장안 검토 및 비교

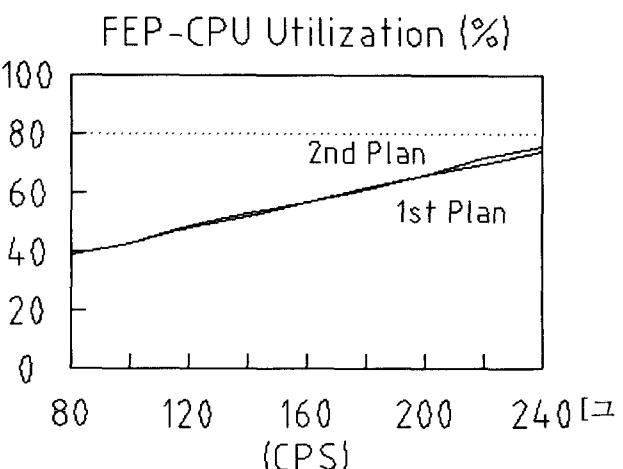
현재 SMP 내에는 4개의 CPU 가 있다. 이 수를 증가시킴으로써 보다 많은 호를 처리할 수 있을 것으로 기대된다. 여기서 CPU 수를 늘림에 따라서 SMP 자체에 보다 많은 오버로드가 생길 것이므로 여기서는 오버로드가 30%로 증가할 것으로 가정하였다.

제1안에서는 BEP를 새로이 추가하기 보다는 현재의 BEP 내에 CPU를 2개씩 추가로 설치한다. 반면에, 제2안에서는 현재 3개의 BEP에 1개의 BEP를 더 증설하여 4개의 BEP로 구성된 지능망을 고려한다. 제2안의 경우는 BEP 1개의 증설로 CPU 가 4개 늘어나는 셈이고, 제1안의 경우는 BEP마다 CPU를 2개씩 늘림으로써 모두 6개의 CPU 가 추가되는 셈이다. 그러나 제1안의 경우에 BEP내의 CPU 증가로 인하여 오버로드가 늘어나서 30%가 되는 것으로 가정하였다. 이러한 가정하에 80 CPS부터 시작하여 240 CPS까지 시뮬레이션을 실시하고 각 안에 대한 실행시간, FEP-CPU, BEP-CPU,

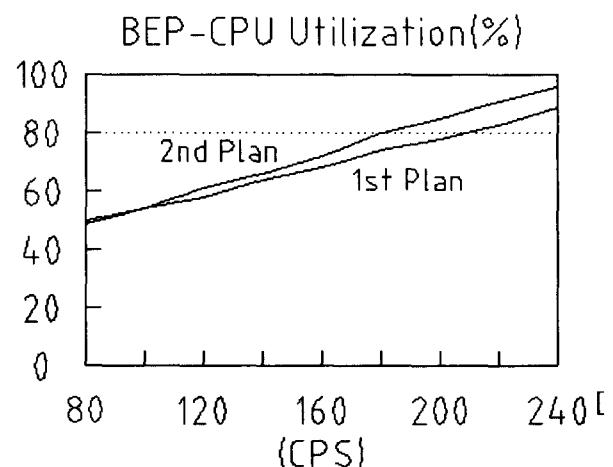
SMP-CPU 의 평균 사용량을 각 CPS별로 [그림 4]부터 [그림 7]까지 정리하였다.



[그림 4] 각 안에 대한 실행시간 비교.



[그림 5] 각 안에 대한 FEP-CPU 사용량 비교.



[그림 6] 각 안에 대한 BEP-CPU 사용량 비교.

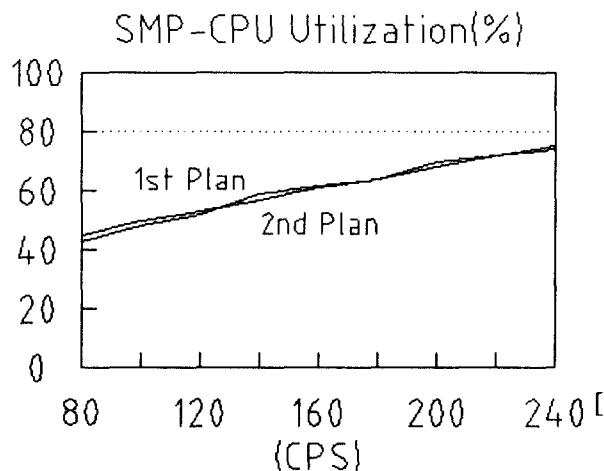


그림 7] 각 안에 대한 SMP-CPU 사용량 비교.

두 안에 대한 실행시간을 보면, 시스템을 확장하더라도 실행시간에 대한 단축은 그다지 크지 않고 있다. 제1안의 경우는 180 CPS부터, 그리고 제2안의 경우에는 160 CPS부터 버퍼에서의 평균대기시간이 늘어나기 시작하고 있어서 시스템의 버퍼 크기를 늘려서 실행시간을 약간 단축시킬 수도 있을 것으로 판단된다.

[그림 5]는 FEP-CPU의 평균사용량을 보이고 있다. FEP-CPU의 평균사용량은 시스템 확장전이나 확장후나 크게 차이가 없고, 확장안끼리도 거의 비슷한 값을 갖는다.

그러나 [그림 6]에서 보듯이 BEP-CPU의 경우는 확연히 차이를 보인다. 당연히 확장 전에 비하여 사용

량이 크게 감소하였으며, 제2안에 비하여 제1안의 BEP-CPU 사용량이 적은 값을 보이고 있다. 제1안의 경우에는 확장전에 비하여 초당 50호 정도가 늘어나서 80% 정도의 사용량을 보이고 제2안의 경우에는 20호 정도 더 처리해야 80%정도의 CPU 사용량을 보이고 있다. SMP-CPU의 경우는 [그림 7]에서 보듯이 확장전에 비하여 평균사용량이 많이 감소하였으나 제1안이나 제2안의 평균사용량은 거의 동일하게 움직이고 있다.

확장안에 대하여 비교하자면, [그림 4]의 실행시간과 [그림 6]의 BEP-CPU 평균사용량으로 보아 제1안이 제2안보다 훨씬 효과적이다. 이는, 제1안이 보다 적은 실행시간과 보다 적은 CPU 사용량으로 보다 많은 가입자의 호출 처리할 수 있는 여유를 갖기 때문인데, 이와같이 제1안이 제2안에 비하여

훨씬 효과적인 확장안으로 결과가 나오는 것은 각 BEP마다 모두 가입자에 대한 데이터베이스를 보유하고, 이를 유지해야 하는 작업때문으로 보인다. 결론적으로 두 확장방안에 대하여 시뮬레이션한 결과에 따르면, BEP의 수를 단순히 증가시키는 방안보다는 현재의 시스템 성능을 향상시키는 확장방안이 훨씬 더 효과적이다.

5. 요약

여기서는 $(N+1)$ 형태의 지능망 서비스 시스템 하에서의 가입자의 수요가 늘어나서 궁극적으로 시스템을 확장하게 되는 경우에 대한 두가지 방안에 대하여 검토하고 시뮬레이션의 결과를 분석하였다. 이러한 작업들은 시뮬레이션의 모델링에 있어서 실제 시스템 동작에 어느 정도 자세히 근접하게 설계했는지에 따라서 결과에 커다란 영향을 미칠 수가 있다. 이 모델에서는 실제 데이터 수집의 어려움으로 시스템내의 프로세스 하나하나에 대하여는 모델링을 하지 않았으므로, 이에 대하여 보다 꼭 넓은 자료 수집과 모델링 작업을 수행하여야 할 것으로 판단된다.

참고문헌

- [1] Thorner, "Intelligent Networks", Artech House, 1994
- [2] 김재혁, 김종주, 서인수, "지능망 서비스 시스템에서의 Load Balancing 기능 적용", 제4회 지능망 학술대회 논문집, 1999
- [3] 김종주, 송창환, 김재혁, 강용구, "지능망 서비스 시스템의 과부하 제어 고찰", AIN'97, 1997
- [4] Pritsker, "SLAMSYSTEM", Pritsker Corporation, 1990
- [5] 노용덕, 김종용, 송상훈, 심장섭, "(N+1) 형식의 지능망 서비스 시스템에서의 부하균형분석", 한국 시뮬레이션학회논문집, 11-20쪽, Vol 1, No 1, 2000년 3월