

(N+1) 형식의 지능망 서비스 시스템에서의 부하균형 분석

Load Balancing Analysis For (N+1) Type Intelligent Network Service Systems

노용덕*, 김종용*, 송상훈*, 심장섭**

Noh Yongdeok, Kim Jongyong, Song Sanghoon, Shim Jangsup

Abstract

The main idea behind the Intelligent Networks (IN) concept is the separation of switching function from the service control, in order to meet various service requirements of subscribers and development of new services in time. The (N+1) architecture of the IN service system consists of front-end processors (FEP) and back-end processors (BEP), which is in service in a domestic telecommunication company, and balances the traffic load for services among the processors. Under this architecture, the IN service system can distribute the traffic load dynamically, and IN services are easily launched and manipulated minimizing impacts on the existing services. This paper traces the performance of the (N+1) structured system by means of the simulation approach, and shed a light on the result.

* 세종대학교 컴퓨터공학과 교수

** 데이콤 종합연구소

1. 서론

지능망은 소비자 또는 가입자의 다양한 욕구에 대하여 신속하고도 효과적으로 대응하는 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]. 여기서 SSP 와 SCP 는 Call Setup 프로토콜에 따르는 신호전달용 패킷망인 SS#7 네트워크로 연결되어 있고, SCP 와 SMS는 데이터 통신 네트워크로 연결된다. SMS는 간혹 SMP (Service Management Point)라고도 한다. 사용자는 SSP를 통하여 지능망에 연결되며, SSP 는 사용자의 지능망 서비스 요청 여부를 감지하고 SCP 에 지능망 서비스를 요청하며, SCP 로부터 받은 정보에 따라서 필요한 네트워크 관련 작업을 수행한다. SCP 에는 데이터베이스 및 SLP (Service Logic Program) 이 있다. SLP는 서비스를 위한 제어로직이 내장된 프로그램이며, 데이터베이스에는 서비스 제공에 필요한 정보가 저장되어 있다. SMS 는 지능망 관리를 위한 시스템으로 SCP 에 지능망 서비스에 필요한 데이터베이스를 다운로드하거나 서비스 가입자와 제공자간의 인터페이스를 제공하고, 여러 가지 측정된 데이터를 수집한다. SCE 는 SLP 개발에 필요한 여러 가지 툴들을 포함하고 있다. IP 는 서비스 이용자와 지능망간의 다양한 정보 교환을 가능하게 하는 것으로, 음성 데이터 수신, 음성의 송출, 안내방송, 등의 다양한 기능을 갖는다.

본 논문에서의 지능망 서비스 시스템은 (N+1) 형식으로 설계된 시스템이다. 여기서 (N+1) 지능망 시스템이란, 기본적으로 지능망 시스템이 하나의 SMS 와 SCP 로 구성되는데 반하여 시스템의 이상으로 서비스 제공이 불가능하게 되는 경우에 대비하기 위하여 추가로 1개의 시스템을 병행으로 설치, 운영하는 시스템을 말한다. 즉, N 이 하나의 지능망 서비스 시스템이

면, (N+1)은 여기에 하나의 시스템을 추가로 설치하였음을 의미한다.

여기서 다루는 지능망 시스템에서는 서비스 제어장치인 SCP (Service Control Point) 가 SCP-FEP (SCP - Front End Processor) 와 SCP-BEP (SCP - Back End Processor) 의 기능적으로 분리된 구조를 갖고 있다. 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 처리 기능을 수행하는 서비스 처리부로 이루어진다. SCP 의 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-BEP 의 경우에 동일한 종류 또는 다른 종류의 서비스를 하나의 SCP-BEP 별로 제공한다.

지능망에 대한 연구는 지능망에 대한 기본적인 개념이 정리된 이후에 꾸준히 계속되어 왔으며 시스템 구성상의 방식에 따라서 관련 연구가 진행되고 있다. [7]에서는 지능망 시스템에서의 SSP 의 망에서의 위치에 대한 수학적 모델을 만들고 가장 적은 비용으로 SSP를 구현하는 방안을 제시하였다. 일반적인 지능망에서 가입자의 서비스 요청이 많아서 오버로드가 생기는 경우에 대한 대처방안이 [8]에 제시되어 있는데, 여기서는 오버로드 해소방안의 하나인 Call Gapping 방법에 새로운 알고리즘을 추가로 적용하고 이에 대한 시뮬레이션 결과를 보이고 있다. 오버로드 해소를 위하여 SS#7 상에서의 트래픽 조절을 위한 분석 작업은 [9]에서 수행되었다. (N+1) 시스템에 관련된 지능망 구조에 대한 연구는 시험호 발생방법에 대한 것[6]이나, 부하균형표의 일반적인 사용방식과 이에 대한 장점[2], 그리고 지능망

시스템에서의 과부하를 판단하는 기준과 이를 제어하기 위한 방안에 대한 연구[6]가 있다. 이러한 연구들은 하나의 안으로서 부하균형표의 사용이나 과부하의 판단 대상들을 제시한 것으로 구체적으로 확실한 해결방안을 제시하는 것은 아니다.

지능망에서는 다양한 서비스의 제공으로 트래픽에서의 오버로드가 생길 수가 있기 때문에 수요자를 만족시키는 QoS를 제공하는 것은 매우 중요한 일이다. 따라서 본 논문에서는 SCP-FEP와 SCP-BEP 간의 부하균형 (Load Balancing)에 대하여 논하고, 여러 가지 부하균형에 대한 경우를 시뮬레이션 기법을 사용하여 분석하고 그 결과를 정리하였다.

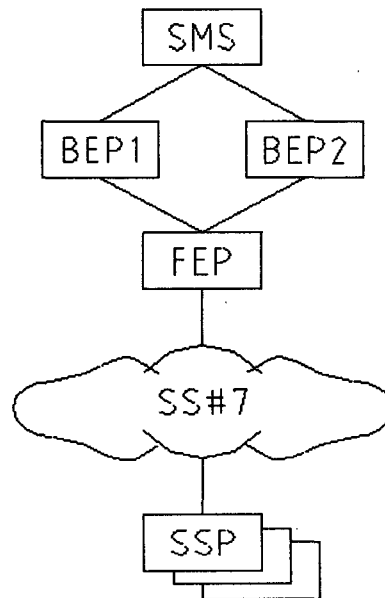
2. SCP-FEP와 SCP-BEP 간의 부하균형

1개의 SCP-FEP 와 2개의 SCP-BEP를 갖는 (N+1) 형식의 지능망 서비스 시스템을 <그림 1>에 정리하였다. SSP 에 전달된 지능망 호는 SS#7 신호망을 거쳐 SCP-FEP 로 전달된다. SSP 에서는 PC(Point Code)를 사용하여 SCP-FEP로 신호를 보낸다. SCP-FEP의 SS#7 Stack을 거쳐 INAP 처리 블록으로 전달된 호는 서비스 구분자 (예: InitialDP의 서비스키)로부터 처리가 요구된 서비스가 결정되고 INAP 처리 블록의 부하균형표에 있는 값에 따라 적절한 SCP-BEP로 전달된다.

<표 1>은 INAP 처리 블록에서 관리하는 서비스별 부하균형표를 나타낸다. 그리고, SCP-BEP에서는 전달된 지능망 호에 대해 요구되는 서비스 로직을 적용하여 서비스를 제공하게 된다.

이러한 부하균형표는 각 SCP-FEP의 INAP 처리 블록마다 존재하며, INAP 처리 블록에서 각 신규 지능망 호가 SCP-FEP에 들어 올 때에, INAP 처리 블록에서는 그 호의 서비스 구분자로부터 서비스를 결정하고 부하균형표로부터 이 호가 처리될 SCP-BEP를 결정하는 것이다. 부하균형표에서 p(i,j)의 값이 0인 것은 서비스(i)가

SCP-BEP(j)로 전달되지 않을 것임을 나타낸다. 또한 모두 0의 값을 갖는 서비스의 경우는 서비스가 이 지능망 서비스 시스템에서 현재 제공하지 않음을 나타내는 것으로, INAP 처리 블록에서는 이러한 호에 대해서는 비정상호 처리 절차를 밟는다. 또한 부하균형표에 정의되지 않은 서비스 구분자를 갖는 신규 호에 대해서도 비정상호 처리절차를 밟는다.



<그림 1> 지능망의 구조

<표. 1> 서비스별 부하균형표.

	BEP(1)	BEP(2)	...	BEP(n)
Service(1)	p(1,2)	p(1,2)	...	p(1,n)
Service(2)	p(2,1)	p(2,2)
...
Service(m)	p(m,1)	p(m,2)	...	p(m,n)

SMS(또는 SMP라고도 함)는 각 SCP-FEP 및 SCP-BEP로부터 정의된 과부하 점검요소들을 주기적으로 점검하고 점검결과에 따라 적절한 성능 제어의 방법 및 수준을 결정한다[2]. 성능제어 방법으로는 SSP로부터 전달되는 호의 수를 제한하는 INAP 오퍼레이션 CallGap을 이용하는 방법 및 SS#7 Stack과의 연결 점 수를 이용한 부하균형 방법이 있다. CallGap의 경우는 지능망 서비스 시스템 전체의 트래픽 과다 등과 같은 시스템 전체 성능 제어시에 사용되며, 부하균형의 경우는 어느 특정 SCP-BEP에 트래픽의 집중되는 경우에 호를 적절히 다른 SCP-BEP로 재분배 하기 위하여 사용한다. 그러나 여기서는 INAP 처리 블록이 SCP-FEP에 위치하므로 INAP 처리 블록은 지능망 호를 부하균형표에 따라 적절한 SCP-BEP로 전달하게 된다.

이 INAP 처리 블록의 부하균형 값은 SMS의 성능 관리 기능에 의해 자동적으로도 변경이 가능하며, 또한 허가된 서비스 운용자에 의한 수동 변경도 가능하다. 이러한 수동 제어에 의한 부하균형 값의 변경은 자동 제어에 의한 변경에 우선하는 것이 일반적이다[3].

이와 같은 지능망 시스템은 여러 종류의 서비스를 제공하며, SCP-FEP와 SCP-BEP 간의 부하균형 값을 할당하는 여러 가지 경우가 발생하므로, 이를 분석하기 위한 가장 좋은 방법은 시뮬레이션에 의한 것이다. 또한, 시뮬레이션 기법은 여러 가지의 시스템 구조와 그에 관계된 인자들에 대하여 조사할 수 있는 장점이 있다. 아울러, 지능망 시스템의 확장이나 새로운 서비스의 제공과 같이 실제 시스템의 구현이나 서비스의 제공에 앞서 이에 대한 타당성 검토나 가능성, 등을 검증하기에 효과적이다.

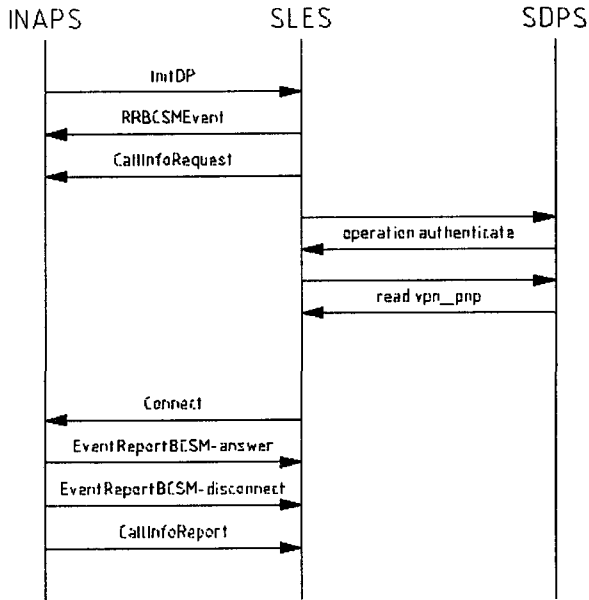
3. 시뮬레이션 모델 및 검증

부하균형분석 작업을 위하여 윈도우 3.1하에 SLAM SYSTEM 4.1[4]을 사용하여 시뮬레이션을 실시하였다. 시뮬레이션 모델 구현시에 다음과 같은 사항을 가정하였다.

- ① 각 호의 도착시간 및 처리시간은 지수분포를 따른다.
- ② FEP 내의 스택에서의 지연시간은 호가 들어 오고 나가는 시간 모두를 합쳐서 70ms 이다
- ③ 각 CPU에서의 기본부하는 FEP와 SMS는 20%이고, BEP의 경우는 25%이다.
- ④ 각 프로세서내의 CPU 개수는 4개이며, 작업 처리에 있어서 각 CPU는 균등한 기회를 갖는다.
- ⑤ VPN, AFS, PN, NP의 4개 서비스가 제공된다.

위와 같은 가정하에서 다음과 같은 과정에 따라서 시뮬레이션이 수행되도록 모델링하였다.

- ① 각 시스템별로 오버헤드를 할당한다.
- ② 호를 발생시키되, 도착간격시간은 지수분포를 가정한다. 예를 들어서, EXPON(25)는 40 CPS의 경우이며, 단위는 ms이다.
- ③ 원하는 서비스를 사전 비율에 따라서 랜덤으로 지정한다.
- ④ 부하균형표를 사용하여 찾아갈 BEP를 미리 정한다.
- ⑤ FEP에 도착하여 스택에 대기한다. 이는, 각 서비스별 작업처리에 있어서 FEP내의 INAP 초기 작업 직전을 의미한다.
- ⑥ CPU의 할당을 기다린 후에, InitDP가 BEP내의 SLES로 전달된다[<그림 2> 참조]. SLES는 CPU 할당을 받은 후에 지정된 시간 동안 이를 처리한 후에 처리 결과를 원래 보내온 FEP내의 INAP로 보내고, 동시에 다음에 도착한 작업을 처리한다.
- ⑦ 이러한 과정을 각 서비스별로 서비스 작업내용에 따라서 수행한다. <그림 2>는 4개의 서비스중에서 VPN의 경우에 FEP내의 INAPS와 BEP내의 SLES 및 SDPS라고 불리는 프로세스 사이에서 이루어지는 작업의 순서를 보이고 있다. 이 모델에서는 각 서비스마다 이루어지는 절차에 따라서 시뮬레이션이 행하여지도록 설계하였다.



<그림 2> VPN의 작업처리 순서

- ⑧ 하나의 도착한 호에 대한 FEP 및 BEP 내에서의 작업이 끝나고 마지막으로 Connect 작업이 이루어지면, 나머지 작업에 관계없이 현재시간에서 최초 도착시간을 뺀 결과를 사용하여 Delay Time을 계산한다.
- ⑨ 다시 각 서비스별로 서비스 작업내용에 따라서 수행하다가, NP를 제외한 작업에 대하여 Call Ticket 작업을 처리한 후에 호의 처리를 종결한다.
- ⑩ 위 작업을 시뮬레이션 시간으로 60,000 ms 동안 수행하며, 다음 항목에 대하여 결과를 수집한다.

- Delay Time
- FEP1 CPU Utilization
- BEP1, BEP2 CPU Utilization
- SMP Utilization

모델의 검증을 위하여 실제의 상업용 지능망에서 시험호[6]를 발생시킨 후에 얻은 결과와 시뮬레이션 모델링을 수행한 후에 얻은 결과를 비

교하였다. 상업용 지능망에서는 시험호를 발생한 후에 이에 대한 처리는 SCP-FEP1, SCP-BEP1, SCP-BEP2, 및 SMS를 사용하였으며, 40 CPS (Calls Per Second) 와 60 CPS 의 경우에 대하여 실시하였다. 여기서는 40 CPS와 60 CPS 시험호를 실제 지능망에서 처리한 당시에 얻은 SCP-FEP, SCP-BEP, SMS 에 대한 CPU 사용량과 실행시간(Turnaround Time = 처리시간 + 대기시간)을 조사하고, 각각의 얻은 값을 시뮬레이션 모델의 결과와 비교하였다. <표 2> 및 <표 3>은 이때 사용한 SCP-FEP에서의 부하균형표와 모델 검증의 결과를 보여주고 있다.

<표. 2> 부하균형표 : 경우1.

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 2>에서 서비스내의 팔호안의 숫자는 도착한 하나의 호가 요구하는 서비스의 확률 값을 나타내며, 각 서비스의 의미는 아래와 같다[5].

- ① VPN (Virtual Private Network) 가상사설망 - 가입자가 지능망을 이용하여 자사의 통신망을 자유롭게 연결하여 자체 단축번호체계를 구성하고, 다양한 부가서비스도 이용하는 등, 마치 공중통신망을 사설통신망처럼 사용하게 하는 서비스.
- ② AFS (Advanced Freephone Service) 고도착신과금 - 발신자대신 서비스 가입자가 전화요금을 지불하는 서비스.
- ③ PN (Personal Numbering) 개인번호 - 고정망에서 망에 접속된 단말에 부여하는 단말주소와 가입자가 밀접하게 결합되어 있는 것과는 달리 가입자 개인에게 번호가 부여되는 서비스. 가입자에게 단말 이동성을 제공하고, 가입자마다 고유의 서비스 프로파일을 유지하여

가입자별 서비스 요구사항을 최대한 지원한다.

- ④ NP (Number Portability) 번호이동 - 회선교환망에서 가입자가 통신사업자, 가입지역, 이용 서비스 등을 변경할 때 통화품질, 서비스 신뢰성, 서비스 편의성의 저하 없이 자신의 전화번호를 유지할 수 있는 서비스.

<표 3>에서는 CPU의 Utilization과 실행시간에 대한 시뮬레이션 모델의 값과 실제 실험의 값을 유의수준 0.05로 검정한 결과를 보이고 있다. <표 3>에서 보듯이, 40 CPS와 60 CPS 모두, 시뮬레이션 모델의 값과 실험의 값이 서로 다르다고는 할 수 없다

<표 3> 모델의 동질성 검증

	SMS	BEP1	BEP2	FEP	Time
40 CPS	38	47	45	40	193
모델	37	46	46	39	204
Chi-Square 값 : 0.2424 < 11.14					

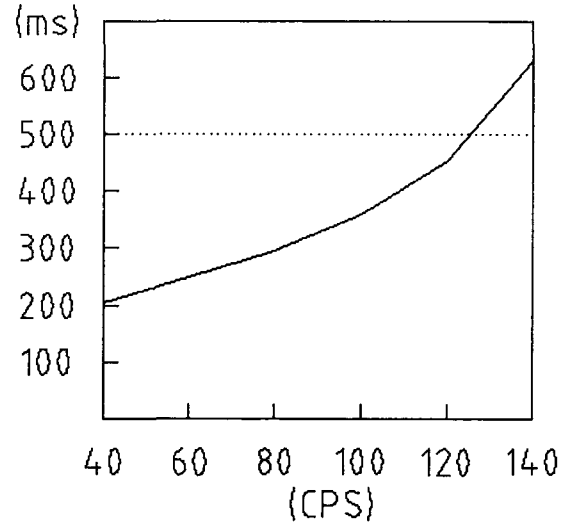
	SMS	BEP1	BEP2	FEP	Time
60 CPS	43	56	56	51	256
모델	44	55	55	49	248
Chi-Square 값 : 0.064 < 11.14					

4. 모델의 분석

위의 모델을 사용하여 1개의 SCP-FEP와 2개의 SCP-BEP, 및 1개의 SMS를 사용하는 환경하에서 호의 수요가 증가함에 따라 반응하는 시스템의 성능에 대하여 살펴보고, 이에 대응하여 여러 가지 부하균형의 값을 적용하였다. 지능망에서는 지능망 시스템의 과부하에 대하여 여러 가지 기준을 정의하여 사용하고 있는데, 여기서는 하나의 서비스 호가 SCP-FEP에 들어온 후에 다시 연결을 끊는 신호를 보낼 때까지의 시간, 즉, 실행시간이 500ms를 넘지 않게 하는 것을 목표로 삼는다.

먼저, <표 2>의 경우1에 대하여 40 CPS, 60

CPS, 80 CPS, 100 CPS, 120 CPS, 및 140 CPS로 호가 증가할 때에 실행시간의 추이를 조사하고 그 결과를 <그림 3>에 정리하였다.



<그림 3> 각 호의 실행시간 추이

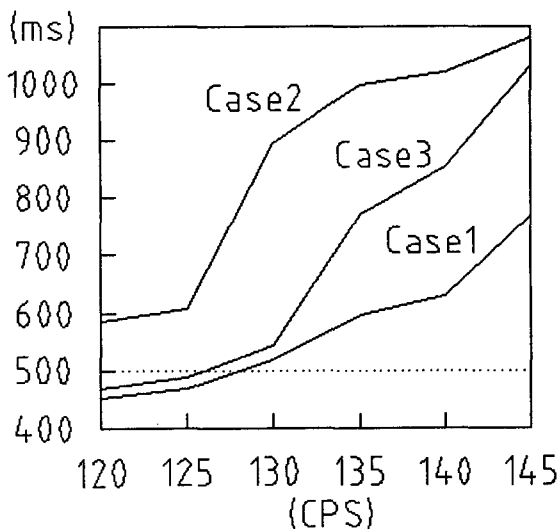
<그림 3>에서 보듯이, 120 CPS와 140 CPS 사이에서 각 호당 평균실행시간이 500ms를 초과하고 있다. 초당 120호의 처리는 많은 호를 처리하는 경우이기기는 하지만 부하균형값에 변화를 줌으로써 호의 처리 수가 늘어나는 지에 대하여 조사하였다. 대개 이러한 경우는 사용량이 적은 CPU에 높은 부하를 할당하는 방식으로 처리하기도 하지만, 경우1에서는 들어오는 호를 SCP-BEP1과 SCP-BEP2에 균등히 배분하므로 <표 3>에서 보는 바와 같이 BEP1과 BEP2의 CPU 사용량이 서로 근사하다. 따라서 CPU에 작업할 당량을 새로이 부과하는 방안보다는 <표 4>와 같은 새로운 부하균형표를 적용하여 시뮬레이션을 실시하였다.

<표 4>에서 경우2와 경우3은 <표 2>의 경우1과 비교하여 볼 때에 서로 극단적인 부하균형값을 갖고 있고 경우1은 중간값을 갖고 있는 셈이다. 또한, 경우2나 경우3에서 BEP1과 BEP2에 할당하는 값을 서로 바꾸어도 들어오는 호에 대하여 균등히 배분하므로 결과는 마찬가지이다.

이러한 부하균형값에 대하여 120 CPS에서 145 CPS까지 5 CPS 단위로 5분간 시뮬레이션한 결과를 <그림 4>에 정리하였다.

<표 4> 부하균형표 : 경우2, 경우3.

	경우2		경우3	
	BEP1	BEP2	BEP1	BEP2
VPN (0.12)	0.0	1.0	0.25	0.75
AFS (0.10)	1.0	0.0	0.75	0.25
PN (0.35)	0.0	1.0	0.25	0.75
NP (0.43)	1.0	0.0	0.75	0.25



<그림 4> 각 경우에 대한 실행시간 추이.

<그림 4>에서 보듯이, 경우1의 부하균형이 여전히 경우2나 경우3보다 실행시간이 적게 걸리는 것으로 보인다. 경우2는 가장 극단적인 부하균형값으로 가장 호의 실행시간이 길게 나오고 있다. 경우3에서는 그 결과가 경우1과 경우2의 중간에 위치하는데, 호의 수가 증가함에 따라서 경우1보다 경우2에 근접하고 있다. 또한, 125 CPS에서 135 CPS 사이에서는 경우마다 실행시간이 급격히 늘어나며 이 정도수의 호에서 시스템 부하가 걸리기 시작하는 것으로 볼 수 있다. 결론적으로는, 현재 요청하는 서비스의 분포

와 지능망 서비스 시스템하에서는 경우1의 부하균형표가 다른 어떤 부하균형 값보다도 더 합리적이라고 결론을 내릴 수 있다.

5. 요약

여기서는 (N+1) 형태의 지능망 서비스 시스템하에서의 부하균형 방식과 부하균형표의 값에 따라서 실행시간이 어떻게 변하는 지에 대하여 살펴 보았다. 그러나, 앞에서 지적하였듯이, 지능망 시스템의 부하를 측정하는 방식에는 실행시간 이외에도 CPU 시간의 사용률이나, 스택에서의 대기시간, 대기열의 크기, 하나의 SSP 에 대한 동시 트랜잭션의 수, 등이 판단기준이 된다. 그러나 여기서는 단순히 실행시간만을 대상으로 고려하여 각 서비스의 절차에 따라 모델링한 후에 시뮬레이션을 실시하였다. 부하균형표를 작성하는 경우에도 매우 많은 경우의 수가 존재하므로 각 서비스별로 소요되는 CPU 시간에 따라서 부하균형값을 계산하고 그에 따른 시뮬레이션을 시도할 필요가 있다. 또한 아직 제공하지 않은 서비스가 실제로 사용될 경우에 대비하여 시스템에게 미치는 영향을 사전에 실험하는 것도 중요한 일이 될 것이다.

그러나 이러한 작업들은 시뮬레이션 모델링이 실제 시스템 동작을 어느 정도 자세히 표현할 수 있는가에 따라서 결과에 커다란 영향을 미칠 수가 있다. 이 모델에서는 실제 데이터 수집의 어려움으로 시스템내의 프로세스 하나하나에 대하여 모델링을 하지 않았으므로, 이에 대하여 보다 폭 넓은 자료 수집 후에 모델링 작업을 수행하여야 할 것으로 판단된다. 또한, FEP 와 BEP 프로세서 수의 변화에 따른 시스템의 서비스 수용능력에 대하여도 조사하여 볼 필요가 있다고 생각된다.

참고문헌

- [1] Thorner, "Intelligent Networks", Artech House, 1994
- [2] 김재혁, 김종주, 서인수, "지능망 서비스 시스템에서의 Load Balancing 기능 적용", 제4회 지능망 학술대회 논문집, 83-86쪽, 1999
- [3] 김종주, 송창환, 김재혁, 강용구, "지능망 서비스 시스템의 과부하 제어 고찰", 133-136쪽, AIN'97, 1997
- [4] Pritsker, "SLAMSYSTEM", Pritsker Corporation, 1990
- [5] 김종용 외, "지능망 서비스 및 시스템의 최적화 방안에 대한 연구", 11-12쪽, 데이콤보고서, 1999
- [6] 한경흠, 강용구, "지능망 서비스 시스템의 시험 방안에 대한 고찰", 164-168쪽, 제4회 차세대 지능망 학술대회
- [7] Samson, L. and J. Lansard, "Optimal SSP Location in an Intelligent Network", p335-339, ISS'95, April 95, Vol 2
- [8] Kawahara R. and Takuya Asaka, "Overload Control for Intelligent Networks Based on an Estimation of Maximum Number of Calls in a Node", IEEE Intelligent Network Workshop IN'97
- [9] McMillian D. and M. Rumsewicz, "Analysis of Congestion Control for SCCP Traffic & the Impact on Intelligent Network Services", IEEE Intelligent Network Workshop IN'96

● 저자소개 ●



노용덕
 서울대학교 산업공학과 학사
 Auburn Univ. 산업공학과 석사, 박사
 국방과학연구소 선임연구원
 현 세종대학교 전자정보공학대학 컴퓨터공학과 교수
 관심분야 : 컴퓨터그래픽스, 시뮬레이션, 성능분석, 가상현실



김종용
 서울대 이학사
 스트라스클라이드대 정보공학 석사
 런던대 전산학 박사
 금성사 (런던법인)
 데이콤연구소 CTI기술연구팀장
 하나로통신 기술연구팀장
 현재 세종대학교 컴퓨터공학과 교수
 관심분야 : 지능망시스템, 정보검색, CTI, 전화부가시스템

송상훈
 연세대학교 전자공학과 학사
 과학원 전자공학과 석사
 미네소타대 전산학 박사
 한국통신 선임연구원
 현재 세종대학교 전자정보공학대학 컴퓨터공학과 교수
 관심분야 : 멀티미디어 통신, 정보 및 통신보안

심장섭
 한양대학교 전자공학 석사
 OTELCO 종합연구소 연구원
 데이콤 기본통신서비스 및 시스템 개발
 지능망 플랫폼 개발, 하나로 통신 지능망 시스템 개발
 현재 데이콤 종합연구소 IMT 2000 시스템 개발팀장
 관심분야 : 전전자교환기시스템, 지능망시스템 및 서비스, IMT2000
 시스템 및 서비스