

지능망 구조에서 공정성과 우선순위를 보장하는 과부하 제어 연구

正會員 이 용*, 송 주 석*

Overload Control of SCP in Intelligent Network with Fairness and Priority

Yong Lee*, Joo-Seok Song* *Regular Members*

요 약

본 논문에서는 지능망에서의 과부하 제어를 위해 SCP와 SSP간에 제어에 필요한 정보를 주고받으며 상호 협동하는 방법으로 고정방식과 도착발생방식의 두 가지 메커니즘을 제시하였다. 이 메커니즘들은 기본적으로 ACG를 따르는 간단한 알고리즘을 사용한다. 공정성을 유지하기 위하여 각 SSP별로 다른 호차단시간을 적용하며 우선순위를 보장하기 위하여 높은 우선 순위의 지능망호에 우선적으로 서비스를 제공하도록 하였다. 두 가지 방식은 모두 호 처리율에 있어서 비슷한 결과를 보인다. 모든 지능망호가 같은 확률로 서비스되는지를 비교하는 공정성도 모든 SSP에 대하여 같은 값으로 유지되고 우선순위 성취도는 도착발생방식이 고정방식에 비해 약간 우수함을 보여준다.

ABSTRACT

This paper proposed two mechanisms of fixed method and new arrival method. They are all interactive mechanisms in that a SCP and the SSPs exchange load information for overload control. These mechanisms using simple algorithm based on the standard ACG, apply different call gapping time on each SSP under the fairness scheme when they allocates SCP capacity to competing demands. Also in the mechanisms, priority scheme is implemented. Two mechanisms show the almost same results on the call throughput. Fairness about the probability of rejection is maintained same value on all SSPs. On Priority scheme, new arrival method has better achievement than fixed method.

*연세대학교 컴퓨터과학과 정보통신연구실
Dept. of Computer Science, Yonsei Univ.
論文番號:96203-0712
接受日字:1996年 7月 12日

I. 서 론

지능망 서비스를 위한 호 처리 과정은 서비스 교환 시스템(SSP: Service Switching Point)에서 지능망 서비스 호가 인식되면 서비스 제어 시스템(SCP: Service Control Point)으로 보내져서 필요한 서비스 처리를 한 후 다시 SSP로 되돌려진다[1, 2, 3]. 하나 또는 소수의 중앙 집중화된 SCP에 동시에 다수의 SSP가 연결되어 서비스를 하는 지능망 구조에서는 SCP에서 서비스 트래픽의 병목 현상이 발생할 가능성이 매우 높다. 따라서 과부하 상태에서 SCP를 보호하는 것은 매우 중요하다.

SCP에서 과부하가 발생했을 때 호를 거절하는 과부하 제어 메커니즘이 필요하며 과부하 제어 방법으로는 SCP가 직접 호를 제어하는 방법과 SCP가 SSP에 제어 메시지를 전송하여 SSP가 호를 제어하도록 하는 방법이 있다. 과부하 발생시 SCP가 직접 호를 제어하는 메커니즘을 사용할 경우 SCP는 서비스를 처리하기보다는 과부하를 제어하는 데 많은 시간을 소비하게 된다. 또한 SSP와 SCP간에 주고받는 메시지로 인한 망자원의 낭비도 커지게 되며 이로 인한 성능의 저하도 심각하게 된다. 따라서 호가 SCP에 도달하기 전에 SSP에서 먼저 호를 차단하여 로드를 제어하는 방식이 바람직하다. 이를 위해서는 SCP에서 과부하를 인식하여 SSP에 자신이 과부하가 발생하였음을 알려주는 방법을 사용한다[1, 3, 4, 5].

본 논문에서는 과부하 제어 방법으로 자동 호차단 방식(ACG: Automatic Call Gapping)을 사용한다[1, 3, 5]. 자동 호차단 방식은 다음과 같다. 과부하 상태를 인지한 SCP는 자신이 과부하 상태임을 알리는 메시지와 호차단시간(gap interval) g 를 SSP에 보낸다. 자동 호차단 방식에 의해 SSP는 어떤 정해진 기간 g 초당 하나의 호 메시지만을 보내도록 호 제어를 하게 한다.

호차단시간 g 는 호차단 방식에서 중요한 요소이다. 모든 SSP에 대하여 동일한 호차단시간을 적용하는 경우 작은 용량의 SSP보다는 트래픽이 많은 대용량의 SSP가 더 높은 비율로 트래픽이 감소하는 문제가 발생한다. 그러므로 이 경우에는 작은 SSP가 트래픽 처리면에서 유리하게 된다[1, 3]. 모든 호가 통과되거나 거절되는 확률을 동일하게 하는 방법으로 이러한

불공정성의 문제를 해결할 수도 있는데 이 경우에는 각 SSP에 다른 호차단시간이 적용된다[4, 6].

서비스가 다양화되고 망의 기능이 보다 지능화되면서 차등화된 가입자 서비스가 가능하게 될 것이다. 우선 순위를 갖는 가입자 서비스는 차등화된 서비스 중의 하나이다. 이러한 우선 순위를 갖는 가입자 서비스는 소방서, 경찰서 등의 긴급한 사항을 다루는 관공서에서 이용되거나 일반 기업의 비상시 연락망으로 사용될 수도 있을 것이다. 우선 순위 서비스는 위와 같이 상당히 유용할 것으로 생각되나 아직까지 우선 순위를 고려한 지능망 구조에 대한 연구는 되어 있지 않다.

따라서 본 논문에서는 우선 순위와 공정성이 고려된 지능망 구조를 다루었다. 공정성이 고려된 과부하 제어 방식으로 자동 호차단 방식을 사용하며 각 SSP에는 각기 다른 호차단시간이 적용된다. 우선 순위에 대해서는 사용자로부터 도착하는 호를 높은 우선순위(High Priority)와 낮은 우선순위(Low Priority)로 분류하고 높은 우선순위를 갖는 호에 먼저 서비스를 제공하도록 하는 방안을 제시한다. 또한 그러한 우선 순위 가입자 서비스 제공 방식이 공정성이 고려된 자동 호차단 방식에서 어떠한 효과를 갖는지에 대하여 수식 및 시뮬레이션을 통하여 비교 분석한다.

II. 지능망에서의 과부하 제어

ACG는 SCP에서는 과부하 상태만을 인지하고 SSP에 과부하 상태임을 알리는 메시지를 전송하여 실제의 과부하 제어 메커니즘은 SSP에 의해 새로 도착하는 호를 필터링하도록 하는 상호 협동적인 과부하 제어 전략이다. 이 장에서는 과부하 제어의 목적을 알아보고 망의 관점에서 ACG의 과정을 면밀히 살펴본다.

2.1 SCP에서의 과부하 제어 과정

먼저 과부하를 정의해야 하며 이를 위해 SCP는 도착하는 TCAP 메시지를 다음의 세 가지 방법으로 측정한다: 평균 응답 시간이나 지연시간, 탈락된 메시지의 수, 입력 메시지 속도. 응답시간은 SCP가 서비스에 대한 요청을 처리하기 위해 걸리는 시간이다. 탈락 메시지는 호의 지연시간이 임계치를 초과하여

SCP가 요청된 서비스를 처리하지 않고 무시한 메시지이다. 입력 메시지 속도는 SCP로 들어오는 총 입력 메시지의 속도로 SCP의 처리속도를 넘지 않아야 한다.

정해진 측정 구간동안 SCP는 세 가지 방법중의 하나를 사용하여 SCP의 과부하 정도를 구해야 한다. 본문에서는 입력 메시지 속도에 의하여 과부하를 측정한다.

2.2 SSP에서의 과부하 제어 과정

SSP와 SCP사이에 지능망호를 처리하기 위해 주고 받는 일련의 메시지들을 트랜잭션이라고 한다. 호 처리는 트랜잭션을 구성하는 메시지들의에도 교환기 사이의 SS No.7 메시지도 생성하지만 여기서는 SCP와 관련된 메시지만을 고려한다. 따라서 ACG는 호차단에 의한 제어가 전체 트랜잭션을 구성하는 일련의 메시지들의 첫번째 질의만을 대상으로 한다. 만일 첫번째 질의 메시지가 SCP에 보내지고 응답을 받는다면 그 트랜잭션은 계속되고 어떤 호차단에 의한 제어도 그 트랜잭션의 후속 메시지에 영향을 끼칠 수 없다. 본 논문에서는 용어의 혼란을 피하기 위해 하나의 지능망 서비스를 처리하기 위한 지능망호는 첫번째 질의 메시지로 시작되고 첫번째 질의 메시지가 허용된 지능망호는 그 서비스가 제공되는 것으로 간주한다. 따라서 지능망호와 질의 메시지를 같은 의미로 구별없이 사용하기로 한다.

ACG는 속도 제어의 형태로 초당 첫번째 질의 메시지의 수를 제한한다. [그림 1]은 ACG의 과정을 보여준다. 화살표는 트래픽 소스에서의 첫번째 질의 시도를 나타낸다. 호차단이 시작되면 하나의 질의 메시지가 일단 통과되고 다음의 모든 질의 메시지들은 호차단시간동안 차단된다. 그러므로 호차단시간 동안 하나의 질의만이 통과된다. 이러한 형태의 호차단 방식은 호차단 주기(gap duration)가 끝날 때까지 반복된다.

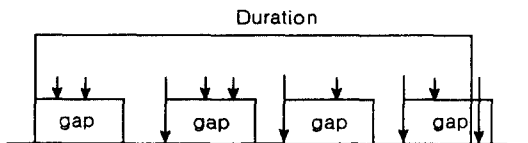


그림 1. SSP에서의 자동 호 차단 방식

2.3 공정성과 우선 순위

모든 SSP에 같은 값을 갖는 호처리시간이 적용되는 ACG 과부하 제어는 모든 트래픽의 소스에 같은 양의 속도 제한을 적용하기 때문에 큰 트래픽 소스가 작은 트래픽 소스보다 더 심각하게 트래픽이 감소하게 된다. 이러한 제어 정책은 작은 트래픽 소스에 더 유리하게 된다. 그런 효과를 해결하기 위해서 먼저 SCP의 과부하 제어의 목적이 무엇인가를 잘 알아야 한다. 좋은 SCP 과부하 제어의 기준은 호 처리율과 공정성이다. 강건한 호 처리율을 보장하는 과부하 제어는 어떤 환경에서도 SCP가 최대의 호 처리율을 제공하도록 하는 것이다. 또한 제어 방식이 공정하다는 것은 미리 정의된 어떤 동등성의 정의에 따라 SCP 용량을 할당하는 것을 의미한다. 강건한 처리율이 다양한 환경 하에서 트래픽 속도의 합이 SCP의 용량에 얼마나 근접한지를 말하는 것이라면 공정성이란 각각의 트래픽 소스의 속도를 말하는 것이다.

본 논문에서는 접속된 SSP에 무관하게 모든 가입자에 대한 서비스 제공이 같은 비율로 거절되도록 하는 것을 공정하다고 가정하고 SSP에서 요청되는 각 서비스를 높은 우선순위와 낮은 우선순위의 두 가지로 분류한다. 우선 순위에 대한 성취도를 얻기 위해 SSP에서는 높은 우선순위의 지능망호를 우선적으로 먼저 통과시키고 SCP에서는 큐에 대기 중인 메시지 중에서 높은 우선순위에 대한 지능망호를 먼저 처리하도록 한다.

III. 우선순위를 고려한 트래픽 환경에서의 과부하 제어

3.1 모델, 정의 및 분석

3.1.1 모델

과부하 제어 방법을 연구하기 위하여 [그림 2]와 같이 하나의 SCP에 n개의 SSP들이 접속되어 있는 모델을 사용한다. SSP에는 높은 우선순위와 낮은 우선순위의 가입자로부터 각각 독립적으로 지능망호가 도착한다. SSP를 통과한 지능망호는 SS No.7 망을 거치고 호의 우선순위에 따라 SCP의 해당 우선순위 큐에 저장된다. SCP는 높은 우선순위와 낮은 우선순위의 두 개의 큐를 가지고 있으며 높은 우선순위의 큐를 먼저 서비스한다. 그러나 높은 우선순위의 지능망호

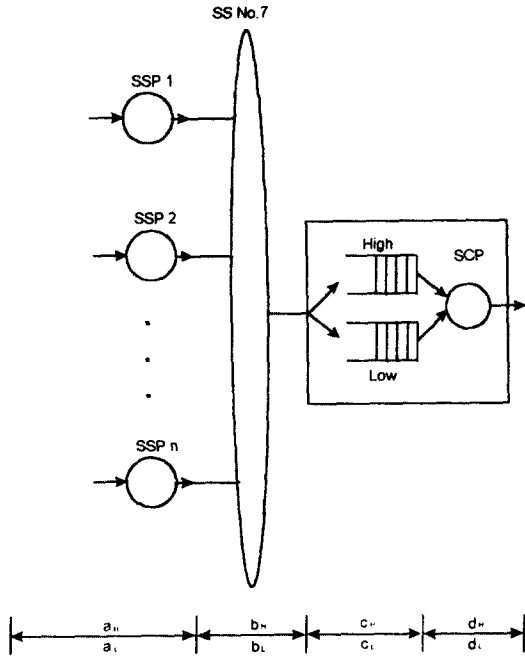


그림 2. 우선 순위를 갖는 트래픽 환경의 모델

는 SCP에서 이미 서비스 중인 낮은 우선순위의 지능망호에 대하여 non-preemptive하게 운영된다.

가입자로부터 입력되는 호들은 SSP에서 거절될 수도 있고 SCP에서 거절될 수도 있다. i 번째 SSP인 SSP _{i} 에는 λ_i 의 속도를 갖는 포아송(Poisson)으로 지능망호가 도착한다고 가정한다. 만일 지능망호가 SSP를 통과한다면 시그널링 시스템, SS No.7 망을 거쳐 SCP에 도착한다.

SSP에서 트래픽을 감소시키는 방법으로 자동 호차단 방식을 이용한다. 즉 SSP에서는 호차단시간, g 동안에 하나의 지능망호를 통과시키는 방식을 취하게 된다. 여기서 호차단시간 g 는 SSP에서 이용되는 일종의 감소 파라미터이다. τ 라고 표시되는 고정 길이를 갖는 제어 구간(Control Interval)이 끝나면 그 구간 동안 도착한 트래픽의 강도에 따라 새로 계산된 호차단시간 g 가 다음 제어 구간 동안에 적용된다.

제어 구간 동안 SCP가 초당 처리할 수 있는 지능망호의 최대 수를 M 이라고 하자. 지능망호에 대한 SCP의 평균 서비스 시간에 따라 M 이 결정된다. 본 논문에서는 SCP가 한 호를 처리하는데 걸리는 시간을 2ms

라 하며, 따라서 $M = 500$, 즉 초당 500개의 메시지를 처리한다고 가정한다.

SSP가 하나의 지능망호를 서비스하기 위하여 SCP측으로 요구하는 질의 메시지의 개수는 지능망 서비스마다 다르다. SCP는 지능망 서비스 호 중에서 일단 첫번째 질의 메시지가 허용되면 나머지 질의 메시지에 대한 응답도 수행해야 한다. 따라서 SCP에서는 제어 구간 τ 동안에 도착한 첫번째 질의 메시지의 개수를 토대로 필요한 파라미터들을 산출한다.

본 연구에서 사용되는 여러 파라미터를 정의하면 다음과 같다.

SSP _{i} : i 번째 SSP, i 는 1에서 n 까지의 값을 갖는다.

τ : 자동 호차단 방식의 제어 구간

g_i : SSP _{i} 에서 자동 호차단 방식의 호차단시간, 매 제어 구간마다 새롭게 정하여진다.

M : SCP가 초당 처리할 수 있는 지능망호의 최대 수

$\lambda_{H,i}$: SSP _{i} 에 도착하는 높은 우선순위 지능망호의 평균 도착율(호/초), 포아송이라고 가정한다.

$\lambda_{L,i}$: SSP _{i} 에 도착하는 낮은 우선순위 지능망호의 평균 도착율(호/초), 포아송이라고 가정한다.

λ_i : SSP _{i} 에 도착하는 모든 지능망호의 평균 도착율(호/초), 높은 우선순위 및 낮은 우선순위 지능망호가 모두 포아송이므로 그 둘의 합도 역시 포아송이 된다.

λ_H : 모든 SSP에 도착하는 높은 우선순위 지능망호의 평균 초당 호수(호/초)

λ_L : 모든 SSP에 도착하는 낮은 우선순위 지능망호의 평균 초당 호수(호/초)

λ : 모든 SSP에 도착하는 모든 지능망호의 평균 초당 호수(호/초)

$\lambda'_{H,i}$: SSP _{i} 를 통과하여 나가는 높은 우선순위 지능망호 평균 초당 호수(호/초)

$\lambda'_{L,i}$: SSP _{i} 를 통과하여 나가는 낮은 우선순위 지능망호 평균 초당 호수(호/초)

λ_i : SSP _{i} 를 통과하여 나가는 모든 지능망호의 평균 초당 호수(호/초)

$\lambda_{H,i}$: SCP의 높은 우선순위 큐에 도착하는 지능망호의 평균 초당 호수(호/초)

λ_L : SCP의 낮은 우선순위 큐에 도착하는 지능망호의 평균 초당 호수(호/초)

- λ' : SCP의 두개의 큐에 도착하는 모든 지능망호의 평균 초당 호수(호/초)
- λ_H' : SCP에서 서비스 받고 나가는 높은 우선순위 지능망호의 평균 초당 호수(호/초)
- λ_L' : SCP에서 서비스 받고 나가는 낮은 우선순위 지능망호의 평균 초당 호수(호/초)
- λ'' : SCP에서 서비스 받고 나가는 모든 지능망호의 평균 초당 호수(호/초)

위의 정의로부터 다음과 같은 관계가 성립한다.

$$\lambda_i = \lambda_{H,i} + \lambda_{L,i} \quad \lambda_H = \sum_{i=1}^n \lambda_{H,i} \quad \lambda_L = \sum_{i=1}^n \lambda_{L,i} \quad \lambda = \lambda_H + \lambda_L$$

$$\lambda'_i = \lambda'_{H,i} + \lambda'_{L,i} \quad \lambda'_H = \sum_{i=1}^n \lambda'_{H,i} \quad \lambda'_L = \sum_{i=1}^n \lambda'_{L,i} \quad \lambda' = \lambda'_H + \lambda'_L$$

3.1.2 공정성의 정의 및 측정

많은 논문에서 공정성을 여러 가지 의미로 정의하지만 여기서는 어느 시점에서 지능망호가 SSP를 통과하거나 거절될 확률이 동일한 것을 공정한 것으로 정의하며 즉, 다음과 같다.

$$\frac{\lambda'_i}{\lambda_i} = \frac{\lambda'_j}{\lambda_j} \text{ for all } SSP_j, SSP_j \quad (3.1)$$

3.1.3 호 처리율의 정의 및 측정

호 처리율은 과부하 상태 하에서 지능망호가 얼마나 서비스되는지를 측정하기 위한 파라미터로서 각 SSP당 트래픽의 입력에 대한 SCP에서 처리되는 호의 비율로 구해진다. 그러므로 호 처리율은 다음과 같이 정의된다.

$$\text{호 처리율} = \frac{\lambda''}{\lambda} \quad (3.2)$$

3.2 과부하 측정 방법

SCP에서 과부하를 측정하는 방법에는 여러가지가 있다. 그 중에서 최대 지연 시간에 의한 것과 입력 메시지의 수에 의한 것을 살펴보겠다.

최대 지연 시간에 의한 과부하 측정은 제어 구간 동안 어떤 메시지가 정해진 시간이 지날 때까지 서비스를 받지 못하고 지연되고 있다면 SCP는 과부하가 발생한 것으로 보는 방법이다. 이 방법은 SCP의 큐에 도착한 지능망호가 지연시간이 지난 후 서비스를 받

지 못하고 무시되므로 가입자들은 긴 시간을 기다린 후 호가 끊어져버리는 서비스의 불편함을 받게된다.

입력 메시지의 수에 의한 방법은 제어 구간동안 SSP에 들어온 지능망호 입력 메시지의 수가 SCP의 처리 용량에 대한 임계값보다 크가를 비교하여 과부하를 결정한다. 이 방법은 지능망호에 대한 SCP의 서비스 시간이 다양하다면 입력 메시지의 수가 SCP의 작업 부하량을 정확하게 나타내지 못하게 된다.

본 논문에서는 입력 메시지에 의한 방법을 사용하여 SCP에서의 과부하를 측정하며 지능망호에 대한 서비스 시간은 일정한 것으로 가정한다. 즉 제어 구간당 입력 메시지의 수가 SCP의 처리 용량의 임계값보다 크면 과부하 상태로 판단하여 자동 호차단 방식을 수행한다. 또한 임계값보다 작은 경우 과부하를 해지하며 이미 자동 호차단 방식이 수행중일 경우에는 이를 해지한다.

3.3 과부하 제어 방법

SCP에서 과부하가 발생했다는 것을 알았을 때, SCP는 SSP와 필요한 정보를 주고받으면서 SSP의 트래픽 양에 따라 호차단시간을 조절하여 SSP로부터 오는 트래픽을 감소시킨다. SSP에서의 지능망 트래픽의 조절은 자동 호차단 방식에 따른다. 필요한 파라미터는 제어 구간 τ 가 끝나는 시점마다 모든 SSP에서 SCP로 보고되는 호의 총수 λ 와 SCP의 처리 용량 등을 고려하여 산출된다. 본 연구에서는 시간축을 호차단시간 간격 g 로 균등하게 나누어 각 호차단시간마다 하나의 지능망호만이 통과하는 고정방식과 호차단시간이 끝난 후 처음 도착하는 지능망호가 새로운 호차단시간을 발생시키는 호 도착발생방식을 제안한다.

3.3.1 고정된 호 차단 시간을 사용한 방법 : 고정 방식

고정된 호 차단 시간이란 시간축을 간격 g 로 균등하게 구분하여 호차단시간 g 동안 입력되는 지능망호가 높은 우선순위일 경우는 바로 통과시키고 나머지 지능망호들은 차단시킨다. 입력되는 지능망호가 낮은 우선순위일 경우는 호차단시간이 끝날 때까지 기다린다. 그동안 높은 우선순위의 지능망호가 하나도 들어오지 않을 경우에는 기다리던 지능망호를 보내며, 높은 우선순위의 지능망호가 도착하는 경우에는 그 높은 우선순위의 지능망호가 곧바로 통과하고 다

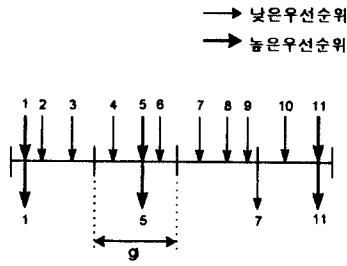


그림 3. 고정방식을 이용한 호제어 방식

른 지능망호들은 차단된다. 이와 같은 과정의 동작 원리를 [그림 3]에 나타내었다.

SSP_i에서 호차단시간 g_i 동안 높은 우선순위의 지능망호가 올 확률을 $K(H, i)$ 라 하고, g_i 동안 높은 우선순위의 지능망호는 오지 않으면서 낮은 우선순위의 지능망호가 올 확률을 $K(L, i)$ 라 하고 g_i 동안 어느 지능망호도 도착하지 않는 확률을 $K(No, i)$ 라 하면 다음과 같은 결과를 얻는다.

$$K(H, i) = 1 - e^{-\lambda_{H,i} g_i} \tag{3.4}$$

$$K(L, i) = e^{-\lambda_{H,i} g_i} \cdot (1 - e^{-\lambda_{L,i} g_i}) \tag{3.5}$$

$$K(No, i) = e^{-\lambda_{H,i} g_i} \cdot e^{-\lambda_{L,i} g_i} = e^{-\lambda_i g_i} \tag{3.6}$$

따라서 $\lambda'_{H,i}$ 와 $\lambda'_{L,i}$ 는 다음과 같이 구해진다.

$$\lambda'_{H,i} = \frac{K(H, i)}{g_i} = \frac{1 - e^{-\lambda_{H,i} g_i}}{g_i} \tag{3.7}$$

$$\lambda'_{L,i} = \frac{K(L, i)}{g_i} = \frac{e^{-\lambda_{H,i} g_i} (1 - e^{-\lambda_{L,i} g_i})}{g_i} \tag{3.8}$$

높은 우선순위와 낮은 우선순위에 대하여 지능망호가 SSP_i를 통과할 확률, $Pass_{H,i}$ 와 $Pass_{L,i}$ 는

$$Pass_{H,i} = \frac{\lambda'_{H,i}}{\lambda_{H,i}} = \frac{1 - e^{-\lambda_{H,i} g_i}}{\lambda_{H,i} g_i} \tag{3.9}$$

$$Pass_{L,i} = \frac{\lambda'_{L,i}}{\lambda_{L,i}} = \frac{e^{-\lambda_{H,i} g_i} (1 - e^{-\lambda_{L,i} g_i})}{\lambda_{L,i} g_i} \tag{3.10}$$

이다.

호 차단시간 g_i 는 다음 제어구간에는 새롭게 계산되어 적용된다. 새로운 호차단시간을 계산하기 위해서는 현 제어구간 동안에 도착한 호의 속도 λ_i 를 이용

한다. 제어구간이 끝났을 때 SCP는 각 SSP_i로부터 λ_i 를 보고 받고 다음 제어구간에도 역시 λ_i 만큼의 트래픽이 도착할 것으로 가정하여 새로운 호차단시간을 구하며 이를 각 SSP에 전달한다. SCP의 유효처리능력을 R 이라고 한다면 R 은 SCP의 최대처리능력에 여유비율 α 를 곱한 것으로 표시된다. 즉,

$$R = M \cdot \alpha \tag{3.11}$$

SCP의 호처리비율 h 는

$$h = \frac{R}{\lambda} \tag{3.12}$$

SCP에서의 호 처리 비율 h 는 SSP_i에 호차단시간 g_i 동안 도착하는 지능망호의 통과확률과 같게 한다. 따라서,

$$h = \frac{\lambda'_i}{\lambda_i} = \frac{1}{\lambda_i} (\lambda'_{H,i} + \lambda'_{L,i}) = \frac{1 - e^{-\lambda_i g_i}}{\lambda_i g_i} \tag{3.13}$$

이며 위의 식으로부터 g_i 를 구해낼 수가 있다.

고정된 호차단시간을 사용하여 과부하 제어를 할 경우 위에서 구한 것과 같이 호차단시간 g_i 를 각 SSP_i에 대한 입력 트래픽을 고려하여 산출한 후 각 SSP_i별로 별도의 호차단시간 g_i 를 적용하게 된다. 이로 인하여 각 SSP_i에 대하여 입력 트래픽의 양에 따른 비율로 공정한 호차단시간을 적용하게 되므로 목적하는 공정성을 얻을 수 있다.

3.3.2 새로운 도착호에 의한 호차단시간의 시작 방법 : 도착발생방법

새로운 도착호에 의한 방식은 각 SSP_i에서 호차단시간 g_i 가 끝난 이후에 지능망호가 새롭게 도착할 때 새로운 호차단시간이 시작되는 방식이다. 새롭게 도착하는 지능망호가 높은 우선순위인 경우는 바로 SSP_i를 통과시키고 그 후 호차단시간 g_i 동안 도착하는 지능망호는 차단한다. 새롭게 처음 도착한 지능망호가 낮은 우선순위인 경우는 이어지는 호차단시간 g_i 동안 높은 우선순위의 지능망호가 도착하지 않는다면 그 낮은 우선순위의 지능망호가 호차단시간 g_i 가 끝나는 시점에서 SSP_i를 통과하게 된다. 만약 g_i 동안 높은 순위의 지능망호가 도착하면 즉시 그 높은

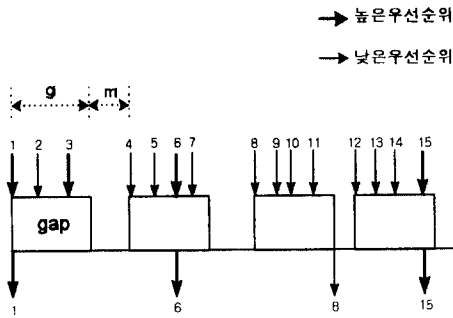


그림 4. 도착발생방식을 이용한 호제어 방법

순위의 지능망호가 SSP를 통과한다. [그림 4]는 이와 같은 과정을 보여준다. [그림 4]에서 m_i 는 호 차단 구간과 호차단구간의 사이를 나타내는 랜덤 변수이다.

높은 우선순위 및 낮은 우선순위의 지능망호가 각각 포아송으로 도착하므로 각 호차단시간 g_i 동안에 높은 우선순위의 지능망호가 통과할 확률 $K(H, i)$ 와 각 호차단시간 g_i 동안에 낮은 우선순위의 지능망호가 먼저 도착하고 높은 순위의 지능망호는 오지 않을 확률 $K(L, i)$ 는 아래와 같이 계산된다.

$$K(H, i) = \frac{\lambda_{H,i}}{\lambda_{H,i} + \lambda_{L,i}} + \frac{\lambda_{L,i}}{\lambda_{H,i} + \lambda_{L,i}} \cdot (1 - e^{-\lambda_{H,i} g_i}) \quad (3.14)$$

$$K(L, i) = \frac{\lambda_{L,i}}{\lambda_{H,i} + \lambda_{L,i}} e^{-\lambda_{H,i} g_i} \quad (3.15)$$

랜덤 변수 m_i 의 평균값은 $\frac{1}{\lambda_i}$ 가 되므로 초당 SSP_i를 통과하는 지능망호의 초당 평균 호수 G_i 는 다음과 같이 표시된다.

$$G_i = \frac{1}{\frac{1}{\lambda_i} + g_i} \quad (3.16)$$

높은 우선순위의 호와 낮은 우선순위의 호가 SSP_i를 통과할 확률 $Pass_{H,i}$ 와 $Pass_{L,i}$ 는

$$Pass_{H,i} = \frac{K(H, i) G_i}{\lambda_{H,i}} \quad (3.17)$$

$$Pass_{L,i} = \frac{K(L, i) G_i}{\lambda_{L,i}} \quad (3.18)$$

이고 SSP_i를 통과하는 높은 우선순위의 낮은 우선순위 호의 속도는 각각 다음과 같다.

$$\lambda'_{H,i} = K(H, i) \cdot G_i \quad (3.19)$$

$$\lambda'_{L,i} = K(L, i) \cdot G_i \quad (3.20)$$

다음 제어구간에 적용될 새로운 호차단시간을 구하기 위하여 다음과 같은 과정을 따른다. SSP_i에 대한 지능망호의 총도착율에 대한 SSP_i에서의 호의 총통과율의 비율은, 모든 SSP에 대한 총 도착율 λ 대 SCP의 유효처리용량의 비율인 h 와 같게 한다. 따라서,

$$\frac{G_i}{\lambda_i} = h \quad (3.21)$$

이며 위의 식으로부터 g_i 를 구하면 다음과 같다.

$$g_i = \frac{1-h}{h\lambda_i} \quad (3.22)$$

도착발생방식에 따른 과부하 제어도 역시 고정방식의 경우와 마찬가지로 호차단시간 g_i 를 각 SSP_i에 대한 입력 트래픽의 양을 고려하여 산출, 적용하게 되므로 원하는 공정성을 얻을 수 있다.

3.4 우선 순위 성취도

높은 우선순위 지능망호가 모든 낮은 우선순위 지능망호보다 우선적으로 먼저 처리되는 것이 원칙이나 각 SSP의 공정성을 보장하는 구조에서는 실질적으로 지켜지기 힘들다. 그러므로 우선순위가 100% 잘 지켜지지 않게 되는 구조에서는 우선순위가 얼마나 잘 지켜지는가를 나타내는 척도가 필요하게 된다. 그러므로 이 절에서는 그 척도가 되는 우선순위 성취도를 정의한다.

SSP의 총입력 트래픽에 대한 SCP에서의 유효처리 비율을 h 라 하고 모든 SSP에 도착하는 높은 우선순위 지능망호의 입력 트래픽 λ_H 와 낮은 순위 지능망호의 입력 트래픽 λ_L 의 합에 대한 높은 우선 순위의 입력 트래픽의 비율을 h' 라고 하면 다음과 같다.

$$h' = \frac{\lambda_H}{\lambda_H + \lambda_L} \quad (3.23)$$

$h \geq h'$ 인 경우는 높은 우선순위의 입력 트래픽이 SCP에서의 처리 능력보다 적으므로 이상적인 경우에는 높은 우선순위의 지능망호는 모두 서비스를 받아야 하고 일부 낮은 순위의 지능망호도 서비스 받을 수 있다. $h < h'$ 인 경우는 높은 우선 순위의 입력 트래픽이 SCP에서의 처리 능력을 초과하여 도착하므로 이상적인 경우에는 낮은 우선순위의 지능망호는 서비스를 받을 수 없고 높은 우선순위의 지능망호일지라도 일부 거절된다. 그러므로 각각의 경우의 우선 순위 성취도는 다음과 같이 정의한다.

$$h \geq h' \text{인 경우: } P = \frac{\lambda'_H}{\lambda_H} \quad (3.24)$$

$$h < h' \text{인 경우: } P = \frac{\lambda'_H}{\lambda'_H + \lambda'_L} \quad (3.25)$$

지능망호에 대한 우선순위 성취도는 다음과 같이 하나의 식으로 표시될 수 있다.

$$P = \frac{\lambda'_H}{\min(\lambda_H, \lambda'_H + \lambda'_L)} \quad (3.26)$$

IV. 시뮬레이션 및 결과 분석

4.1 시뮬레이션 모델

3.3절에서 제안한 두 가지의 과부하 제어 방법을 연구하기 위하여 [그림2]와 같은 지능망 모델을 이용하여 시뮬레이션을 수행하였다. SCP는 지능망 서비스 작업을 요청하는 여러 개의 SSP들에 접속되었고 SSP들은 입력 트래픽의 소스로, 여러 그룹으로 나뉘어 각각 다른 트래픽 사이즈를 SCP에 제공한다. 시뮬레이션에 필요한 파라미터들은 다음에서 정의된다.

4.1.1 SCP의 처리능력에 따른 파라미터

SCP의 처리능력 M은 500 호/초로 하나의 지능망호가 성공적으로 SCP에서 처리되거나 과부하로 인한 지연시간 초과로 거절되는 데는 2ms가 소요된다고 가정하고 호의 거절에 따른 가입자들의 재시도는 고려하지 않았다. SCP에서의 과부하 상태 측정은 제어 구간당 모든 SSP의 총 입력메시지의 수가 SCP의 유효처리용량을 넘으면 과부하 상태로 인지하는 방법을 사용하였다. 질의 메시지에 대한 SCP의 큐에서의

최대 지연시간의 임계치는 1초로 하고 이 임계치를 초과한 경우는 SCP에서 해당하는 호의 처리를 거절하였다. 본 연구에서는 SCP의 여유비율을 $\alpha = 0.8$ 로 가정하여 유효처리용량은 초당 $R = 400$ 호/초로 정하였다. 또한 SSP의 총 입력 트래픽이 SCP의 유효처리용량에 해당될 때의 트래픽을 100% 부하로 표현한다.

SCP는 높은 우선순위와 낮은 순위의 큐를 두 개 가지고 있어서 높은 우선순위의 큐에 대기 중인 모든 지능망호를 먼저 처리한 후 낮은 순위의 지능망호를 처리한다. 또한 우선 순위 처리 방법은 non-preemptive 방식으로 이루어진다.

4.1.2 과부하 발생에 필요한 파라미터

하나의 SCP에 접속된 SSP의 수는 80개로 하고 SSP들을 4개의 그룹으로 나누어 각 그룹간의 입력 트래픽의 비율은 1:2:3:4가 되게 하였다. 즉 각 그룹은 20개의 SSP로 이루어진다. SCP의 유효처리용량에 대한 SSP 입력의 과부하 비율은 50%에서부터 100, 200, 300, 400, 500%씩 증가하도록 하고 과부하 제어과정을 관찰하기 위하여 한 단계당 500초씩 관찰하였다. 또한 높은 우선순위와 낮은 우선순위의 트래픽 입력의 비율은 1:3이 되도록 하였다.

SSP에 도착하는 지능망 트래픽은 포아송을 따른다고 가정하였다. 과부하 제어에 필요한 제어 구간 τ 는 2초로 하고 제어 구간의 끝에서 과부하 상태인지를 감시하여 과부하가 측정되지 않으면 자동 호차단 방식의 구동을 중지하고 과부하가 측정되면 각 SSP, 별로 호차단시간을 산출하여 자동 호차단 방식을 구동한다.

호 차단 방식은 3.3절에서 제안한 두 가지 방식을 각각 따로 사용하여 시뮬레이션을 수행하였으며 각 방식에 따른 결과는 다음절에 설명하였다.

4.2 시뮬레이션의 결과 및 분석

이 절에서는 앞 절에서 정의한 시뮬레이션 모델과 파라미터 구성에 따라 시뮬레이션을 수행한 결과를 분석하도록 하겠다. SSP로의 트래픽 입력은 포아송을 가정하고 트래픽의 입력율이 다른 SSP 그룹간의 공정성, 지능망호의 처리율 및 우선순위 성취도 등을 분석한다.

4.2.1 공정성

[그림 5]와 [그림 6]은 각각 고정방식과 도착발생방식의 경우에 SSP에 들어오는 로드의 변화에 따라 SSP 그룹들이 얼마나 공정하게 서비스를 받는지를 보여준다. 그림을 보면 두 가지 방식 모두 로드가 증가함에 따라 SSP에 입력되는 트래픽에 비해 SSP를 통과하는 트래픽의 비율은 줄어들지만, 트래픽 제공 비율이 다른 SSP 그룹간에 공정성은 로드의 변화에 상관없이 일정하게 유지됨을 볼 수 있다. SSP 그룹간의 트래픽 제공 비율이 1:2:3:4로 SSP 그룹 4의 트래픽 입력 정도가 가장 많지만 SSP 그룹간에 적용되는 호차단시간이 트래픽 정도에 따라 다르므로 같은 확률로 지능망호가 서비스됨을 알 수 있다. 과부하의 비율이 50%, 100%, 200%, 300%, 400%, 500% 일 때 SSP에서의 통과확률은 이론적으로는 SSP 그룹에 관계없이 1, 1, 0.5, 1/3, 0.25, 0.2로 계산되며 이는 시뮬레이션 결과와 거의 일치하는 것을 알 수 있다.

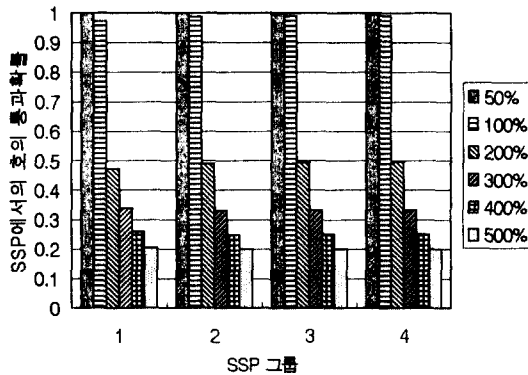


그림 5. SSP 그룹간의 공정성의 비율(고정방식)

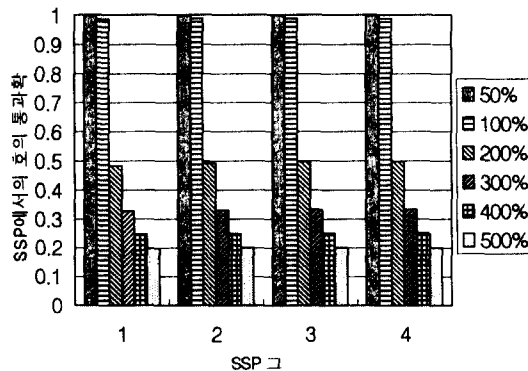


그림 6. SSP 그룹간의 공정성의 비율(도착발생방식)

4.2.2 호 처리율

호 처리율은 지능망 서비스를 사용하는 가입자가 얼마나 최대한의 서비스를 받는지를 나타내는 값으로 입력 로드에 대해서 얼마나 많은 지능망호가 궁극적으로 SCP에서 서비스되는 지를 구한다. 즉 SSP의 입력 트래픽이 얼마나 SCP의 서비스를 받는가를 나타내는 척도로 [그림 7]과 [그림 8]에서 정의한 심볼은 $\frac{\lambda''}{\lambda}$ 로 표시된다. [그림 7]과 [그림 8]을 보면 높은 우선순위와 낮은 우선순위가 각각 서비스되는 처리율이 입력 로드의 증가에 따라 점점 감소함을 보여준다. 또한 낮은 순위의 지능망호가 높은 우선순위의 지능망호에 비해 더 많은 트래픽 입력을 제공함에도 불구하고 높은 우선순위의 지능망호가 더 많이 처리됨을 알 수 있다.

과부하에 대한 제어는 SCP에서 호를 거절하는 것 보다는 SSP에서 호를 거절하는 것이 효율적이다. 이것이 만족될 경우 단위시간 당 전체 거절된 호의 수에 대한 SSP에서 거절된 호의 수의 비는 $\frac{\lambda - \lambda'}{\lambda - \lambda''} \approx 1$ 이므로 $\lambda' \approx \lambda''$ 의 관계를 갖는다. 이같은 경우에 호 처리율은 $\frac{\lambda'}{\lambda}$ 로 근사화할 수 있다. 다음 [표 1]과 [표 2]는 시뮬레이션과 수식적으로 계산된 $\frac{\lambda'}{\lambda}$ 의 값을 비교한 것이다. [표 1]과 [표 2]에서 보다시피 계산된 값과 시뮬레이션 값이 거의 일치함을 알 수 있다.

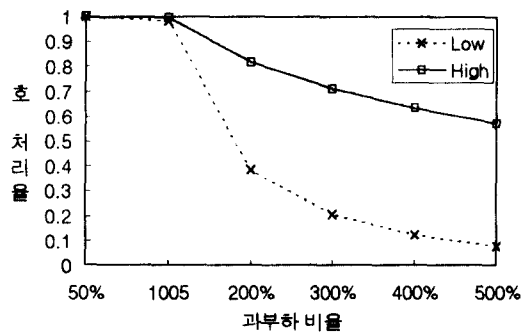


그림 7. 호 처리율(고정방식)

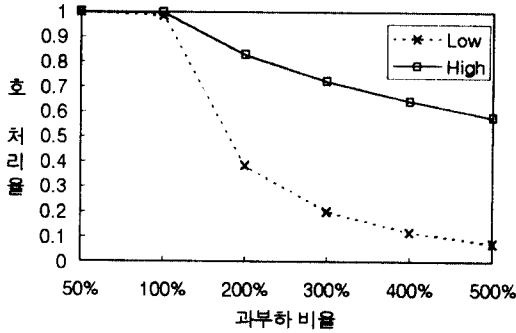


그림 8. 호 처리율(도착발생방식)

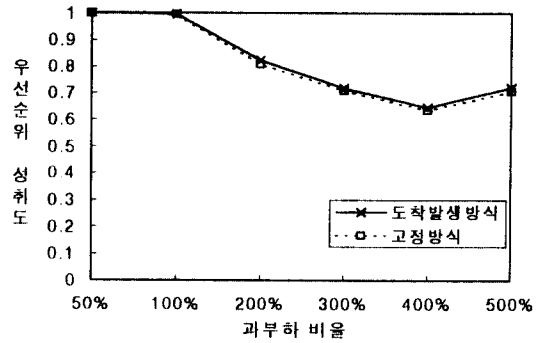


그림 9. 우선 순위 성취도(시뮬레이션)

표 1. 수식에 의한 호처리율(고정방식)

과부하 비율	High		Low	
	수식	시뮬레이션	수식	시뮬레이션
100%	0.999123	0.995770	0.995624	0.982074
200%	0.824799	0.814897	0.391697	0.384180
300%	0.717453	0.709609	0.205278	0.205918
400%	0.637400	0.632959	0.120876	0.122370
500%	0.572793	0.569651	0.075743	0.077592

표 2. 수식에 의한 호처리율(도착발생방식)

과부하 비율	High		Low	
	수식	시뮬레이션	수식	시뮬레이션
100%	1.0	0.996632	1.0	0.983186
200%	0.831799	0.826301	0.389400	0.384053
300%	0.726803	0.720095	0.202177	0.201172
400%	0.645725	0.640825	0.118192	0.118009
500%	0.579272	0.576389	0.073576	0.073981

4.2.3 우선 순위 성취도

이 논문에서는 지능망호가 높은 우선순위와 낮은 우선순위의 두 가지로 구별하여 입력된다고 가정하기 때문에 높은 우선순위가 얼마나 우선적으로 서비스되는지에 대한 척도가 필요하다. 이에 대하여 앞장에서 정의한 식을 사용하여 과부하 제어를 위한 두 가지 방식을 사용한 경우에 대하여 각각 값을 산출하면 다음 [그림 9]와 같다. 우선 순위에 대한 입력 트래픽의 비율은 높은 순위와 낮은 순위의 비율이 1:3으로

로 낮은순위의 지능망호가 더 많이 입력되지만 높은 우선순위의 지능망호가 더 우선적으로 서비스되어야 한다. [그림 9]를 보면 도착발생방식이 고정차단방식에 비하여 우선 순위 성취도가 더 우수함을 알 수 있다. 또한 [표 3]에는 분석값과 시뮬레이션값을 비교하였다. 시뮬레이션과 수식이 일치함을 알 수 있다.

표 3. 수식에 의한 우선순위성취도

과부하 비율	고정방식		도착발생방식	
	수식	시뮬레이션	수식	시뮬레이션
100%	0.999123	0.994467	1.0	0.996603
200%	0.824799	0.808172	0.831799	0.824239
300%	0.717453	0.708186	0.726803	0.718546
400%	0.637400	0.634482	0.645725	0.644571
500%	0.715071	0.705952	0.744090	0.720828

V. 결 론

본 논문에서는 지능망에서의 과부하 제어를 위한 두 가지 다른 메커니즘을 제시하였다. 이 메커니즘들은 SCP와 SSP가 서로 상호 협동하여 문제를 해결하기 때문에 가입자에게 불편함을 주는 것을 피하고 과부하에 대하여 즉각적으로 반응하여 동적으로 각 트래픽 소스에 영향을 끼치도록 하였다.

본 논문에서 제안한 메커니즘들은 간단한 알고리즘을 사용하여 동시에 SCP에 접속되어 서비스를 받

는 SSP들간에 최대한으로 공정성을 유지하며 높은 순위의 지능망호에 우선적으로 서비스를 제공하도록 하였다. 본 논문에서 제안하는 메커니즘은 고정방식과 도착발생방식이다. 이 두 가지 방식은 알고리즘에 있어서 기본적으로 ACQ를 사용하지만 호차단 방법에 있어서는 차이를 가진다. 두 가지 방식 모두 호 처리율에 있어서 비슷한 결과를 보이며 모든 지능망호가 같은 확률로 서비스되는지를 비교하는 공정성도 트래픽 입력에 상관없이 모든 SSP에 대하여 같은 값으로 유지됨을 알 수 있다. 또한 우선 순위 성취도는 도착발생방식이 고정방식에 비하여 약간 우수함을 알 수 있다.

참 고 문 헌

1. G. Hebuterne, L. Romoef, and R. Kung, "Load Resulation schemes for the intelligent Network," in Proc. XIII Int. Switching Symp., Stochholm, Sweden, 1990.
2. ITU-T SG11, "CCITT New Recommendation Q.12XX Series," 1993.
3. 정현필, 이윤현, "지능망에서의 과부하 제어방식," 한국통신학회 논문집 '95-1, Vol. 20, No. 1, pp. 221-229
4. M.Y.Chung and D.K.Sung, "Performane Evaluation of Intelligent Networks Accommodation Various IN Services as well as Basic ISDN Services," International Conference on Computer Communication '95, pp. 195-200.
5. Donald E. Smith, "Ensuring Robust Call Throughput and Fairness for SCP overload Controls," IEEE/ACM Trans. on Networks, Vol. 3, No. 5, October 1995.
6. Maria Kihl and Christian Nyberg, "Transient and Stationary Investigations of Overload Control in Intelligent Networks," International Conference on Computer Communication '95.
7. W. berger, "Comparision of Call gapping and Percent Blocking for Overload Control in Distributed Switching Systems and Telecommunicationa Networks," IEEE Trans. on Commun, Vol. 39, No. 4, April

1991.

8. Pham X and Betts R, "Congestion control for intelligent network," 1992 International Zurich Seminar on Digital Communications.
9. Tsolas N, Abdo G and Botthem R, "Performance and Overload Considerations when introducing IN into an existing Network," 1992 International Zurich Seminar on Digital Communications
10. R. A. Farel and M. Gawande, "Design and anaysis of overload control strategies for transaction network databases," in Proc. 13th Int. Teletraffic Congr., Copenhagen, Denmark, June 1991.



李 庸(Yong Lee) 정회원

1966년 1월 25일생

1988년 2월:연세대학교 식품공학과 졸업(학사)

1992년 2월:덕성여자대학교 전산학과 졸업(학사)

1993년 3월~1994년 5월:디지콤 정보통신연구소 근무

1996년 8월:연세대학교 대학원 컴퓨터과학과 졸업(석사)

현재:연세대학교 대학원 컴퓨터과학과 박사과정 재학중

※주관심분야: Intelligent Network, ATM network, Network Security

宋 周 錫(Joo Seok Song)

정회원

제20권 제3호 14쪽 참조