

# 계층적 도메인간 부하분산 QoS 라우팅

홍종준<sup>†</sup> · 김승훈<sup>††</sup> · 이균하<sup>†††</sup>

## 요약

본 논문에서는 대규모 네트워크에서의 부하분산을 고려한 계층적 도메인간 QoS 라우팅을 제안하였다. 이를 위해 계층적 라우팅에서의 비용 산정 방식을 선계산 방식의 경로계산에서 제안하였고 선계산된 다중경로들 중에서 경로를 선택하는 방안을 제안하였다. 제안한 비용산정 방식은 도메인으로 분할된 대규모 네트워크의 QoS 라우팅에서 선계산 방식으로 경로를 설정하고, 이들 경로들의 예약 가능성을 높이기 위해 자원 예약 측면에서 가장 좋은  $K$  개의 경로들을 부하분산을 고려하여 계산한다. 또한 이러한 다중 경로들 중에서 비용산정에 근거한 경로선택확률에 따라 경로를 선택하는 방안을 제안하였다. 그리고 제안한 QoS 라우팅 방식을 transit traffic과 intra traffic에 모두 적용하여 도메인간 라우팅과 도메인내 라우팅을 무리 없이 연계하면서 transit traffic을 우선으로 하는 방식을 제안하였다. 따라서 본 논문에서 제안한 부하분산 QoS 라우팅은 전체 네트워크 자원의 최적사용 뿐만 아니라 자원의 부하 분산을 가능하게 할 것으로 기대된다.

## Hierarchical Inter-domain Load Balanced QoS Routing

Jong-Joon Hong<sup>†</sup> · Seung-Hoon Kim<sup>††</sup> · Kyoon-Ha Lee<sup>†††</sup>

## ABSTRACT

In this paper, we propose hierarchical inter-domain load balanced QoS routing in a large scale of network. And a scheme of calculating resources' costs for precomputation schemes in hierarchical routing is proposed, and a scheme for selecting a path among multiple precomputed paths is proposed. We also propose a QoS routing scheme for transit and intra traffic in a large scale of domain-based network. For a domain in the network, the routing scheme first precomputes  $K$  multiple paths between all pairs of ingress and egress border routers while considering balancing of the expected load then selects paths with the probability of path selection. The routing scheme combines inter and intra domain routings seamlessly and uses the same cost calculation scheme. And these schemes make possible both global network resources' utilization and also load balancing.

키워드 : QoS 라우팅(QoS Routing), 계층적 라우팅(Hierarchical Routing), 부하분산(Load Balancing)

## 1. 서론

QoS 라우팅은 경로 설정을 위해 사용할 링크 및 노드의 비용을 산정하고 이에 근거하여 라우팅이 이루어진다[4, 12]. 네트워크 자원을 고려한 기존의 연구들은 대부분 비용산정 단계가 아닌 라우팅 단계에서 전체 네트워크 자원의 사용을 고려하였다[1, 4]. 또한 네트워크 일부지역에서의 혼잡을 방지하기 위하여 전체 네트워크에 걸쳐 각 자원의 부하가 분산되도록 하여 사용될 필요가 있다. 한편, 응용에서 요구하는 QoS를 만족하는 경로를 사용할때 네트워크 자원을 실제로 소모하게 되며, 이를 많은 논문에서 QoS 라우팅에서 경로 사용을 위한 비용요소로 다룬다[1-4, 12]. 본 논문에서는 이러한 실제 자원사용 비용과 별도로 QoS 라우팅에서 경로

설정을 위하여 사용할 비용을 산정하는 방식을 제안하여 비용산정시 부하분산을 위한 요소도 함께 고려하도록 하였다. 본 논문에서는 전자를 자원사용 비용, 후자를 경로설정 비용이라 한다. 어떤 요청이 링크를 사용할 경우 네트워크의 부하에 따라 요청의 비용이 결정되므로 사용자에게 불합리한 측면이 있다. 따라서 링크 사용에 따른 자원사용 비용과 QoS 라우팅의 경로 설정에 이 링크를 사용할 경우 네트워크에 초래되는 비용인 경로설정 비용을 분리할 필요가 있다. 주어진 경로설정 비용을 최소화한 경로가 네트워크 자원을 균등하게 사용하는 QoS 경로가 된다. 이러한 비용 산정을 통해 전체 네트워크 자원의 최적사용은 물론 부하분산을 고려한 QoS 경로 설정이 가능해진다. 본 논문에서 앞으로 비용이라 하면 경로설정 비용을 의미한다.

본 논문에서는 부하분산 QoS 라우팅을 위해 선계산 방식(precomputation scheme)[1, 4, 5]의 QoS 라우팅에서 비용산정 방식을 제안한다. 또한 QoS 경로를 미리 계산할때 주어

† 정희원 : 청강문화산업대학 컴퓨터소프트웨어과 교수  
 †† 정희원 : 단국대학교 전자·컴퓨터학부 교수  
 ††† 종신희원 : 인하대학교 컴퓨터공학부 교수  
 논문접수 : 2002년 7월 26일, 심사완료 : 2002년 9월 2일

진 송신자/수신자간에 QoS 요구를 만족하는 경로를 하나만을 설정한다면, 발생할 수 있는 경로의 과부하를 줄이기 위해 경로의 전부 혹은 일부를 공유하지 않고, 자원 예약 측면에서 가장 좋은  $K$  개의 경로를 미리 설정하고 요청시 선택하도록 하였다[8, 10, 11].

본 논문에서는 부하 분산을 위한 요소를 경로 설정을 위한 계산의 비용에 포함하도록 하여, 실제로 요청의 자원 예약시 선택된 경로상의 자원 예약의 가능성을 높이고자 한다. 계산된 경로들 중에서 경로선택은 경로설정 요구시 선계산 방식으로 결정된  $K$  다중 경로중에서 하나의 경로를 선택해야 한다. 이에 본 논문에서는 선계산 방식에서의 비용산정 방식에서 사용된 예측부하율과 경로선택 확률을 적용하여 경로를 선택하는 방식을 제안한다.

제안된 다수의 QoS 라우팅 방식은 링크상태 전체 시계(link-state full-view)에 근거하였다. 이는 경로 계산, 통신 및 저장측면에서 대규모 네트워크에 그대로 적용하는데 어려움이 있다. 따라서 전통적으로 대규모 네트워크에 확장성이 있는(scalable) 라우팅 방식으로 계층적(hierarchical) 라우팅 방식이 사용된다[6, 7].

본 논문에서는 대규모 네트워크의 QoS 라우팅을 위해 계층적 시계에 근거한 응용의 QoS 요구를 만족하는 경로를 제안한 선계산 방식의 비용산정 방식에 적용하여  $K$  개의 다중 경로를 계산한다. 한편 계층적 라우팅은 송신 노드와 수신노드가 같은 도메인 내에 있을 경우에 적용할 도메인내 라우팅(intra-domain routing)은 물론 송신 노드와 수신 노드가 서로 다른 도메인 내에 있을 경우 여러 도메인을 지나가는 경로를 찾기 위한 도메인간 라우팅(inter-domain routing)도 고려하여야 한다[2]. 또한 대규모 네트워크에서 선계산 방식의 도메인간 QoS 라우팅에서는 특히 도메인의 ingress border router로부터 egress border router까지의 transit traffic에 대한 경로설정 및 자원예약의 성공이 중요하다. 이는 transit traffic을 위한 경로를 intra traffic을 위한 경로보다 더욱 빈번하게 요구하며, 따라서 이러한 경로를 중심으로 혼잡이 일어날 가능성이 높기 때문이다. 따라서 본 논문에서는 다음과 같은 방식으로 transit traffic 및 intra traffic을 위한 다중 경로 설정을 제안한다. 임의의 도메인에 대하여, 먼저 도메인의 모든 transit traffic을 위한  $K$  개의 경로를 그들 간의 부하분산을 고려하여 설정한다. 이제 transit traffic을 위한 경로에서 사용된 자원은 부하 분산을 위한 비용이 고려되었으므로 비용이 증가되었을 것이다. 따라서 이제 intra traffic을 위한 경로 설정을 할때, 고비용의 자원 사용을 회피하도록 유도함으로써 transit traffic을 위한 경로와 전부 혹은 일부가 가능한 중복되지 않도록 할수 있다. 결과적으로 transit traffic간의 경로 중복은 물론, transit

traffic과 intra traffic간의 경로 중복을 회피하도록 하여 자원예약 단계에서 transit traffic을 위한 자원예약이 성공할 가능성을 높일 수 있다. 이처럼 설정된  $K$  개의 경로들 중에서 제안한 경로선택확률에 따라 경로를 선택하도록 한다.

## 2. 네트워크 모델 및 정의

본 논문에서 고려하고 있는 계층적 네트워크는 그래프  $G(V, E)$ 로 표현되고, 노드들의 집합  $V$ 와 링크들의 집합  $E$ 로 구성되어 있다.  $N = |V|$ ,  $M = |E|$ 라 하자.

하나의 경로는 노드들의 유한 집합으로 다음과 같이 표현된다.

$$p = (v_0, v_1, \dots, v_h) \text{ for } 0 \leq n \leq h-1, (v_n, v_{n+1}) \in E$$

여기서  $h$ 는 경로  $p$  상에서 hop의 수를 나타내어,  $h = |p|$ 이다.

이러한 집합들에서 계층적 네트워크는 도메인이라는 영역으로 분할되어 있으며 도메인은 계층적으로 조직화되어 있다. 도메인은 복수 개수가 허용되는 링크에 의하여 서로 연결되어 있다. 도메인 계층구조는 다음과 같은 부모-자식 관계로 구성된다.

계층  $l(l > 1)$  도메인은 이를 포함하는 계층  $l-1$  도메인의 부모이다. 최고 계층의 도메인들은 부모를 갖지 않으며, 최저 계층의 도메인, 계층 1의 도메인은 자식을 갖지 않는다.  $D_i$ 을 계층  $l$  도메인이라 하자. 첨자  $i$ 는 계층  $l$ 의 많은 도메인들 중 특정 하나를 나타내기 위하여 사용된다.

다음과 같은 표현 및 연산을 제안하는 비용산정 방식에서 사용된다.  $D$ 를 네트워크에서의 도메인,  $b$ 를 노드라고 하자.  $\text{Domain}(b)$ 는 노드  $b$ 가 위치하는 도메인을 나타낸다.  $\text{Level}(D)$ 는 계층적 구조에서 도메인  $D$ 의 계층을 나타낸다. 따라서  $\text{Level}(D_i) = l$ 이 된다.  $\text{Ancestor}(b)$ 는 노드  $b$ 의 선조도메인,  $\text{Siblings}(D)$ 를 형제도메인이라 한다.

$\text{Nodes}(D)$ 는 도메인  $D$  내 노드들의 집합을 나타낸다.  $\text{Border}(D_i)$ 는  $l$  계층 도메인내 border router 들의 집합을 나타내며,  $\text{Intra}(D_i)$ 는 계층 도메인 내 border router들을 제외한 노드들의 집합을 나타낸다. 따라서  $\text{Intra}(D_i) = \text{Nodes}(D_i) - \text{Border}(D_i)$ 라 할 수 있다.

계층적 라우팅에서의 비용산정 방식은 네트워크 자원 중 링크만을 다룬다. 자원의 비용을 정의하기 위해 분리된 자원사용 비용과 경로설정 비용은 다음 절에 제시하는 계층적 라우팅에서의 비용산정 방식에 적용하여 전체 네트워크 자원의 최적사용 뿐만 아니라 부하분산이 고려된 경로 설정이 가능하도록 하였다.

### 3. 선계산 방식에서의 비용산정 방식 제안

#### 3.1 선계산 방식의 비용산정 방식

선계산 방식의 경로설정에서는 비교적 정적인 정보를 사용하여 경로설정 계산을 하게 된다[1, 4, 5]. 만일 그렇지 않다면, 동적인 정보의 변화에 따라 빈번하게 경로설정 계산을 하게 되며 선계산 방식을 사용하게 되는 장점을 잃게 된다. 만일 주어진 송신자/수신자간에 QoS 요구를 만족하는 경로를 하나만 설정한다면, 요청시 경로의 과부하로 자원예약이 불가능할 경우가 많다. 이를 보완하기 위하여  $K$  개의 경로를 미리 설정하고 요청시 선택하는 것이 바람직할수 있다[8]. 자원예약 측면에서 가장 좋은  $K$  개의 경로들은 경로의 전부 혹은 일부를 서로 공유하지 않는 경로이다[10, 11]. 주어진 송신자/수신자간의 주어진 QoS를 만족하는  $K$  개의 경로를 미리 설정할 경우, 비용을 최소화하는 경로 즉 전체 네트워크상의 자원 사용을 최소화 하는 경로를 설정하는 것이 일반적이었다[4]. 전체 네트워크 자원 사용을 위하여 간단하지만 효과적인 척도로 최소 홉 경로가 주로 고려되었다[4, 12]. 전체 네트워크 자원 사용은 전체 네트워크의 사용 효율을 위하여 고려되어야 하지만, 네트워크 일부 지역에서의 혼잡을 방지하기 위하여, 전체 네트워크에 걸쳐 각 자원의 부하가 분산되게 사용될 필요가 있다. 또한 한 송신자/수신자 쌍의 어떤 주어진 QoS 요구만을 대상으로 하는 부하 분산도 중요하지만, 그보다는 가능한 모든 송신자/수신자 쌍의 가능한 모든 QoS 요구에 대한 경로를 미리 설정할때의 부하 분산이 더욱 중요할 수 있다. 본 논문에서는 부하 분산을 위한 요소를 경로 선정을 위한 계산시의 비용에 포함하도록 하여, 실제로 요청의 자원 예약시 선택된 경로상의 자원예약의 가능성을 높이고자 한다. 이와 같은 방식으로 동적인 자원변화에 따른 혼잡을 정적인 비용으로 예측하여 고려하는 것이 어느 정도 가능할 수 있다. 다음 식 (1)은 본 논문에서 제안한 선계산 방식에서의 비용 산정을 위한 수식이다.

$$l_k(e) = aL_k(e)d(e) + (1-a)c(e) \quad (1)$$

여기서

- $l_k(e)$  ( $k = 1, 2, \dots, K$ ) : QoS라우팅에서  $k$ 번째 경로를 미리 설정하기 위하여 사용할 링크 비용
- $c(e)$  : 요청이 링크  $e$ 를 사용할 경우의 실제 비용으로 전체 네트워크 자원 사용을 고려하는 비용. 간단히 모든 링크에 대하여 같은 비용을 부과할 경우, 즉  $\forall e \in E, c(e) = \hat{c}$  경로상의 홉의 수를 비용으로 산정하게 된다. 여기서  $\hat{c}$  은 상수이다.

$d(e)$  : 링크  $e$ 가 네트워크에 미치는 영향의 정도, 즉 링크  $e$ 가 과부하되어 더 이상 사용되지 못하게 될 경우 네트워크에 미치는 피해(damage)의 정도. 이는 네트워크에서 각 링크의 역할의 다양성을 고려하기 위한 것으로, 만일 고려할 필요가 없다면, 모든 링크에 대하여 같은 비용을 부과해도 된다. 즉  $\forall e \in E, d(e) = \hat{d}$  여기서  $\hat{d}$ 은 상수이다.

$a$  : 부하 분산 요소(load balancing factor),  $0 < a \leq 1$

$l_k(e)$ 에  $c(e)$ 와  $d(e)$ 를 어떤 비율로 반영할 것인지를 결정하는 요소를 의미한다.

if  $a = 0$ , 즉  $l_k(e) = c(e)$  부하분산된 분배는 고려치 않고, 전체자원 사용만을 고려하여 경로계산

if  $a = 1$ , 즉  $l_k(e) = L_k(e)d(e)$  부하분산된 분배만을 고려하고, 전체자원 사용은 고려치 않고 경로계산

$L_k(e)$  :  $k$  번째 경로설정을 위하여 사용할 링크  $e$ 의 예측 부하율로 다음과 같이 정의된다. ( $0 < L_k(e) \leq 1$ )

링크  $e$ 가 어떤 경로상에서도 사용되지 않는다면,  $L_k(e) = 0$ 이 된다.

링크  $e$ 가 여러 경로상에서 충분히 사용되어, 실제로 혼잡이 일어날 것으로 예측된다면,  $L_k(e) = 1$ 이 된다.

예를 들어,  $L_k(e)$ 는 다음과 같이 정할 수 있다.

$$L_0(e) = 0 \quad : \text{초기}$$

$$L_k(e) = K_{k-1}(e) \quad : \text{만약 링크 } e \text{가 } k \text{번째 경로에 사용되지 않는다면}$$

$$L_k(e) = L_{k-1}(e) + Pr(k) \quad : \text{만약 링크 } e \text{가 } k \text{번째 경로에 사용될 경우}$$

$$= L_{k-1}(e) + \delta^k / \Delta_{sum}$$

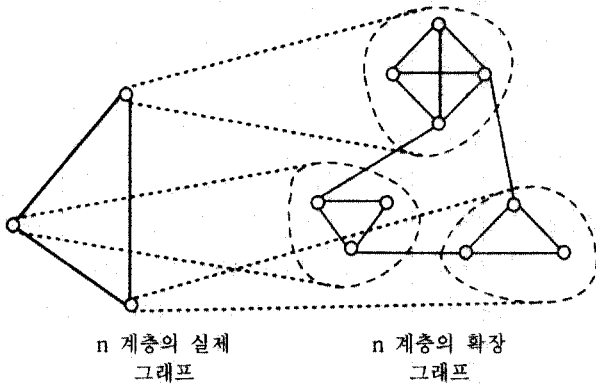
$$\text{여기서, } \Delta_{sum} = \sum_{i=1}^K \delta^i \quad (0 < \delta < 1)$$

$\delta$  값의 사용과  $L_k(e), Pr(k)$ 에 대한 구체적인 설명, 그리고  $\Delta_{sum} = \sum_{i=1}^K \delta^i$ 의 유도과정은 4.2절에 제시하였다.

선계산 방식에서의 비용산정 방식에대한 시간 복잡도는  $O(KM)$ 이다. 여기서  $M = |E|$ 이고  $E$ 는 링크들의 집합을 나타내며,  $K$ 는 다중 경로의 개수를 의미한다. 기존의 비용산정은 하나의 경로를  $O(M)$ 으로 계산하나, 제안한 비용산정은  $K$ 개의 경로를  $O(KM)$ 으로 계산하므로 동일한 시간 복잡도를 갖는다. 따라서 제안한 비용산정 방식은 홉수를 비용으로 하는 기존 라우팅의 비용산정과 같은  $O(M)$ 의 시간 복잡도를 갖으나 실제 계산시간은 상수 차이 정도로 더 소요된다.

3.2 계층적 라우팅에서 확장그래프를 적용한 비용산정

계층적 라우팅을 위해 n계층의 각 노드 쌍간의 비용을 계산한 후 이들 비용을 근거로 n계층의 도메인은 (n+1)계층의 노드로 변환한다. 이를 위해 n계층의 각 노드 쌍간의 비용들 중에서 도메인내 라우터들간의 링크 비용들은 (n+1)계층의 노드 비용이 된다. 이처럼 n계층의 도메인이 (n+1)계층의 노드로 집약됨에 따른 (n+1)계층의 비용산정의 어려움을 해결하기 위해 본 논문에서는 (그림 1)과 같이 확장 그래프를 생성한 후 비용산정을 수행하는 방안을 제안하였다.



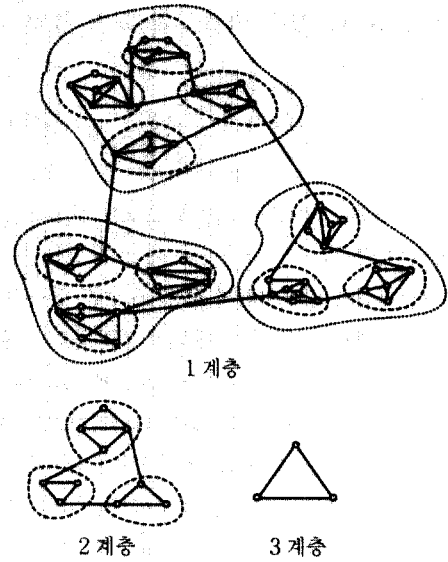
(그림 1) 확장그래프 예제

(n+1)계층의 비용 산정을 위해 (n+1)계층의 도메인은 (n+1)계층의 확장그래프로 구성되도록 한 후, 구성된 확장그래프 상의 각 노드 쌍간의 비용을 계산하도록 한다. (n+1)계층의 확장 그래프는 n계층의 도메인내 선택된 노드들, 즉 선택된 border router들과 (n+1)계층의 물리적 링크(physical link), 그리고 (n+1)계층의 논리적 링크(logical link)로 구성된다. (n+1)계층의 물리적 링크는 (n+1)계층의 실제적인 링크로서 n계층의 각 도메인간 링크를 의미한다. 그리고 (n+1)계층의 논리적 링크는 n계층의 선택된 border router간의 연결로 구성된 링크로서 (n+1)계층의 실제 그래프에서는 노드 내에 내포된다.

3.3 계층적 라우팅에서 선계산 방식의 비용산정

계층적 라우팅에서 선계산 방식은 (그림 2)와 같이 계층적으로 분류된 각 계층에서 각 노드 쌍간의 자원 예약 측면에서 가장 좋은 K개의 경로들을 미리 계산하고 요청시 경로를 선택하도록 한다.

이를 위해 각 계층별로 경로를 계산할때 해당 계층의 확장 그래프를 생성하고 여기서 각 노드 쌍간의 가장 좋은 K개의 경로 비용을 계산하도록 한다. 따라서 최상위 계층까지 각 계층별 확장 그래프를 생성하고 그래프상의 각 노드 쌍간의 비용을 계산한다. 이를 알고리즘으로 표현하면 다음과 같다.



(그림 2) 계층적 도메인 예제

```

Algorithm Precomputation-HIE (G, s, t)
/* 각 계층별 확장 그래프를 생성한 후 각 노드 쌍간의 K 다중
  경로설정 비용 계산 */
{
   $l_k(e) = 0$ , for  $\forall e \in E$ 
  for  $k = 1$  to  $K$  do
  {
    for  $h = 2$  to  $N-1$  do
    {
      /* construct the extended graph of layer  $h$ ,  $\hat{D}_h$ , */
      for all domains  $D_{h-1}$ , do
        for each  $b_x, b_y \in \text{Border}(D_{h-1})$  do
        {
          /* define all nodes in the extended graph of
            layer  $h$ ,  $\hat{D}_h$ , */
           $\hat{b}_x \leftarrow b_x$ ;
           $\hat{b}_y \leftarrow b_y$ ;
          /* compute logical link cost of the extended
            graph of layer  $h$ ,  $\hat{D}_h$ , */
          computes  $l_k(e_{\hat{b}_x, \hat{b}_y}^{k-1})$ ;
        }
      /* assign 0 to layer  $h$ 's physical link cost of all
        domains */
      for all pairs of domains  $D_{h-1}, D_h$ , do
        for each  $b_x \in \text{Border}(D_{h-1}), b_y \in \text{Border}(D_h)$  do
           $l_k(e_{b_x, b_y}^k) = 0$ ;
      /* compute all link cost of the extended graph of
        layer  $h$ ,  $\hat{D}_h$ , */
      for all domains  $\hat{D}_h$ , do
        for each  $\hat{b}_x, \hat{b}_y \in \text{Nodes}(\hat{D}_h)$  do
          compute  $l_k(e_{\hat{b}_x, \hat{b}_y}^k)$ ;
    }
  }
}
  
```

(그림 3) 계층적 도메인에서의 선계산 방식의 비용계산 알고리즘

### 4. 부하분산 QoS 라우팅

#### 4.1 transit traffic 및 intra traffic을 위한 QoS 라우팅

도메인간 라우팅에서는 여러 도메인들을 거치며 경로가 설정되게 되는 경우가 빈번하다. 이러한 경우 처음과 마지막 도메인에서는 intra traffic을 위하여 설정된 경로를 이용하며 중간 도메인들에서는 transit traffic을 위하여 설정된 경로를 이용하게 된다. 설정된 경로상의 자원예약 단계에서 중간 도메인에서의 transit traffic을 위한 자원예약이 실패할 경우가 더욱 심각한 문제가 될 수 있다.

본 논문에서는 3절에서 제시한 비용산정 방식을 이용하여 다음과 같은 순서로 transit traffic 및 intra traffic의 다중 경로설정을 제안한다. 먼저 도메인의 모든 incoming border router로부터 outgoing border router간의 transit traffic을 위한 K개의 경로를 설정하며 이때 경로들간의 부하분산을 이루도록 한다. 이제 transit traffic을 위한 경로에서 사용된 자원은 부하분산을 위한 비용이 고려되었으므로 비용이 증가되었을 것이다. 따라서 이제 intra traffic을 위한 경로 설정을 할때, 고 비용의 자원 사용을 회피하도록 유도함으로써 transit traffic을 위한 경로와 전부 혹은 일부가 가능한 한 중복되지 않도록 한다. 다음 (그림 4)는 K 다중 경로계산을 위한 제안된 비용산정 방식과 transit traffic 우선 정책을 적용한 선계산 방식 알고리즘이다.

```

Algorithm Precomputation(D)
/* 주어진 도메인 D에 대하여 border router 쌍간에 먼저 K다중 경로 계산 후에 intermediate router 쌍간에 K다중 경로 계산 */
{
     $l_0(e) = 0, \text{ for } \forall e \in E$ 
    for  $k = 1$  to  $K$  do
        for each pair of border routers in decreasing order of significance do
            {
                computes  $l_k(e), \text{ for } \forall e \in E$ 
                computes the  $k$ -th best path for the pair by using a path computation algorithm
            }
        for  $k = 1$  to  $K$  do
            for each pair of intermediate routers do
                {
                    computes  $l_k(e), \text{ for } \forall e \in E$ 
                    computes the  $k$ -th best path for the pair by using a path computation algorithm
                }
    }

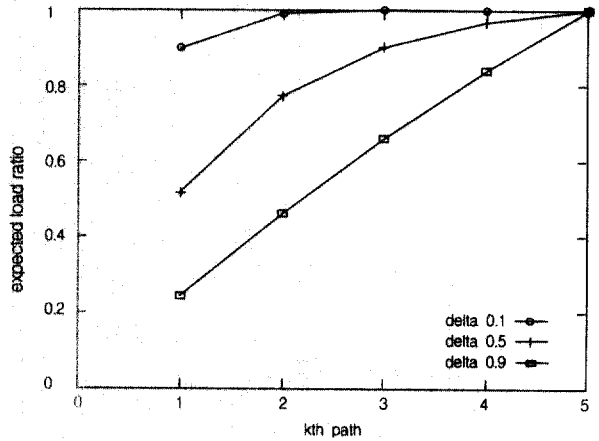
```

(그림 4) 제안한 transit traffic우선의 선계산 방식 알고리즘

#### 4.2 선계산 방식을 이용한 경로 선택

제안한 선계산 방식에서의 경로선택은 선계산 방식으로 결정된 K 다중 경로중에서 하나의 경로를 선택해야 한다. 이에 본 논문에서는 선계산 방식에서 사용된  $L_k(e)$ 와 경로 선택 확률  $pr(k)$ 를 이용하여 경로 선택하는 방식을 제안한다. (그림 5)에서는 링크 e의 예측부하율  $L_k(e)$ 를  $K=5$ 에

대해 표현하였다.  $\delta$ 의 값이 1에 가까울수록 링크 e의 예측 부하율  $L_k(e)$ 가 k의 증가( $1 \leq k \leq K$ )에 따라 고르게 증가하는 것을 알 수 있다. 반면  $\delta$ 의 값이 0에 접근할수록, 급격히 증가하는 것을 알 수 있다. 이것은 실제 경로 선택과정에서도 이러한 증가와 비례하여 경로를 선택하여야 함을 의미하게 된다. 이와 같이 어떠한  $\delta$ 의 값을 선택하여야 하는가 하는 문제는 실제 고려중인 네트워크의 특성을 반영하여 경로 선택과정과 연계하여야 한다.



(그림 5) K=5에서의 예측부하율( $L_k(e)$ )

실제 경로요구시 어떤 경로를 어떤 확률로 선택하여야 하는가는 비용산정시 사용된 값에 근거하게 된다. 이들 경로들의 선택은 다음의 k에 따른 확률을 통해 이루어진다.

$pr(k)$ 를 k번째 경로가 선택될 확률이라 할때, 앞의 식 (1)에서 다음을 유추할 수 있다.

$$pr(k) = \delta^k / \Delta_{sum} \quad \text{여기서} \quad \Delta_{sum} = \sum_{i=1}^K \delta^i$$

그러면 다음의 2가지 관계가 성립된다.

관계 1:  $pr(1^{st}) \geq pr(2^{nd}) \geq \dots \geq pr(K) > 0$

관계 2:  $\sum_{k=1}^K pr(k) = 1$

여기서  $pr(k) = \delta^k / \Delta_{sum}$ 이다. 이의 유도 과정은 다음과 같다.

<표 1> 경로 선택 확률  $pr(k)$ 의 경로별 누적

path	1 <sup>st</sup> path	...	k <sup>th</sup> path	...	K <sup>th</sup> path
	$\delta^1 \cdot 1$	...	$\delta^k \cdot 1$	...	$\delta^K \cdot 1$
	$\delta^1 \cdot q$	...	$\delta^k \cdot q$	...	$\delta^K \cdot q$
	$\delta^1 \cdot q^2$	...	$\delta^k \cdot q^2$	...	$\delta^K \cdot q^2$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$pr(k)$	$\delta^1(1 + q + q^2 + \dots)$	...	$\delta^k(1 + q + q^2 + \dots)$	...	$\delta^K(1 + q + q^2 + \dots)$

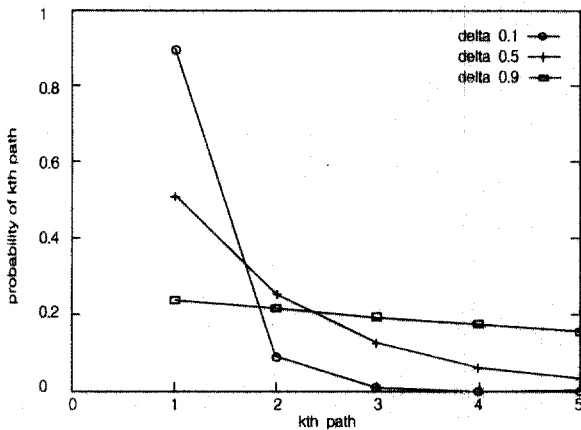
즉,  $k$  번째 경로가 선택될 확률인  $pr(k)$ 는 다음과 같다.

$$pr(k) = \delta^k (1 + q + q^2 + q^3 + \dots)$$

$$= \delta^k \left( \frac{1}{1-q} \right) = \frac{\delta^k}{\sum_{i=1}^K \delta^i} = \frac{\delta^k}{\Delta_{sum}}$$

여기서  $\Delta_{sum} = \sum_{i=1}^K \delta^i$  이다.

(그림 6)에서는 경로 선택 확률  $pr(k)$ 를  $K=5$ 에 대해 표현하였다.



(그림 6)  $K=5$ 에서의  $k^{\text{th}}$  경로들의 경로선택 확률

(그림 6)에 나타난 바와 같이  $\delta$ 의 값이 1에 접근할수록  $k$ 의 증가에 따라 경로선택 확률  $pr(k)$ 가 고르게 감소한다. 따라서  $K$  다중경로를 비슷한 확률로 고르게 선택할 수 있다는 것을 알 수 있다. 반면 0에 접근할수록 선택의 폭이 줄어들게 된다.

다음 (그림 7)은  $K$  다중경로중 하나의 경로를 확률  $pr(k)$ 에 따라 선택하는 경로선택 알고리즘이다.

```

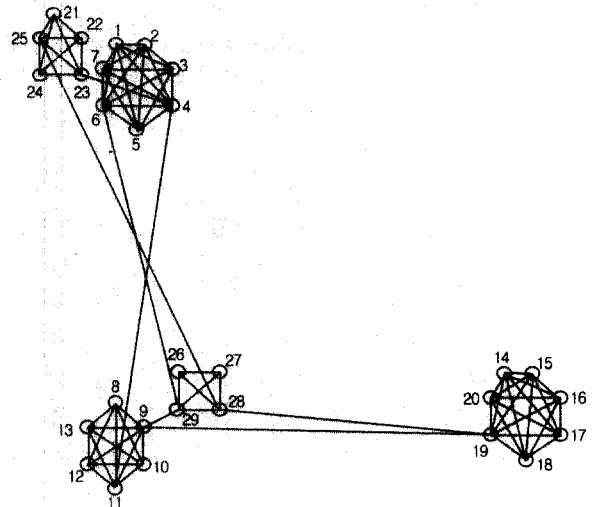
Algorithm PathSelection (D, s, t)
/* 주어진 도메인 D의 송신자 s, 수신자 t에 대한 K 다중경로 중
  하나의 경로를 선택 */
{
     $\Delta_{sum} = 0$ 
    for k = 1 to K do
         $\Delta_{sum} += \delta^k$ 
    for k = 1 to K do
         $pr(k) = \delta^k / \Delta_{sum}$ 
    selects the k-th path with the probability  $pr(k)$ 
    among K precomputed paths from s to t
}
    
```

(그림 7) 제안된 경로선택 알고리즘

### 5. 시뮬레이션 및 성능분석

본 논문에서 제안한 계층적 도메인에서의 부하분산 QoS

라우팅을 분석하기 위하여 시뮬레이션을 수행하였다. 1계층의 계산된 비용을 근거로 2계층을 구성하기 위해 (그림 9)와 같이 Waxman의 그래프 모델[9]을 토대로 2계층의 확장 그래프(extended graph)를 구성하였다.



(그림 9) 계층적 라우팅을 위한 그래프 모델(2계층의 확장 그래프)

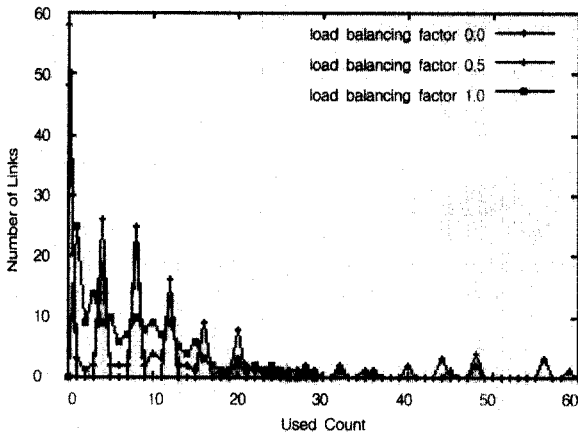
시뮬레이션은 범용 시뮬레이션 언어인 SIMSCRIPT II.5을 사용하였다. 부하분산과 전체 네트워크 자원 사용을 검증하기 위해 노드 쌍간 발생할 수 있는 최대연결개수의 약 18%에 해당하는 150개 노드 쌍에 대해서 실험하였다. 시뮬레이션을 간단히 하기 위해 네트워크내에 모든 링크는 동일한 대역폭을 갖고 있고, 각 연결은 동일한 대역폭을 요구한다고 가정하고  $c(e)$ 와  $d(e)$ 의 값을 각각 1로,  $\delta$ 의 값을 0.25로 가정하였다.

먼저 네트워크 자원의 부하분산된 사용은 가정된 모든 노드 쌍의 연결에서 각 링크의 사용횟수(used count)를 측정한 후, 측정된 각 사용횟수를 갖는 링크들의 개수(number of links)에서 찾도록 하였다. 사용횟수가 많은 링크들의 개수가 적을수록 혼잡이 발생할 가능성이 낮게되어 네트워크 자원의 부하 분산된 사용의 결과를 얻을 수 있게 된다. 이와 반대로 사용횟수가 많은 링크의 개수가 많을수록 혼잡의 발생확률은 높아지고 부하분산된 사용의 결과를 얻기 어렵게 된다.

시뮬레이션은 부하분산요소( $\alpha$ ) 값에 따른 부하분산 정도를 나타내기 위하여 0에서 1까지의 값들을 적용하였다. 본 논문에서 제안한 계층적 라우팅에서의 부하분산 비용산정방식을 적용하기 위하여 2계층의 확장 그래프상의 각 노드 쌍의 연결에 대하여 Dijkstra 알고리즘에 의한 경로설정 후 식 (1)을 적용하여 경로상의 모든 링크의 비용을 증가시켰다.

(그림 10)과 <표 2>와 같이  $\alpha = 0$ 에서 사용횟수가 높은

링크들의 개수가 많이 있음을 발견할 수 있었다. 사용횟수가 많다는 것은 부하분산의 효과는 떨어지고 혼잡의 발생확률이 높다고 판단할 수 있다. 또한  $\alpha = 1$ 에 접근할수록 사용횟수가 많은 링크수가 적어지게 되었다. 이는 혼잡이 덜 발생함을 의미하고 네트워크 전체에 걸쳐 부하가 분산된 경로 설정이 되었다고 판단할 수 있다.



(그림 10) 계층적 도메인에서의 부하분산의 시뮬레이션 결과

<표 2> 계층적 도메인에서 최대 사용 개수

$\alpha$	1st 최대사용개수	2nd 최대사용개수	3rd 최대사용개수
0.0	59	56	48
0.25	52	47	40
0.5	48	45	35
0.75	33	31	30
1.0	28	25	24

한편, 계층적 라우팅에서 전체 네트워크 자원의 사용은 가해진 모든 노드쌍의 연결에서 각 링크의 사용횟수를 측정 후, 이들을 누적하여 전체 사용횟수를 구하였다. 전체 사용횟수가 적을수록 네트워크 자원을 적게 사용하는 것의 의미한다.

<표 3> 계층적 도메인에서 네트워크 자원 사용에 대한 결과

$\alpha$	0	0.25	0.5	0.75	1
전체사용횟수	1635	1644	1687	1735	1827

<표 3>과 같이  $\alpha = 0$ 에서 최소 링크수로 가장 적은 자원을 사용함을 알 수 있고,  $\alpha = 1$ 에서 최대 링크수로 가장 많은 자원을 사용함을 알 수 있다.

이와 같은 시뮬레이션에서 나타난것과 같이 부하분산은  $\alpha = 1$ 에서 최적의 결과를 찾을 수 있었고, 네트워크 자원 사용은  $\alpha = 0$ 에서 최적의 결과를 찾을 수 있었다. 따라서 본 논문에서 제안한 계층적 라우팅에서의 비용산정방식은 부하분산과 전체 네트워크 자원 사용의 목적에 따라 부하

분산요소의 값을 변화하면서 네트워크 특성에 맞추어 적용할 수 있다

## 6. 결 론

본 논문에서는 계층적 라우팅에서의 부하분산 QoS 라우팅을 제안하였다. 부하분산 QoS 라우팅을 위해 자원의 비용산정을 선계산 방식의 QoS 라우팅 경로 계산에서 제안하였고, 선계산 방식으로 계산된 경로의 자원예약 가능성을 높이기 위해 임의의 도메인내 모든 transit traffic을 위한  $K$ 개의 경로를 그들 간의 부하분산을 고려하여 설정하는 방안을 제안하였다. 이들 경로들은 자원예약 측면에서 가장 좋은  $K$ 개의 경로들로서, 경로의 전부 혹은 일부를 서로 공유하지 않는 경로들이다. 부하분산을 위한 요소를 경로 설정을 위한 계산시의 비용에 포함하도록 하여, 실제로 요청의 자원 예약시 선택된 경로상의 자원예약의 가능성을 높일 수 있도록 하였다. 또한 계층적 도메인간 라우팅에서 transit traffic의 자원 예약을 intra traffic에 우선하는 방식을 제안하여 도메인간 라우팅과 도메인내 라우팅을 무리 없이 연계할 수 있도록 하였다. 또한 제안된 비용산정 방식에 따라 계산된  $K$  다중 경로들중에서 제안한 경로선택확률에 따라 경로를 선택하는 방식을 제안하였다. 따라서 전체 네트워크 자원의 최적 사용과 부하분산을 위한 효율적 해결방안을 제시할 수 있게 되었다.

본 논문에서 제안한 계층적 라우팅에서의 부하분산 QoS 라우팅의 분석을 위하여 시뮬레이션을 수행한결과 부하분산은 부하분산요소  $\alpha = 1$ 에서 최적의 결과를 찾을 수 있었고, 네트워크 자원 사용은  $\alpha = 0$ 에서 최적의 결과를 찾을 수 있었다. 따라서 본 논문에서 제안한 비용산정방식은 네트워크 자원의 부하분산과 전체 네트워크 자원 사용 목적에 따라 부하분산요소의 값을 변화하면서 네트워크 특성에 맞추어 적용할 수 있다.

## 참 고 문 헌

- [1] 홍종준, 김승훈, 이균하, "QoS 라우팅을 위한 부하균등 비용산정 방식", 정보처리학회논문지C, 제9-C권, 제1호, pp.135-140, 2002.
- [2] 김승훈, 김치하, "분산멀티미디어 응용을 위한 대규모 고속통신망에서의 QoS-근거 계층적 도메인간 라우팅 방식", 한국통신학회논문지, 제24권 제7호, 1999.
- [3] Seung-Hoon Kim, Kyungshik Lim and Cheeha Kim, "A Scalable QoS-based Inter-Domain Routing Scheme in a High Speed Wide Area Network," *Computer Communica-*

tions, Vol.21, No.4, pp.390-399, 1998.

[4] Ariel Orda and Alexander Sprintson, "QoS Routing : The Precomputation Perspective," *Proceedings IEEE INFOCOM '2000*, Mar., 2000.

[5] A. Shaikh, J. Rexford, and K. Shin, "Efficient Precomputation of Quality-of-Service Routes," *Proc. Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV '98)*, Jul., 1998.

[6] C. Alaettinoglu and A. U. Shanker, "The Viewserver Hierarchy for Interdomain Routing : Protocol and Evaluation," *IEEE J Select. Areas Comm.*, Vol.13, No.8, pp.1396-1410, Oct., 1995.

[7] "Private Network-Network Interface Specification Version 1.0 (PNNI 1.0)," *ATM Forum PNNI Subworking Group*, af-pnni-0055.000, Mar., 1996.

[8] Y. L. Chen, "An Algorithm for Finding the  $k$  Quickest Paths in a Network," *Computers & Operations Research*, Vol.20, No.1, pp.59-65, 1993.

[9] B. M. Waxman, "Routing of Multipoint Connections," *IEEE J. Select. Areas Commun.*, Vol.6, No.9, pp.1617-1622, Dec., 1988.

[10] Murali Kodialam, T. V. Lakshman, "Minimum Interference Routing with Applications to MPLS Traffic Engineering," *Proceedings of IEEE INFOCOM '2000*, Mar., 2000.

[11] Hiroyuki Saito, Yasuhiro Miyao, and Makiko Yoshida, "Minimum Interference Routing with Applications to MPLS Traffic Engineering," *Proceedings of IEEE INFOCOM '2000*, Mar., 2000.

[12] A. Orda and A. Sprintson, "A Scalable Approach to the Partition of QoS Requirements in Unicast and Multicast," *Proceedings of IEEE INFOCOM '02*, Jun., 2002.

### 홍종준



e-mail : jjhong@mail.chungkang.ac.kr

1991년 인하대학교 전자계산공학과 졸업  
1993년 인하대학교 대학원 전자계산공학과  
(공학석사)

1997년~현재 인하대학교 대학원 전자계산공학과 박사과정 수료

1993년~1998년 LG산전 연구소 주임연구원

1999년~현재 청강문화산업대학 컴퓨터소프트웨어과 조교수

관심분야 : 초고속통신망, 분산시스템, 알고리즘

### 김승훈



e-mail : edina@anseo.dankook.ac.kr

1985년 인하대학교 전자계산학과 졸업  
1989년 인하대학교 대학원 전자계산학과  
(공학석사)

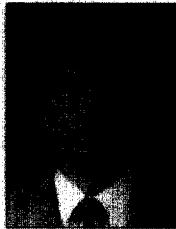
1998년 포항공과대학 전자계산학과(공학박사)

1998년~2001년 상지대학교 전자계산공학과 조교수

2001년~현재 단국대학교 전자·컴퓨터학부 조교수

관심분야 : 컴퓨터네트워크, 초고속통신망, 알고리즘

### 이균하



e-mail : khlee@inha.ac.kr

1970년 인하대학교 전기공학과 졸업  
1976년 인하대학교 전자공학과 석사  
1981년 인하대학교 전자공학과 박사

1977년~1981년 광운대학교 교수

1981년~현재 인하대학교 컴퓨터공학부 교수

관심분야 : 지능통신망, 패턴인식