

QoS 경로의 선계산을 적용한 부하분산 QoS 라우팅

홍종준[†]

요약

본 논문에서는 대규모 네트워크에서의 부하분산을 고려한 QoS 라우팅을 제안한다. 이를 위해 경로설정을 위한 비용 산정 방식을 선계산 방식의 경로계산에서 제안하고 선계산된 다중경로들 중에서 경로를 선택하는 방안을 제안한다. 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 선택한다. 그리고 제안한 QoS 라우팅 방식을 transit traffic과 intra traffic에 모두 적용하여 도메인간 라우팅과 도메인내 라우팅을 무리 없이 연계하면서 transit traffic을 우선으로 하는 방식을 제안한다. 따라서 본 논문에서 제안한 부하분산 QoS 라우팅은 전체 네트워크 자원의 최적사용 뿐만 아니라 자원의 부하 분산을 가능하게 할 것으로 기대된다.

Load Balanced QoS Routing for Precomputation of QoS Routes

Jong-Joon Hong[†]

ABSTRACT

In this paper, we propose a load balancing QoS routing on a large scale network. We use the scheme in which it calculates the cost of resources with a precomputation and then selects a path among multiple paths based on the calculated cost for each resources. We also use the QoS routing scheme for the transit and the intra-traffic on a large scale domain-based network. For a domain in networks, this routing scheme precomputes K multiple paths between all pairs made up with each ingress and egress border routers along with the consideration of a load balancing, and then picks up paths having the high probability on the path selection. This routing scheme can be applied for the transit traffic and the intra traffic and combines the inter-domain and intra-domain routings seamlessly and put a high priority on the transit traffic. Thus, these schemes make possible both the global use of network resources and the load balancing.

Key words: QoS라우팅, 선계산, 부하분산

1. 서론

효율적인 QoS 라우팅은 각각의 요구에 대한 대역폭, 중단간 지연 등의 제한조건(constraints) 충족은 물론, 전체 네트워크 자원의 최적화에 대한 고려가 필요하다[1-4]. 이는 적은 자원으로 주어진 QoS 요구를 만족시키는 경로를 설정한다면, 이후의 다른 요청에 대한 경로 설정 가능성이 더욱 높아지기 때문이

다. QoS라우팅을 위한 선계산 방식[1,5]에서는 계산 및 통신의 부하를 줄이기 위하여 비교적 정적인 정보를 사용하여 경로 설정 계산을 하여, 동적인 네트워크 자원 변화를 경로설정 계산 단계에서 반영하기 어렵다. 또한 QoS요구를 만족하는 경로를 하나만을 설정한다면 항상 이 경로만을 사용하므로 과부하의 가능성이 많아지고 자원예약이 불가능할 경우가 많다. 따라서 이를 보완하기 위해 주어진 QoS를 만족하는 K 개의 경로를 미리 설정하고 요청 시 이들 중에서 선택하는 방식이 사용된다[1,6]. 자원예약 측면에서 가장 좋은 K 개의 경로들은 경로의 전부 혹은 일

부를 서로 공유하지 않는 경로들이다[7,8]. 경로 설정은 전체 네트워크 자원의 최적화 뿐만 아니라 혼잡(congestion)을 방지하기 위한 네트워크 자원의 분산된 사용을 고려해야 한다. 네트워크의 어느 지역에 경로가 집중적으로 선택되었다면, 그 지역이 실제로 항상 혼잡이 발생한다는 것을 의미하는 것은 아니지만, 자원 예약의 실제 가능성이 다른 지역의 경로에 비하여 비교적 높다고 판단할 수 있다. 이에 본 논문에서는 가능한 모든 QoS요구에 대한 경로 설정을 미리 계산할 때, 네트워크의 자원에 대한 부하분산을 고려하기 위한 비용 산정 방식을 제안하였다. 즉, 부하분산을 위한 요소들을 경로 선정 계산 시의 비용에 포함하도록 하여 실제 자원 예약 시 선택된 경로상의 자원 예약의 가능성을 높이도록 한다. 또한 계산된 경로들 중에서 경로선택은 경로설정 요구시 선정 방식으로 결정된 K 다중 경로 중에서 하나의 경로를 선택해야 한다. 이에 본 논문에서는 선계산 방식에서의 비용산정방식에서 사용된 예측부하율과 경로선택 확률을 이용하여 경로를 선택하는 방식을 제안한다.

대규모 네트워크에서는 정책적인 이유 및 확장성 문제로 인하여 네트워크를 도메인으로 분할하는 것이 일반적이다. 선계산 방식의 도메인간 QoS 라우팅에서는 특히 도메인의 ingress border router로부터 egress border router까지의 transit traffic에 대한 경로 설정 및 자원예약의 성공이 중요하다. 설정된 경로상의 자원예약 단계에서 중간 도메인에서의 transit traffic을 위한 자원예약이 실패할 경우가 송신자 혹은 수신자 도메인에서의 intra traffic에서의 자원예약 실패보다 더욱 심각한 문제가 된다. 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. 결과적으로 transit traffic간의 경로 중복은 물론, transit traffic과 intra traffic 간의 경로 중복을 회피하도록 하여 자원예약 단계에서 transit traffic을 위한 자원예약이 성공할 가능성을 높일 수 있다. 이처럼 설정된 K 개의 경로들 중에서 제안한 경로선택 확률에 따라 경로를 선택하도록 한다.

2. 네트워크 모델 및 정의

본 논문에서 대상으로 하는 네트워크에 관한 모델

의 정의에서 시작하여 다음 절에서 제시할 비용산정을 위한 각각의 요소에 대한 정의가 제시된다. 네트워크는 그래프 $G(V, E)$ 로 표현된다. 여기서 V 는 노드들의 집합을 의미하고, E 는 링크들의 집합을 나타낸다. $N = |V|$, $M = |E|$ 라 하자.

하나의 경로는 노드들의 유한 집합으로 다음과 같이 표현된다.

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

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

주어진 QoS요구에 맞는 경로, 즉 모든 제한조건을 만족하는 경로를 가용 경로(feasible path)라 한다. 그러나 주어진 제한조건에 대하여 최소가중치 경로가 전체 네트워크 사용 측면에서 반드시 좋은 선택일 수 없다. 선정된 경로상의 hop의 수가 많을 경우 많은 네트워크 자원을 소모하기 때문이다 [5]. 따라서 가용 경로들 중에서 자원 사용에 따른 비용을 최소화하는 경로를 최소 비용 경로라 할 수 있다.

본 논문에서는 네트워크 자원 중에서 링크만을 다루며, 링크 비용을 정의하기 위해 자원사용 비용과 경로설정 비용으로 분리하여 최소비용 경로를 계산한다. 이러한 자원사용 비용과 경로설정 비용을 다음 절에서 제시하는 경로 비용산정 방식에 적용하여 최소 비용 경로를 설정할 수 있도록 하였다.

3. 선계산 방식의 비용산정 방식 제안

선계산 방식의 경로설정에서는 비교적 정적인 정보를 사용하여 경로설정 계산을 하게 되며, 그렇지 않다면, 동적인 정보의 변화에 따라 빈번하게 경로설정 계산을 하게 되며 선계산 방식을 사용하게 되는 장점을 잃게 된다[5]. 따라서 동적인 네트워크 정보가 반영되지 못하므로, 이미 설정된 경로를 따라 자원을 예약하고자 할 때 자원예약이 실패될 가능성이 더 높게 된다. 만일 주어진 송신자/수신자간에 QoS 요구를 만족하는 경로를 하나만을 설정한다면, 이 경로에 과부하가 발생하여 자원예약이 불가능할 경우가 많다. 이를 보완하기 위하여 단순히 하나의 경로만을 설정하기 보다는 자원예약 측면에서 가장 좋고, 경로의 전부 혹은 일부를 서로 공유하지 않는 K 개의

경로를 미리 설정하고 요청 시 선택하는 것이 바람직하다[6,7].

주어진 송신자/수신자간의 주어진 QoS를 만족하는 K 개의 경로를 미리 설정할 경우, 비용을 최소화 하는 경로 즉 전체 네트워크 상의 자원 사용을 최소화 하는 경로를 설정하는 것이 일반적이었다. 전체 네트워크자원의 최적화는 전체 네트워크의 사용 효율을 위하여 고려되어야 하지만, 네트워크 일부 지역에서의 혼잡을 방지하기 위하여, 전체 네트워크에 걸쳐 각 자원의 부하가 분산되게 사용될 필요가 있다. 물론 어느 지역에 집중적으로 경로를 미리 설정한다는 것이 그 지역에 실제로 항상 혼잡이 일어난다는 것을 의미하지는 않지만, 자원예약이 실패할 가능성이 비교적 높다고 할 수 있다. 또한 한 송신자/수신자쌍의 어떤 주어진 QoS 요구만을 대상으로 하는 부하분산도 중요하지만, 그보다는 가능한 모든 QoS 요구에 대한 경로를 미리 설정할 때의 부하분산이 더욱 중요하다. 본 논문에서는 부하분산을 위한 요소를 경로 선정을 위한 계산시의 비용에 포함하도록 하여, 실제로 요청의 자원 예약 시 선택된 경로상의 자원예약의 가능성을 높이고자 한다. 다음은 선계산 방식에서의 비용 산정을 나타낸다.

$$l_k(e) = \alpha L_k(e)d(e) + (1-\alpha)c(e) \quad (1)$$

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 자원사용을 고려하는 비용.

간단히 모든 링크에 대하여 같은 비용을 부과할 경우, 즉 $\forall e \in E, c(e) = \hat{c}$, 경로상의

hop의 수를 비용으로 산정하게 된다. 여기서 \hat{c} 는 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} 는 상수이다.

α : 부하분산요소(load balancing factor), $0 \leq \alpha \leq 1$

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

if $\alpha = 0$, 즉 $l_k(e) = c(e)$

부하분산 분배는 고려치 않고, 전체 자원사용만을 고려하여 경로선택

if $\alpha = 1$, 즉 $l_k(e) = L_k(e)d(e)$

부하분산 분배만을 고려하고, 전체 자원사용은 고려치 않고 경로선택

$L_k(e)$: k 번째 경로설정을 위하여 사용할 링크 e 의 예측 부하율로 다음과 같이 정의된다.

$$(0 \leq L_k(e) \leq 1)$$

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.

$$L_0(e) = 0 \quad : \text{초기}$$

$L_k(e) = L_{k-1}(e)$: 만약 링크 e 가 k 번째 경로에 사용되지 않는다면

$L_k(e) = L_{k-1}(e) + \delta^k / \Delta_{sum}$: 만약 링크 e 가 k 번째 경로에 사용될 경우

$$\Delta_{sum} = \sum_{l=1}^K \delta^l \quad (0 < \delta < 1)$$

여기서,

δ 값의 사용과 $L_k(e)$, $Pr(k)$ 에 대한 구체적인 설명은 4.2절에 제시하였다.

선계산 방식에서의 비용산정 방식에 대한 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 함수를 비용으로 하는 기존 라우팅의 비용산정과 같은 $O(M)$ 의 시간 복잡

도를 갖으나 실제 계산시간은 상수 차이 정도로 더 소요된다.

4. 부하분산 QoS 라우팅

4.1 transit traffic 및 intra traffic을 위한 QoS 라우팅

선계산 방식의 QoS 라우팅에서는 특히 도메인의 ingress border router로부터 egress border router까지의 transit traffic에 대한 경로 설정 및 자원예약의 성공이 중요하다. 이는 transit traffic을 위한 경로를 intra traffic을 위한 경로보다 더욱 빈번하게 요구하며 이러한 경로를 중심으로 혼잡이 일어날 가능성이 높기 때문이다. 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 문제라 하겠다.

본 논문에서는 앞 절에서 제시한 비용산정 방식을 이용하여 다음과 같은 순서로 transit traffic 및 intra traffic의 다중 경로를 설정할 것을 제안한다. 먼저 도메인의 모든 ingress border router로부터 egress border router들간의 transit traffic을 위한 K 개의 경로를 설정하며 이 때 경로간의 부하 분산을 이루도록 한다. 이제 transit traffic을 위한 경로에서 사용된 자원은 부하분산을 위한 비용이 고려되었으므로 비용이 증가되었을 것이다. 따라서 이제 intra traffic을 위한 경로 설정을 할 때, 고 비용의 자원 사용을 회피하도록 유도함으로써 transit traffic을 위한 경로와 전부 혹은 일부가 가능한 한 중복되지 않도록 한다.

위와 같은 방법으로 transit traffic 간의 경로 중복은 물론, transit traffic과 intra traffic 간의 경로 중복을 회피하도록 함으로써, 결과적으로 자원예약 단계에서 transit traffic을 위한 자원예약이 성공할 가능성을 높이게 된다. 다음 그림1은 K 다중 경로계산을 위한 제안된 비용산정 방식을 이용한 선계산 알고리즘이다.

Algorithm Precomputation(D)

/*주어진 도메인 D에 대하여 border router쌍간

에 먼저 K 다중 경로 계산 후에 intermediate router 쌍간에 K 다중 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 1. 선계산방식의 알고리즘과 제안된 비용산정 방식

4.2 선계산 방식을 이용한 경로 선택

제안한 선계산 방식에서의 경로선택은 경로설정 요구시 선계산 방식으로 결정된 K 다중 경로 중에서 하나의 경로를 선택해야 한다. 이에 본 논문에서는 선계산 방식에서 사용된 $L_k(e)$ 와 경로선택 확률 $pr(k)$ 를 이용하여 경로 선택하는 방식을 제안한다.

그림 2에서는 링크 e 의 예측부하율 $L_k(e)$ 를 $K = 5$ 에 대해 표현하였다. δ 의 값이 1에 가까울수록 링크 e 의 예측부하율 $L_k(e)$ 가 k 의 증가($1 \leq k \leq K$)에 따라 고르게 증가하는 것을 알 수 있다. 이것은 실제 경로 선택 과정에서도 K 개의 경로를 비슷한 확률로 선택하여야 함을 의미하게 된다. 반면 δ 의 값이 0에 접근할수록, 링크 e 의 예측부하율 $L_k(e)$ 가 k 의 증가에 따라 급격히 증가하는 것을 알 수 있다. 이것은 실제 경로 선택과정에서도 이러한 증가와 비례하여 경로를 선택하여야 함을 의미하게 된다. 이와 같이 어떠한 δ 의 값을 선택하여야 하는가 하는 문제는 실제

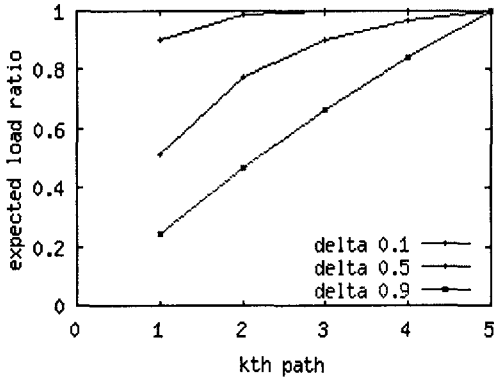


그림 2. K = 5에서의 예측부하율($L_k(e)$)

고려중인 네트워크의 특성을 반영하여 경로선택 과정과 연계하여야 한다.

제안된 비용산정 방식을 이용한 선계산 알고리즘을 통해 K개의 다중 경로가 계산되었다. 실제 경로요구 시 어떤 경로를 어떤 확률로 선택하여야 하는가는 비용산정시 사용된 값에 근거하게 된다. 이들 경로들의 선택은 다음의 k에 따른 확률을 통해 이루어진다.

$pr(k)$ 를 k 번째 경로가 선택될 확률이라 할 때, 앞의 식(1)에서 다음을 유추할 수 있다.

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

그러면 다음의 2가지 관계가 성립된다.

(관계1) $pr(1st) \geq pr(2nd) \geq \dots \geq pr(K) > 0$

(관계 2) $\sum_{k=1}^K pr(k) = 1$

그림 3에서는 경로선택 확률 $pr(k)$ 를 K=5에 대해 표현하였다.

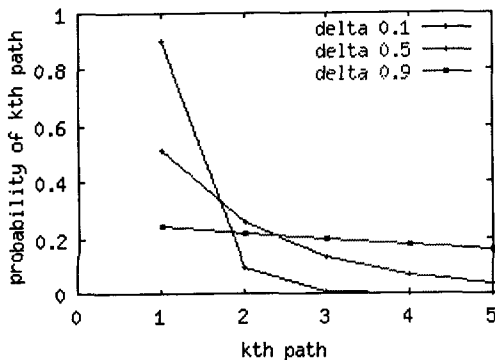


그림 3. K=5에서의 kth경로들의 경로선택 확률

그림 3에 나타난 바와 같이 δ 의 값이 1에 접근할수록 k의 증가에 따라 경로선택 확률 $pr(k)$ 가 고르게 감소한다. 따라서 K 다중경로를 비슷한 확률로 고르게 선택 할 수 있다는 것을 알 수 있다. 반면 0에 접근할수록 선택의 폭이 줄어들게 된다.

다음 그림 4는 K 다중경로 중 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 */

```

{
    Δsum = 0
    for k = 1 to K do
        Δsum + = δk;
    for k = 1 to K do
        pr(k) = δk / Δsum;
        selects the k-th path with the probability
        pr(k)
        among K precomputed paths from s to t;
}
    
```

그림 4. 제안된 경로선택 알고리즘

5. 시뮬레이션 및 결과 분석

본 논문에서 제안한 부하분산 QoS 라우팅을 분석하기 위하여 시뮬레이션을 수행하였다. 한 도메인을 구성하는 네트워크를 모델링하기 위하여 Waxman의 그래프 모델[9]을 근거하여 50개의 노드를 가지는 그래프를 구성하였다. 부하분산을 검증하기 위한 네트워크 내의 상당한 트래픽을 가정하기 위하여 300개의 노드 쌍의 연결에 대하여 100회 실험하여 평균치를 측정하였다. 시뮬레이션은 범용 시뮬레이션 언어인 SIMSCRIPT II.5를 이용하였다. 시뮬레이션을 간단히 하기 위해 네트워크내 모든 링크는 같은 대역폭을 갖고 있고, 각 연결은 동일한 대역폭을 요구한다고 가정하고 $c(e)$ 와 $d(e)$ 의 값을 각각 1로, δ 의 값을 0.25로 가정하였다.

시뮬레이션을 통해 제안한 부하분산 QoS 라우팅에서 고려하고 있는 전체 네트워크 자원의 사용과 네트워크 자원의 부하분산된 사용의 결과를 검증할

수 있도록 하였다. 이를 위해 연결상의 모든 링크의 사용횟수(used count)를 측정하였다.

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 낮게 되어 네트워크 자원의 부하분산된 사용의 결과를 얻을 수 있게 된다. 이와 반대로 사용횟수가 많은 링크의 개수가 많을수록 혼잡의 발생확률은 높아지고 부하분산된 사용의 결과를 얻기 어렵게 된다.

모든 연결에 대한 각 링크의 사용 횟수를 부하분산의 척도로 측정하였다. 시뮬레이션은 부하분산요소(α) 값에 따른 부하분산 정도를 나타내기 위하여 0에서 1까지의 값들을 적용하였다. 부하분산요소가 0일 때는 부하분산을 고려치 않는 최소 hop경로를 의미한다.

본 논문에서 제안한 부하분산 비용산정 방식을 적용하기 위하여 각 연결에 대하여 *Dijkstra* 알고리즘에 의한 경로계산 후 식(1)을 적용하여 경로상의 모든 링크의 비용을 증가시켰다.

분석 결과, 그림 5와 표 1과 같이 $\alpha=0$ 에서 사용횟수가 높은 링크들의 개수가 많이 있음을 발견할 수 있었다. 사용횟수가 높다는 것은 부하분산의 효과는 떨어지고 혼잡의 발생확률이 높다고 판단할 수 있다. 또한 $\alpha=1$ 에 접근할수록 사용횟수가 높은 링크수가 적어지게 되었다. 이는 혼잡이 덜 발생함을 의미하고 네트워크 전체에 걸쳐 부하가 분산된 경로설정이 되었다고 판단할 수 있다.

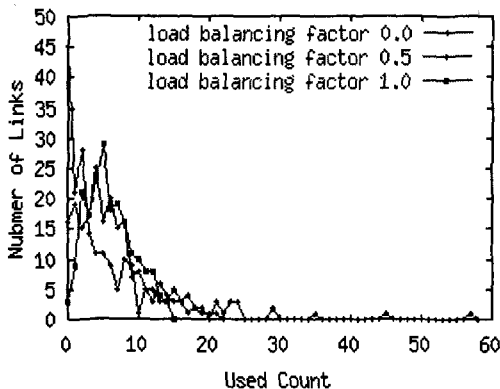


그림 5. 부하분산 비용산정의 시뮬레이션 결과

표 1. 단일 도메인에서 300개 연결의 최대 사용 개수

α	1st 최대사용개수	2nd 최대사용개수	3rd 최대사용개수
0.0	57	45	34
0.25	45	35	24
0.5	21	20	19
0.75	18	16	15
1.0	14	13	12

한편, 전체 연결들의 네트워크 자원 사용은 표 2와 같이 부하분산요소 α 값에 따라 반복된 실험을 거쳐 사용된 링크 총 개수를 구하였다.

표 2와 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. 즉 $\alpha=0$ 에 비하여 $\alpha=1$ 일 경우 약 12.4%의 자원을 더 사용하였다.

이와 같은 시뮬레이션에서 나타난 것과 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 부하분산 QoS 라우팅 방식은 부하분산과 네트워크자원 사용의 최적화 목적에 따라 부하분산요소의 값을 변화하면서 네트워크 특성에 맞추어 적용할 수 있다.

표 2. 네트워크 자원 사용에 대한 결과

α	0.0	0.25	0.5	0.75	1.0
전체사용횟수	1308	1325	1364	1403	1470

6. 결 론

본 논문에서는 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. 또한 도메인 간 라우팅에서 transit traffic의 자원 예약을 intra traffic에 우선하는 방식을 제안하여 도메인간 라우팅과 도메인내 라우팅을 무리 없이 연계할 수 있도록 하였다. 그리고 제안된 비용산정 방식에 따라 계산된 K 다중 경로들 중에서 제안한 경로선택확률에 따라

경로를 선택하는 방식을 제안하였다. 따라서 전체 네트워크자원의 최적 사용과 부하분산을 위한 효율적 해결방안을 제시할 수 있게 되었다.

본 논문에서 제안한 계층적 라우팅에서의 부하분산 QoS 라우팅의 분석을 위하여 시뮬레이션을 수행한 결과 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 부하분산 QoS 라우팅 방식은 네트워크 자원의 부하분산과 전체 네트워크 자원 사용 목적에 따라 부하분산요소의 값을 변화하면서 네트워크 특성에 맞추어 적용할 수 있다.

참 고 문 헌

[1] 김승훈, 김치하, “분산멀티미디어 응용을 위한 대규모 고속 통신망에서의 QoS-근거 계층적 도메인간 라우팅 방식,” 한국통신학회논문지, 제24권 제7호, 1999.

[2] Seung-Hoon Kim, Kyungshik Lim and Cheeha Kim, “A Scalable QoS-based Inter-Domain Routing Scheme in a High Speed Wide Area Network,” *Computer Communications*, Vol. 21, No.4, pp. 390-399, 1998.

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

[4] G. Apostolopoulos, R. Guerin, S.Kamat, A. Orda and S. K. Tripathi, Intra-Domain QoS Routing in IP Networks: A Feasibility and

Cost/Benefit Analysis, *IEEE Networks*, Vol. 13, No. 5, pp. 42-54, 1999.

[5] Ariel Orda and Alexander Sprintson, QoS Routing: The Precomputation Perspective, *Proceedings IEEE INFOCOM2000*, Mar., 2000.

[6] 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.

[7] Murali Kodialam, T. V. Lakshman, Minimum Interference Routing with Applications to MPLS Traffic Engineering, *Proceedings of IEEE INFOCOM2000*, Mar., 2000.

[8] Hiroyuki Saito, Yasuhiro Miyao, and Makiko Yoshida, Minimum Interference Routing with Applications to MPLS Traffic Engineering, *Proceedings of IEEE INFOCOM2000*, Mar., 2000.

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



홍 종 준

1991년 인하대학교 전자계산학과 (공학사)
 1993년 인하대학교 전자계산공학과 (공학석사)
 2002년 인하대학교 전자계산공학과 (공학박사)
 1993년~1998년 LG산전연구소

1999년~현재 청강문화산업대학 컴퓨터소프트웨어과 조교수
 관심분야 : 라우팅, 네트워크 보안, 무선통신

교 신 저 자

홍 종 준 (220-130) 강원도 원주시 우산동 동보렉스Apt. 101동 306호