

QoS 응용 서비스를 위한 효율적인 다중 메트릭 라우팅 방안

정회원 전한얼*, 김성대**, 이재용*, 김동연***, 김영준**

Network Efficient Multi-metric Routing Algorithm for QoS Requiring Application

HahnEarl Jeon*, Sung-Dae Kim**, Jai-yong Lee*, Dong-Yeon Kim**, Young-Joon Kim**

Regular Members

요약

본 논문은 여러 개의 다중 메트릭을 이용하여 경로 선택을 수행하는 문제에 대해 연구했다. 현재 인터넷에서의 라우팅은 하나의 메트릭을 이용하여 경로선택을 수행하고 있으며, 하나의 메트릭에 의해 선택된 경로는 하나의 요구사항만을 충족시켜 줄 수 있는 best-effort 서비스이다. 여러 서비스 품질(QoS)을 요구하는 콜에 대해 만족시켜주기 위해서는 경로가 여러 제한에 걸리게 된다. 많은 경우에 경로 선택문제라는 것은 NP-complete이다. 본 논문에서는 지연 메트릭에 기반을 두고 네트워크 자원에 대한 정보에 영향을 받는 비용 메트릭을 이용한 widest-least cost 라우팅 알고리즘을 제안한다. 이 알고리즘은 다중의 메트릭을 이용하는 라우팅 알고리즘으로써 트래픽 분산 능력을 갖고 있어서, 네트워크 링크가 부하가 적을 때는 최소거리 경로를 선택하도록 하는 것이고, 네트워크 부하가 클 경우에는 트래픽 엔지니어링이 수행되어 트래픽을 다른 경로로 이동시키도록 하는 것이 목표이다. 이러한 결과는 다른 알고리즘과의 실험 결과를 통해 비교하였다

ABSTRACT

In this paper, we have studied path selection problem using multiple metric. Current Internet selects a path using only one metric. The path selected by one metric is a best-effort service that can satisfy one requirements. In order to satisfy a call with various Quality-of-Service(QoS) requirements, the path must satisfy multiple constraints. In many cases, path selection is NP-complete. The proposed algorithm is widest-least cost routing algorithm that selects a path based on cost metric which is basically a delay metric influenced by the network status. The proposed algorithm is a multiple metric path selection algorithm that has traffic distribution ability to select shortest path when network load is light and move traffic to other alternate path when the link load is high. We have compared the results with other routing algorithms.

I. 서론

전통적인 통신 네트워크 구조는 사용자들에게 단순한 서비스 품질(QoS)을 보장하도록 설계되었다. 현재 인터넷에서의 라우팅은 단순한 best-effort

서비스를 제공하는 능력에 연결성이 초점 맞추어져 있다. 그러나, 디지털 비디오나 오디오와 같은 다양한 멀티미디어 서비스들이 나타나면서 이러한 서비스들은 네트워크 서비스에 대한 요구를 증가시켰으며, 다중의 다양한 서비스 품질을 기대하게 되었다.

* 연세대학교 전기,전자공학과 네트워크 연구실(hearlj@yonsei.ac.kr),

** LG전자,

*** 한경대학교

논문번호 : 020174-0418, 접수일자 : 2002년 4월 18일

* 본 연구는 LG 전자 연구과제 지원으로 수행되었습니다.

OSPF나 RIP와 같은 현재 인터넷에서 사용되는 라우팅 프로토콜은 최단 경로 라우팅을 사용한다. 이는 하나의 메트릭에 최적화된 알고리즘이다^[2]. QoS 라우팅은 네트워크에서 다양한 서비스들이 자신만의 특정한 QoS를 보장시켜 주기를 기대하는 연결서비스에 대해서 그 요구를 만족시켜주는 것이 가능하도록 하는 도구중 하나이다. 이는 네트워크에서 충분한 자원을 가진 경로를 찾도록 해준다. QoS 라우팅은 현재 인터넷에서 필요하면서도 빠져있는 조각이며 네트워크가 요구하는 다양한 품질을 제공하도록 해주는 중요한 요소이다. QoS 라우팅은 IP가 가진 제한을 극복하기 위해 주목을 받고 있으며, 최근 몇 년간 QoS 라우팅은 여러 연구와 제안의 주제가 되어왔다.^{[3]-[6],[10],[12]-[15],[17]-[21], [23],[25]}

현재의 라우팅을 확장시키기 위해, 라우팅 알고리즘은 다중의 제한내에서 경로를 계산해야 한다. 현재까지의 연구결과에 의하면, 하나 이상의 메트릭을 이용해서 가장 적절한 경로를 계산해내는 것은 불가능한 것으로 알려져 있다^[5]. 다중 제한을 가진 최단 경로 문제는 잘 정의되지 않는다. 그러므로, 여러 경로 중에 가장 믿을 수 있는 경로를 선택하고 이를 최적화하는 것은 매우 중요한 일이다. 만약 메트릭이 네트워크 상태에 따라 동적으로 변하는 것이라면, 경로를 선택할 때 정확한 값을 아는 것도 중요하다. 그러나, 네트워크 크기와 토폴로지 요약은 네트워크 트래픽과 라우팅의 tradeoff 관계가 되며, 부정확한 라우팅 정보의 원인이 된다. [10]은 경로 선택에서 부정확한 정보의 영향을 다루고 있다. 그럼에도 불구하고, QoS를 요구하는 연결이나 여러 연결들에 대해 가장 최선의 서비스를 제공할 수 있는 경로는 반드시 선택해야 한다.

네트워크 부하가 작을때는 best-effort 서비스나 집중된 서비스에 차이가 없다. 성능은 트래픽 부하가 클 때 생기는 것이다. 네트워크 정체는 네트워크 자원의 부족이나 트래픽이 불균형하게 분포되었을 경우 발생한다. 트래픽 엔지니어링 (Traffic Engineering)은 트래픽을 적절하게 배치하여 네트워크 정체를 피하도록 하는 과정이다. 트래픽 엔지니어링은 한정된 망 자원을 가지고 더 높은 수익을 올리하고자 하는 상용 ISP 사업자들에게는 아주 중요한 기법이다. 네트워크가 들어나는 트래픽 양을 최대한 수용하면서, 그러한 트래픽이 요구하는 QoS를 적절히 보장해 줌과 동시에 운용하는 네트워크의 자원을 최적화해서 사용할 수 있는 네트워크 구조로 변화

하는 QoS를 적절히 보장해 주면서, 네트워크 자원의 사용도 최적화하기 위한 다양한 연구가 필요하다. 낮은 비용의 경로로 변경^{[7],[21]}, 다중 경로를 통한 트래픽 전송^[13], 대역폭에 따른 비용 설정^[23]은 traffic engineering을 실현하는 방법들이다. 본 논문에서는 다중 메트릭을 이용한 경로선택 문제에 초점을 맞추고 있다. 제안한 알고리즘은 또한 트래픽 분산 능력을 갖고 있어서 트래픽 엔지니어링 기능을 수행할 수 있다.

본 논문의 구성은 다음과 같다. II 장에서는 현재 제안되어 있는 QoS 라우팅 알고리즘과 문제점을 살펴본다. III 장에서 widest-least cost 다중 메트릭 라우팅 알고리즘을 제시하고, IV장에서 실험을 통한 그 결과를 다른 알고리즘과 비교한다. V장에서 그 결과를 요약한다.

II. QoS를 고려한 라우팅 알고리즘

라우팅 알고리즘의 목적은 존재하는 가능한 경로를 찾는 것이다. 주어진 네트워크에서 서비스를 요구하는 소스와 그 목적지 사이에 요구되는 서비스를 만족시키는 가능한 경로는 하나 이상 존재할 수 있다. 사용 가능한 경로는 Dijkstra의 최소 경로 알고리즘이나 Bellman-Ford 최소 경로 알고리즘을 사용하여 찾을 수 있으며, 많은 경우 하나 이상의 사용 가능한 경로들이 존재할 수 있는데, 이렇게 여러 경로를 찾았을 때, 결국 어느 경로가 자원을 가장 효율적으로 사용하는 지 선택하는 것이 필요하다.

네트워크는 여러 응용 서비스를 지원하는데, 이러한 응용 서비스를 이용하는 사용자들의 다양한 요구사항들이 네트워크를 발전하게 하고, 각기 다른 네트워크 응용 서비스들은 자신에 맞는 다른 네트워크 서비스를 요구하므로 네트워크로 하여금 다른 동작을 하게끔 요구하는 것이다. 이러한 서비스들이 제공되면서 증가된 네트워크 트래픽은 기존의 갖고 있는 네트워크 대역폭 또한 증가할 것을 요구하지만, IP 전화와 같은 새로운 응용서비스들은 대역폭 증가 외에 또 다른 요구사항을 가진다. 따라서, 단순히 양적인 대역폭 증가가 그 유일한 해답이 아니다. 요구 조건이 서로 다른 트래픽들의 성능을 적절히 보장해 주면서 가능한 효율적인 경로를 찾도록 하는 것이 QoS 라우팅의 목적이다. [10]에서는 여러 QoS 라우팅 알고리즘에 대한 성능 비교를 수행했다. 라우팅 오버헤드에 직접적인 영향을 주는 것들로 라우팅을 수행하기 위해서 어떤 라우팅 정보

가 사용되고, 이러한 정보를 이용하여 언제 경로가 계산되는지 등이 있다.

라우팅 알고리즘은 자원 소비를 제한하고 네트워크 부하 균형을 맞추므로써 자원 효율을 얻게 한다. 대역폭을 보장하는 트래픽에 대해서는 대역폭을 반드시 맞추어 주어야 하기 때문에 최소 홉 수를 가질 필요가 있으며, 네트워크 부하는 최소 부하의 경로를 선택함으로써 균형을 맞출 수 있다. 그러나, 이러한 두 가지 최적화 과정은 서로 맞지 않을 수 있다. [19]에서는 최적화에 대해 비교를 했다. 여러 경로 선택 알고리즘들은 제한적인 홉 수와 네트워크 부하의 균형을 맞추기 위해 다른 가중치를 준다. 현재 제안되어 있는 알고리즘들을 살펴보면, Widest-shortest 알고리즘은 소스와 목적지 사이에서 가능한 모든 경로 중에서 최소 홉 수를 가진 경로를 선택한다. 만약 이러한 경로가 여러 개 존재하는 경우 최대 예약 대역폭을 가진 경로가 선택된다. 또 그러한 경로가 2개 이상 존재하는 경우에는 임의의 한 경로가 선택된다. Shortest-widest 알고리즘은 이전 알고리즘과 달리 소스와 목적지 사이에서 가능한 모든 경로 중에서 최대 대역폭 경로를 가진 경로를 선택하며, 이러한 경로가 여러 개 존재하는 경우 최소 홉 수로 이루어진 경로가 선택된다. 이러한 경로가 2개 이상 존재하는 경우에는 임의의 한 경로가 선택된다. 최소거리 알고리즘은 정의된 거리함수를 이용하여 경로 p에 대한 최소 거리를 가진 가능한 경로를 선택하는 것이고, 거리 함수는 다음과 같이 남아있는 대역폭 R_{ij} 를 이용하여 정의할 수 있다.

$$dist(p) = \sum_{i,j} 1/R_{ij}.$$

마지막으로, Dynamic-alternative 알고리즘은 네트워크가 복잡하지 않을 때 최소 홉 수 경로가 n 홉일 경우, dynamic-alternative 경로는 n+1 홉을 넘지 않는 widest-shortest 경로이다.

QoS 라우팅을 수행하는 방법으로 알고리즘에 대한 연구를 언급했는데, 이 외에도 이러한 라우팅 알고리즘을 효과적으로 적용하기 위한 방법들에 대한 연구들이 있다. 라우팅 정보의 정확성과 라우팅의 확장성 또한 중요한 문제이다. [11]에서는 부정확한 라우팅 정보를 이용하면서도 QoS 라우팅을 수행하기 위한 방안이 제기되었고, [21]에서는 서버를 기반으로 한 라우팅, [25]에서는 확장성을 위한 토폴로지 요약물 다루고 있다.

네트워크에서 라우팅을 수행할 경우, 메트릭은 네트워크를 표현하는 수단이다. 그러므로, 노드 사이

에서 경로를 계산할 때, 사용하는 메트릭에 따라서 결과적인 경로를 결정한다. Widest-Shortest 경로 계산 방법과 Shortest-Widest 경로 계산 방법이 하나 이상의 메트릭으로 경로를 계산하지만, 그 결과적인 경로는 첫 번째 메트릭의 영향을 주로 받는다. 그 이유는 경로 계산을 수행하는 중에 두 번째 메트릭은 첫 번째 메트릭이 최선의 경로를 선택할 수 없는 경우에만 사용되기 때문이다. 따라서 사용되는 첫 번째 메트릭에 의해 라우팅 알고리즘과 네트워크 효율의 대부분이 결정된다. Shortest -Widest 알고리즘은 load 분배에 초점을 두고 있으며, Widest-Shortest 알고리즘은 짧은 경로에 초점을 두고 있다. Shortest-Widest 알고리즘의 단점은 충분한 대역폭을 가진 짧은 경로가 존재하더라도 넓은 대역폭의 먼 경로를 선택하게 될 수 있다는 점이다. 이전 연구([18],[25])에 따르면, Widest -Shortest 경로는 다른 알고리즘보다 네트워크 부하가 많을 경우, 토폴로지 요약물 사용할 경우에 좋은 성능을 보여주고 있다.

III. 제안하는 라우팅 알고리즘 : Widest-Least Cost 라우팅 알고리즘

본 장에서는 하나 이상의 메트릭을 이용하여 경로 계산을 수행했을 경우의 제한점과 이를 개선하기 위한 방법을 제안한다.

제안하는 알고리즘의 핵심은 네트워크 링크를 사용하고 있는 정도에 따라 네트워크 지연과 대역폭 메트릭을 이용해서 이용할 수 있는 새로운 메트릭을 정의하는 것이다. 이 메트릭은 지연 메트릭을 기반으로 하여 대역폭 메트릭을 적용한다. 이 메트릭은 네트워크 부하가 증가되어서 미리 정의한 임계치를 넘어설 경우 이 링크에 가중치를 주어서 선택할 가능성을 줄이도록 하여 트래픽을 다른 링크로 옮기도록 한다. 즉, 현재 링크가 임계치 이상 얼마만큼 이용하고 있는가를 가중치로 지연에 부과함으로써 라우팅 계산에서 그 링크 이용을 줄이는 것이다.

그림 1에서와 같이 링크가 임계치 이상 사용되고 있는 경우 임계치 이상 링크 대역폭(I-T)에서 현재 링크가 사용하고 있는 비율((1-b/c)/(b/c-T))을 가중치로 지연에 주어 링크가 임계치 이상 사용되는 정도가 클수록 가중치를 늘이도록 하였다.

이 메트릭 값은 임계치 이하일 경우 지연값을 그대로 사용하지만, 임계치를 넘어설 경우 다음 식을 사용해서 링크 l에서 사용할 메트릭 비용값 $cost(l)$ 을 계산하게 된다.

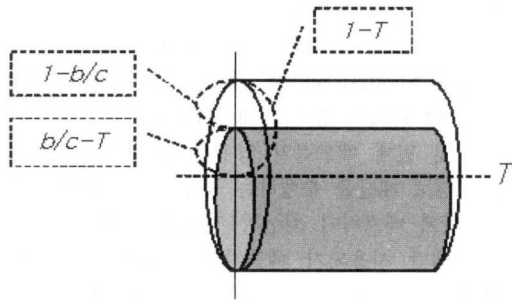


그림 1. 임계치이상 네트워크 링크가 사용된 경우

$$\cos t(l) = d_l \times \left(1 + \sigma \times \left(\frac{b_l}{c_l} - T \right) \times \frac{1-T}{1-\frac{b_l}{c_l}} \right) \quad (1)$$

여기서 d_l 은 링크 l 의 지연값이고, T 는 대역폭에서 임계값이며, σ 는 바이어스값으로 계산식에서 사용하는 부하값이며, b_l 은 현재 링크 l 의 대역폭이며, c_l 은 링크 l 의 대역폭 용량이다. 바이어스값 σ 는 전체 비용값을 조절하기 위해서 사용한다. 이 식은 대역폭율과 임계율의 비율 형태를 띠고 있다. 만약 σ 값이 클 경우, 결과적인 비용값을 증가시키게 된다. 따라서 이 값은 결과 비용이 변화되는 값의 변화 정도를 조절하는 값으로 이용할 수 있다. 그러므로, σ 는 알고리즘의 성능을 변화시키게 될 것이다.

이제 widest-least 비용 알고리즘을 제안한다. 방향성 그래프 $G = (N,A)$ 에서 노드 $1, \dots, N$ 과 호 A 를 보자. 임의의 노드 i, j 사이의 각 호 (i,j) 에서는 비용 또는 길이 a_{ij} , 가용 대역폭 b_{ij} , 그리고 그와 연관된 대역폭 용량 c_{ij} 을 갖고 있다. 노드를 연결하는 주어진 방향성 전달 경로 $p=(i, j, k, \dots, l, m)$ 에서 경로의 넓이 $width(p)$ 는 경로에서 병목 대역으로 정의한다. $width(p) = \min(b_{ij}, b_{jk}, \dots, b_{lm})$. 그리고, 경로의 길이 $length(p)$ 는 호의 길이의 합으로 정의한다. $length(p) = a_{ij} + a_{jk} + a_{lm}$. 이 경로는 같은 출발지와 목적지인 경우에 다른 모든 경로에 비해 최소의 길이를 가질 경우 최단값으로 정의한다. l 에서 i 로의 몇몇 경로 p_i 에 대해 d_i 는 모든 i 에 대해 경로 p_i 의 길이로 해석될 수 있다.

$d_i = length(p_i)$. V 는 후보 리스트이고, B_i 는 경로 p 를 따르는 넓이 $width(p)$ 로 정의한다. widest-least cost 경로 설정 알고리즘은 동일한 비용의 경로가 있을 경우에 대역폭 확인하는 절차를 추가함으로써 만들어질 수 있다.

widest-least cost 알고리즘은 다음과 같다.

1) Initially,

$$V = \{1\} \quad d_1 = 0, \quad d_i = \infty, \quad B_i = \infty, \quad \forall i \neq 1.$$

2) The method proceeds in iterations and terminated when V is empty.

Iteration of Widest-Least Cost

3) Remove from the candidate list V a node i such that

$$d_i = \min_{j \in V} d_j$$

4) For each outgoing arc $(i, j) \in A$,

a) if $\frac{b_{ij}}{c_{ij}} > T$

a_{ij} = result from cost equation

b) if $d_j > d_i + a_{ij}$, set

$$d_j = d_i + a_{ij}$$

$$B_j = \min\{B_i, a_{ij}\}$$

c) else if $d_j = d_i + a_{ij}$

if $B_j < \min\{B_i, a_{ij}\}$

$$d_j = d_i + a_{ij}$$

$$B_j = \min\{B_i, a_{ij}\}$$

5) add j to V if it does not already belong to V . go to 2)

제안하는 알고리즘 동작은 다음과 같다. 초기에 소스노드를 제외한 다른 노드는 연결이 되어 있지 않은 무한대로 정의된다. 우선 소스노드부터 시작하여 후보 노드리스트 V 에 추가되며, 후보노드 리스트에 속한 노드들에 연결된 노드들로 최소 경로를 계산하면서 거리 비용이 작은 값을 노드거리로 계산하면서 loop을 반복한다. loop을 계산하는 도중에 네트워크 부하가 임계치를 넘어설 경우, 제안하는 widest-least cost 알고리즘은 가중치를 계산하여 지연값을 계산하게 되며 부과된 가중치는 트래픽을 다른 링크로 분배시키기 시작한다. 만약 링크가 임계치 이하일 경우에는 링크의 지연을 그대로 사용하여 목적지까지 최소 거리를 비용값으로 유지한다. 같은 거리 비용일 경우에는 Widest-Shortest 경로 알고리즘과 유사하게 넓은 대역의 경로를 선택한다. 이런 과정을 통해 loop을 반복하면서 거리가 계산된 노드들은 후보노드리스트에서 제거된다. 모든 노드에 대해서 비용을 계산하고 나면 알고리즘이 종료한다. 그러므로, Widest-Shortest 경로 알고리즘은 widest-least cost 알고리즘 동작 형태 중 한 가지 경우와 동일하게 된다.

IV. 시뮬레이션 및 결과

본 장에서는 앞에서 제안한 QoS 라우팅 알고리즘과 라우팅 비용함수(cost function)를 사용하여 시뮬레이션을 한 결과를 보인다.

QoS를 만족시켜주어야 하는 트래픽에 대해 제안한 라우팅 알고리즘을 실험을 통해 기존의 다른 라우팅 알고리즘과 성능을 비교한다. 이 실험의 목적은 제안한 알고리즘이 네트워크 효율이나 네트워크 부하 분배에 있어서 다른 알고리즘보다 좋은 성능을 보이는 것을 확인하는 것이다. 이를 통해 네트워크에서 연결을 받아들이는 비율을 증가시키려는 것이다.

실험은 여러 토폴로지 형태에서 실험을 수행하였다. 그 중 그림 2은 개념적 모델 토폴로지로서 기본적인 동작과 제안된 알고리즘의 특성을 확인하기 위한 토폴로지이다. 성능 평가를 위해서 여러 토폴로지를 사용하였고, 그림 2와 같은 사각형 토폴로지의 결과를 살펴본다.

시뮬레이션은 소스와 목적지는 랜덤하게 발생되어 연결을 시도한다. 모든 연결은 포아송(poisson) 분포를 따라 발생하고, 그 연결이 성공적으로 이루어졌을 때, 그 연결 유지 시간은 지수(exponential) 분포를 따른다. 콜이 발생한 후에, 경로 설정은 성능 평가를 위한 라우팅 알고리즘에 따라 각각 선택이 된다. 각기 알고리즘에서 경로를 선택하면 콜은 각 알고리즘 결과에 따라 주어진 경로를 따라 연결을 시도하고, 경로를 설정해본다.

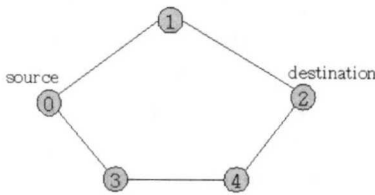


그림 2. 토폴로지 1

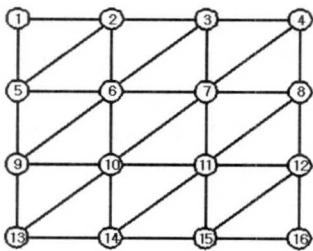


그림 3. 토폴로지 2

시뮬레이션 프로그램은 소스와 목적지 호스트를 임의로 선택하고 콜 연결을 발생시킨다. 모든 콜은 포아송(poisson) 분포를 따라 발생한다. 각 콜이 성공적으로 연결이 이루어지면, 그 연결이 유지되는 시간은 지수(exponential) 분포를 따르는 값을 갖는다. 콜이 발생된 후에, 각 콜에 대해 가능한 경로는 그 성능을 비교하기 위한 알고리즘들에 따라 찾게 된다. 만약 알고리즘이 연결을 위한 경로를 찾게 되면, 각 콜은 그 선택된 경로를 따라 연결을 시도한다. 만약 소스와 목적지 사이에서 요구되는 대역폭을 만족시키지 못하거나 종단간 지연이 요구되는 값을 초과한다면 그 콜은 거절된다. 성능 비교를 위해 사용한 알고리즘은 dijkstra 알고리즘, shortest-widest widest-shortest 알고리즘, 그리고 제안된 알고리즘이다.

실험에서 여러 임계치(threshold) 값과 바이어스 값을 사용했는데, 임계치와 바이어스값은 네트워크의 구조와 형태, 그리고 요구되는 서비스에 따라 성능이 차이가 난다. 주어진 그림2와 그림3의 토폴로지에서 임계치를 50%에서부터 계속 증가시키면서 결과를 확인하였고 성능이 조금씩 향상되는 중 80%에서 성능이 좋았으며, 본 논문에서는 그 결과를 나타내었다. 네트워크 사용량이 경로 대역폭의 80% 사용량에서 우회경로를 고려하도록 하기 위해 사용한 임계치는 0.8이고 바이어스 값은 커질수록 고려하는 우회경로가 많기 때문에 여러 경로를 사용할 수 있다. 바이어스 값이 추가된 제안 알고리즘에서 가중치가 주어진 경로 비용 d_1 은 식(1)을 통해 구할수 있는데, 이웃 노드까지 최단 경로 d_1 의 값이 그 경로를 제거한 경우 최단 경로 d_2 보다 값이 커야 경로를 우회시킬 수 있다. 그 경우 바이어스 값을 구하자면 다음과 같다.

$$a) > \frac{(\frac{d_2}{d_1} - 1)(1 - \frac{b}{c})}{(\frac{b}{c} - T)(1 - T)} \quad (2)$$

본 논문에서 제시한 조건과 환경에서 링크 사용량이 90%가 되었다면 식(2)를 사용하여 계산한 노드 7과 노드 4 사이에서 필요한 바이어스 값은 4이상이고, 노드 7과 노드 8에서 계산된 바이어스 값은 6이다. 그러나, 링크 사용량이 95%이면 각기 1.333과 2이다. 따라서 적절한 바이어스 값은 상황에 따라 달라지게 된다. 본 논문에서는 바이어스 값을 1에서 5까지 증가시키면서 실험을 수행한 결과, 3까

지 성능이 좋아지다가 3이상의 값에서는 큰 차이를 보이지 않았다. 따라서 결과 그래프에서는 바이어스 값을 3을 사용한 결과를 보였다.

가장 적절한 바이어스 값과 임계치라는 것은 네트워크 구조와 네트워크에 주어지는 트래픽의 특성에 따라 변할 수 있는 값이므로, 최적의 값은 주어진 환경 속에서 실험을 수행하여 얻어낸 결과가 해당될 것이다.

제안된 알고리즘의 성능을 비교하기 위해서, 두 가지 성능값을 사용했다: 콜 거절 비율(call rejection rate(CRR))과 대역폭 효율(bandwidth efficiency (BWE))이다. CRR은 네트워크에서 연결 설정을 요구한 콜의 갯수 중에서 연결을 거절당한 콜의 수를 비율로 나타낸 것이고, ABWE(Average BWE)는 하나의 링크에 대해 전체 실험동안 효율의 평균을 구한 결과를 말한다. CRR과 BWE는 다음과 같이 구할 수 있다.

$$CRR = \frac{n_{rej}}{n_{req}}$$

$$BWE = \frac{b_l}{c_l}$$

여기서 n_{rej} 는 거절당한 콜의 수로써 네트워크에서 연결을 요청하여 발생한 전체 콜 중에서 연결을 시도하였다가 거절당한 콜의 수를 합산하였으며, n_{req} 는 연결을 요청한 전체 콜의 수로써 네트워크에서 연결을 요청하여 발생한 전체 콜의 개수를 합산하였다. b_l 은 현재 네트워크 링크가 사용되고 있는 대역폭으로 구할 수 있다.

우리의 경로 설정의 목표는 CRR을 줄이면서 네트워크 자원을 효율적으로 이용하는 경로를 찾는 것이다.

그림 2의 토폴로지에서 두 가지 상황을 고려하였다. 이 간단한 실험은 제안된 알고리즘의 성능을 보여 줄 것이다. 첫번째 경우는 토폴로지의 윗쪽 경로와 아랫쪽 경로의 지연이 동일한 경우이다. 표 1의 결과를 보면 트래픽이 네트워크에 골고루 분포되어 있는 것을 알 수 있다. 링크들의 대역폭 사용 평균 값을 구한 ABWE를 보면, 그 최대값이나 최소값이 비슷하여 평균적으로 대역폭 사용이 모든 링크에서 동일하며, 거절된 콜이 없는 것을 알 수 있다.

이러한 상황은 윗쪽 경로가 아랫쪽 경로보다 지연이 작을 경우 달라진다. 각 링크의 지연이 모두 1이라는 값을 갖는다고 하면, 윗쪽 경로의 지연은 2가 되고, 아랫쪽 경로는 3이 된다.

표 2의 결과를 보면 dijkstra와 widest-shortest 알

고리즘은 모든 트래픽이 한쪽 경로에 치우쳐 있는 것을 알게 된다. ABWE값을 보면 최대값은 커졌으며 최소값은 0인 링크도 있다. 이 링크는 전혀 사용이 되지 않는다는 것이다. 한쪽 링크로 트래픽이 모두 전달되며 다른 링크로 전달되는 트래픽은 없다. 그러므로, 네트워크의 일부만이 사용되는 것이다. 반대로, 제안한 알고리즘은 ABWE값을 살펴보면 위쪽 링크를 주로 이용하지만, 윗쪽과 아랫쪽 경로 모두를 사용하고 있는 것을 알 수 있다. 결과적으로 콜 거절율이 줄어들게 된다.

표 1. 그림 1의 토폴로지에서 동일한 경로길이에 대한 결과

	Dijkstra	Widest-shortest	Widest-least cost
CRR	0.0%	0.0%	0.0%
Max ABWE	53%	46%	47%
Min ABWE	52%	46%	47%

표 2. 그림 1의 토폴로지에서 다른 경로길이에 대한 결과

	Dijkstra	Widest-shortest	Widest-least cost
CRR	0.28%	0.24%	0.0%
Max ABWE	84%	85%	71%
Min ABWE	0%	0%	22%

첫 번째와 두 번째 상황을 비교해 볼 때, widest-shortest 라우팅 알고리즘은 두 가지 메트릭을 이용해서 경로설정을 수행하지만, 몇몇 상황에서는 단지 첫 번째 메트릭에만 결과가 의존하게 된다. 그러므로, 트래픽이 한쪽으로 몰리게 되는 것이다. 제안한 알고리즘은 각각의 경우에 좋은 성능을 보여준다. 이는 경로 선택 뿐만 아니라, 트래픽을 분산하는 능력을 갖추고 있다는 것이다.

그림 3에 보여진 토폴로지에 대해, 시뮬레이션은 모든 소스와 목적지 호스트를 임의로 선택하도록 하였다. 각 콜이 요구한 대역폭은 512Kbps이고, 지연은 70ms이다. 트래픽이 네트워크에 주는 부하는 1분당 평균 콜 발생률로 변화를 주었다. 각 시뮬레이션은 10번 수행되었으며, 각기 600분의 시뮬레이션 시간을 수행했다.

그림 4은 토폴로지 2에서 실험을 수행한 결과 콜 거절 비율에 대한 결과이다. 그림 4에서 보면 제안

한 알고리즘이 하나의 메트릭을 사용해서 계산한 shortest path first(SPF)나 Widest-shortest (WS), Shortest-widest(SW) 등의 다른 알고리즘보다 거절 비율이 낮은 편이다. 이 결과는 모든 노드에서 랜덤하게 콜이 발생하여 어느정도 망에서 균일하게 트래픽이 분포된 결과이다. 이 결과는 일부 노드에 부하가 몰렸을 경우 성능 결과에서 차이를 보인다. 그림 5는 일부 노드에 부하가 있어서 트래픽이 균등하게 네트워크에 분포되지 않았을 때의 결과를 보여주고 있다. 다른 노드들 보다 특정 노드들에서 트래픽이 많이 발생하여 일부 노드에 트래픽이 많이 몰렸을 경우의 결과이다. 결과 그래프를 살펴보면 그림 4에서보다 그림 5의 결과에서 제안한 알고리즘 성능이 매우 좋음을 알 수 있다.

그림 5는 일부 링크에 부하의 50%를 집중 발생시킨 것이다. 초기에 트래픽이 적었을 때보다 적당히 네트워크에 트래픽이 몰릴 경우 성능차이가 확연하게 드러나는 것을 알 수 있다. 그러나 네트워크 전체에 트래픽이 너무 많이 발생하게 되면 어느 알고리즘이든 성능을 발휘할 수 없게 된다. 발생된 콜에 대해 거절율은 제안한 알고리즘과 shortest-widest가 비슷한 값을 유지하게 되는 것을 볼 수 있지만, 그림 6에서 보인 선택한 경로의 길이를 비교해보면 제안한 알고리즘은 최단 경로를 사용하는 shortest path first와 widest-shortest 알고리즘처럼 짧은 경로를 사용하다가 네트워크에 부하가 생기면 우회를 하는 먼 경로를 사용한다는 것을 알 수 있으나, shortest-widest는 처음부터 먼 경로를 사용하기 때문에 만약 경로 지연이 50ms였다면 모든 콜이 거절되었을 것이다.

우회 경로가 많지 않은 망구조 실험에서는 우회해 갈 수 있는 경로가 적어서 다양한 경로를 선택하지 못했기 때문에 제안한 알고리즘과 다른 알고리즘과의 성능 차이가 크게 나타나지 않았다.

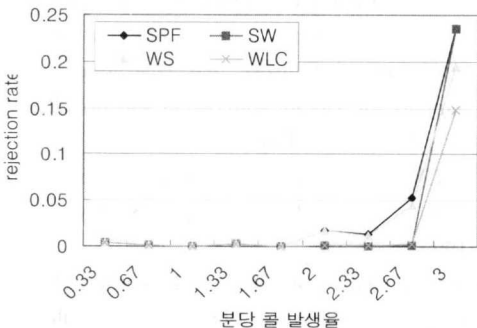


그림 4. 토폴로지 2에서 콜 거절 비율

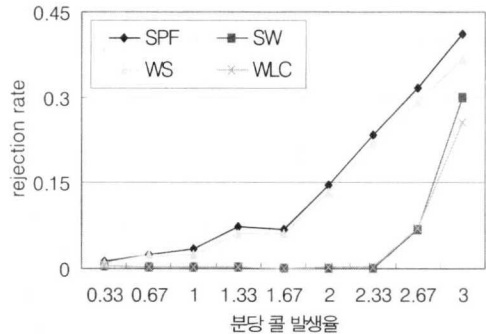


그림 5. 토폴로지 2에서 일부 노드에 부하가 있을 때 콜 거절 비율

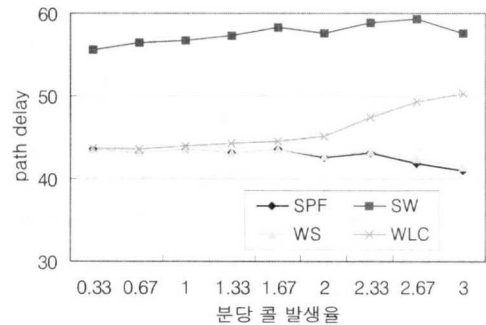


그림 6. 토폴로지 2에서 선택된 경로의 평균 지연시간

실험 결과에서 보면 모든 노드에서 일정하게 트래픽이 발생했을 경우 한쪽으로 트래픽이 치우는 결과가 흔하게 발생되지 않으므로 성능의 개선 효과를 기대하기 어려운 것이다. 그러나, 만약 한쪽으로 트래픽이 많이 몰리는 상황에서의 결과는 다르게 나타난다. 각각의 토폴로지에서도 모든 노드가 균등하게 트래픽을 발생시키는 것이 아니고 일부 노드, 일부 링크에서 트래픽이 몰리게 되는 경우 그 결과는 다르다. 제안하는 알고리즘은 최대 대역만을 찾는 shortest-widest 알고리즘 만큼 거절 비율이 낮아지면서도 경로는 가능한 최선의 경로를 찾게 되는 것이다.

실험에서 보면, 연결요청이 균일한 망에서보다 연결이 한쪽으로 몰려서 망의 효율이 떨어진 상황에서 보다 좋은 성능을 나타냈다. 또한 우회경로나 대체 경로가 많은 토폴로지 네트워크 구조에서 특히 좋은 성능을 나타냈으며, 우회경로가 존재하나 그 수가 많지 않은 경우 기존의 알고리즘보다 개선은 있었으나 큰 성능 향상을 기대할 수 없었다.

이러한 실험에서 확인할 수 있었던 것은 트래픽이 한쪽으로 몰려서 네트워크 효율이 좋지 않은 경

우에 제한한 알고리즘이 보다 좋은 성능결과를 나타내었으며, 평소와 같은 환경에서도 다른 어느 알고리즘보다 성능이 떨어지지 않았다.

V. 결론

QoS를 보장받으려 하는 트래픽에 대해서 widest-least cost 알고리즘이라는 새로운 라우팅 알고리즘을 제안하였으며 다른 알고리즘과 비교하였다. 제한한 알고리즘은 네트워크 부하가 낮을 경우 widest-shortest 알고리즘으로 동작하고, 네트워크 부하가 높을 경우 widest-least cost로 동작함으로써 다른 알고리즘보다 좋은 성능을 보여주고 있다. 우회경로가 많은 네트워크 구조에서 그 우회경로를 효과적으로 사용하게 하는 알고리즘이며, 트래픽이 일부 링크에 몰렸을 경우 좋은 성능을 보였으며, 링크 사용량에 따라 다른 링크를 사용하는 것이 가능하게 하는 점은 트래픽을 덜 복잡한 링크로 옮김으로써 네트워크에 부하를 분산시키는 traffic engineering 역할도 수행한다. 이는 또한 콜 거절을 줄여주게 되며, 트래픽 부하 배분과 트래픽 혼잡을 피하는데 도움이 된다.

참 고 문 헌

[1] D.Bertsekas and R.Gallager, Data Networks, Prentice Hall, 1992.
 [2] C.Huitema, Routing in the Internet, Prentice Hall, 1999.
 [3] W.C.Lee, M.G. Hluchyi, and P. Humblet, "Routing Subject to Quality of Service Constraints in Integrated Communication Networks", *IEEE Network*, pp.46-55, July/August, 1995.
 [4] R.Vogel, R.Herrtwich, W.Kalfa, H.Wittig, and L.Wolf, "QoS-Based Routing of Multimedia Streams in Computer Networks", *JSAC*, vol 14, no.7, September 1996.
 [5] Z.Wang and J.Crowcroft, "Quality-of-Service Routing for Supporting Multimedia Applications", *IEEE Journal on Selected Areas in Communications*, Vol. 14, No. 7, Sep., 1996.
 [6] R.Guerin, A.Orda, and D.Williams. "QoS Routing Mechanisms and OSPF Extensions", Internet draft, Nov., 1996.

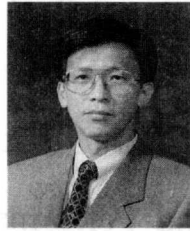
[7] Zhang, Sanchez, Salkewicz, and Crawley, "Quality of Service Extensions to OSPF", Internet draft, Sep., 1997.
 [8] H.Salama, D. Reeves, and Y.Viniotis, "A Distributed Algorithm for Delay -Constrained Unicast Routing", *IEEE INFOCOM 97*.
 [9] S.Murthy, and J.Garcia-Luna-Aceves, "Loop-Free Internet Routing Using Hierarchical Routing Trees", *IEEE INFOCOM 97*.
 [10] Q.Ma, and P.Steenkiste, "On path Selection for Traffic with Bandwidth Guarantees", *IEEE International conference on Network Protocols*, Oct., 1997.
 [11] R. Guérin, and A. Orda, "QoS-based Routing in Networks with Inaccurate Information: Theory and Algorithms", *INFOCOM97*, pp.75-83, 1997.
 [12] D.Bertsekas, Network Optimization: Continuous and Discrete Models, Athena Scientific, 1998.
 [13] Eric Crawley, Raj Nair, Bala Rajagopalan, and Hal Sandick, "A framework for QoS-based Routing in the Internet", IETF RFC 2386, August 1998.
 [14] N.Rao and S.Batsell, "QoS Routing Via Multiple Paths Using Bandwidth Reservation", *IEEE INFOCOM98*, March 1998.
 [15] A. Orda, "Routing with End to End QoS Guarantees in Broadband Networks", *INFOCOM 98*, pp.27-34, 1998.
 [16] D. Lorenz, and A. Orda, "QoS Routing in Networks with Uncertain Parameters", *INFOCOM 98*, pp.3-10, 1998.
 [17] Q. Ma and P. Steenkiste, "Routing Traffic with Quality-of-Service Guarantees in Integrated Services Networks", *The 8th International workshop on Network and Operating Systems Support for Digital Audio and Video(NOSSDAV 98)*, Cambridge, England, July 1998.
 [18] X.Xiao, and L. Ni, "Internet QoS: A Big Picture", *IEEE Network*, pp.8-18, March/April, 1999.
 [19] Q. Ma and P. Steenkiste, "Supporting Dynamic Inter-Class Resource Sharing : A Multi-Class QoS Routing Algorithm", *INFOCOM99*, pp.649-

660,1999.

- [20] A.Orda, "Routing with End-to-End QoS Guarantees in Broadband Networks", *IEEE/ACM Transactions on Networking*, Vol.7, No.3, June 1999.
- [21] G.Apostolopoulos, R.Guerin, S.Kamat and S.K. Tripathi, "Server Based QoS Routing", *Globecom99*, pp.762-768, 1999.
- [22] S.Vutukury and J.J.Garcia-Luna-Aceves, "A Distributed Algorithm for Multipath Computation", *Globecom99*, pp.1689-1693, 1999.
- [23] T.Korkmaz and M.Krunz, "A Randomized Algorithm for Finding a Path Subject to Multiple QoS Constraints", *Globecom99*, pp.1694-1698, 1999.
- [24] B.Fortz and M.Thorup, "Internet Traffic Engineering by Optimizing OSPF Weights", *INFOCOM2000*, 2000.
- [25] F.Hao and E.Zegura, "On Scalable QoS Routing: Performance Evaluation of Topology", *INFOCOM2000*, 2000.

이 재 용(Jai-yong Lee)

정회원



1977년 2월 : 연세대학교
전자공학과 졸업
1984년 5월 : IOWA State
University 공학 석사
1987년 5월 : IOWA State
University 공학 박사

1987년 6월~1994년 8월 : 포항공과대학 교수
1994년 9월~현재 : 연세대학교 전자공학과 교수
<주관심 분야> Protocol design for QoS management, network management, high speed networks and conformance testing

김 동 연(Dong-Yeon Kim)

1986년 2월 : 연세대학교 전자공학과 졸업.
1988년 2월 : 연세대학교 전자공학과 석사.
1995년 2월 : 연세대학교 전자공학과 박사.
1988년~1996년 : (주)데이콤 과장.

1996년~현재 : 국립한경대학교 전자공학과 교수.
<주관심 분야> PCS Management: Location Management/Mobility Management, Satellite ATM Network

전 한 얼(HahnEarl Jeon)

정회원



1995년 2월 : 연세대학교
전자공학과 졸업
1997년 8월 : 연세대학교
전자공학과 석사
~현재 : 연세대학교
전기, 전자공학과 박사

<주관심 분야> QoS Routing , Traffic Engineering

김 영 준(Young-Joon Kim)

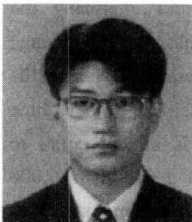
정회원



1994년 2월 : 한양대학교
전자통신공학과 졸업
1997년 8월 : 한양대학교
전자통신공학과 석사
1996년 1월~현재 : LG 전자
핵심망연구소 이동통신
연구실

<주관심 분야> 이동통신 , 교환시스템 설계

김 성 대(Sung-Dae Kim)



2000년 2월 : 연세대학교
전기전자공학과 졸업
2002년 2월 : 연세대학교
전기전자공학과 석사
~현재 : LG 전자 GSM
단말 연구소

<주관심 분야> MPLS, 이동통신