

전송지연을 적용한 적응균등부하조절 경로설정 알고리즘

正會員 주 만 식*, 백 이 현*, 주 판 유*, 강 창 언*

Adaptive Routing Algorithm for Equitable Load Balancing with Propagation Delay

Man-Sik Joo*, Lee-Hyun Baek*, Pan-Yuh Joo*,
Chang-Eon Kang* *Regular Members*

요 약

본 논문에서는 다중경로가 가능한 고속망에서 병목을 감소시키며 또한 평균지연을 감소시키는 새로운 경로설정 방법을 제안한다. 기존의 적응균등부하조절 경로설정기법과 경로의 거리에 비례하는 전송지연을 비용의 개념으로 고려하는 경로설정기법이 결합된 이 알고리즘은 계산에 드는 시간이 적고 효율적인 동적 트래픽 할당을 하는 알고리즘이다. 사용자의 트래픽 형태가 바뀔 때 따라 각 링크의 전송용량에 비례하면서, 전송지연이 적은 경로로 적절하게 트래픽을 할당하도록 적응성 있게 경로를 설정한다. 그렇게 함으로써 병목을 줄이고 최단 링크의 사용도를 높여 망의 평균지연을 감소시키게 된다. 이는 다른 경로설정기법보다 고 품질의 서비스를 사용자에게 제공하는 기반이 된다. 그리고 제안된 알고리즘을 시뮬레이션을 통해서 저부하에서 고부하상태까지 기존의 적응균등부하조절 경로설정기법보다 평균지연이 감소됨을 보인다.

ABSTRACT

In this paper, a routing algorithm is proposed in order to reduce average end-to-end delay and congestion in the high speed network. The algorithm proposed here uses the existing one which adaptively modifies routes and the amount of traffic allocated to each link as user traffic patterns fluctuate. This algorithm is ELB(Equitable Load Balancing). Also, the new algorithm considers the propagation delay proportional to the distance between source and destination. It reduces congestion from the ELB and average end-to-end delay from the propagation delay concepts respectively. Through the simulation, it shows that the algorithm proposed here reduces average end-to-end delay over low load to high load, and it also guarantees the congestion control.

*연세대학교 전자공학과
論文番號:96022-0118
接受日字:1996年 1月 18日

I. 서 론

광학 전송 매체의 발전은 고속의 전송을 가능하게 하였고 그로 인한 속도와 지연의 비를 변화시켰다. 또한 컴퓨터 기술의 발전은 데이터 통신망이 더욱 광범위하고 다양한 사용자의 요구사항을 수용하도록 하고 있다.

데이터 통신망은 ATM을 근간으로 하는 광대역 종합 정보 통신망(B-ISDN)으로 발전해 가고 있으며 고도의 지능망(Advanced Intelligent Network)으로 발전해 가고 있다. 일반적으로 ATM과 같은 망에서는 각 트래픽에서 요구되는 QOS(Quality of Service)가 중요하며 이에 맞는 서비스를 제공해 주어야 한다. 즉, 음성과 비디오와 같은 정보는 지연에 민감하며 파일 전송과 전자 사서함과 같은 정보는 손실에 민감하다. 또한, 상호 교환적인 그래픽스나 계산의 응용 같은 정보는 지연과 손실 모두에 민감하다.

ATM과 같은 고속망에서는 광섬유를 사용하며 이 매체는 5 μ s/km의 전송지연을 갖게 된다. 이러한 망에서는 패킷을 고속으로 전송하게 되므로 노드간의 거리에 따른 전송지연을 무시할 수 없게 된다. 즉 전송지연이 적은 경로에 많은 패킷을 할당하고 전송지연이 긴 쪽에 적은 패킷을 할당하면 평균 시간은 줄어들게 된다. 하지만 지연을 근거로 하는 경로설정은 망의 병목을 피할 수 없게 된다. 유한한 버퍼를 가지고 있는 고속망에서는 한번 병목이 발생하면 많은 셀의 손실을 유발시키므로 적절한 정도의 부하를 유지시키는 것이 중요하다. 반면에 병목을 근거로 하는 경로설정은 지연 문제를 해결할 수 없고 망자원의 낭비를 발생시킬 수 있다. 그러므로 고속망에서는 셀손실과 지연 모두에 근거한 적절한 경로설정 알고리즘이 필요하다.

기존의 적응균등부하조절(Equitable Load Balancing: ELB) 알고리즘[3]은 전송지연이 적용되어 있지 않다. 단지 전송 용량만을 고려하게 되는 것이다. 한 링크를 공유하는 양종단간의 노드들이 많을수록 그 링크가 하나의 양종단간 노드에 대해 가장 짧은 경로가 되어도 이 링크에 많은 패킷을 할당할 수는 없게 된다. 그러므로 각 링크의 용량을 넘어서지 않는 한도 내에서 전송지연을 적용하면 양종단간의 짧은 경로에 더 많은 패킷을 할당 할 수 있게 된다.

본 논문에서는 전송용량을 고려한 ELB 알고리즘에 전송지연을 고려한 부분을 추가하여 전체적인 평균 지연을 개선한다.

본 논문의 구성은 다음과 같다. 제 2 장에서는 기존의 적응균등부하조절 알고리즘을 서술한다. 제 3 장에서는 본 논문에서 제안하는 전송지연을 적용한 적응균등부하조절 알고리즘에 대해 논하며, 제 4 장에서는 일반적인 망에 대한 실험 모델을 제시한다. 그리고 제 5 장에서는 제안된 알고리즘에 대해서 그 효과 및 성능을 분석하며 제 6 장에서 결론을 맺는다.

II. 적응균등부하조절(ELB) 경로설정 알고리즘

2.1 트래픽의 성질

포와송 프로세스는 가장 간단하고 유용한 통계적 프로세스중의 하나이다. 이 프로세스는 하나의 구간에서 발생하는 사건의 수를 세는 프로세스이다. 구간 t 에서 n 개의 사건이 발생할 확률 $P_n(t)$ 는 식(1)과 같다.

$$P_n(t) = \frac{(\lambda t)^n}{n!} e^{-\lambda t} \quad (1)$$

식 (1)은 변수 λt 를 갖는 포와송 분포이다. 여기서 λt 는 구간 t 에서의 평균 사건발생 수를 나타낸다. 이 경우 사건 사이의 시간 분포는 지수적인 분포를 갖는다. 포와송 분포와 지수분포는 'memoryless' 성질 때문에 구별되는 구간들에서의 사건의 수가 단지 그 구간의 길이에 의존한다. 즉 다음의 사건에 대한 시간은 바로 전에 사건이 발생한 시간과는 독립적이다. 이러한 성질은 망에서 발생하는 패킷의 발생 성질과 유사하므로 발생하는 패킷의 성질을 포와송 분포로 모델링할 수 있다.

2.2 ELB 알고리즘

망은 N 개의 노드와 링크들로 구성되어 있다고 가정한다. 각 노드들은 수개의 입력과 수개의 출력을 가지고 있다. 그리고 각 노드는 이를 통과하는 패킷들에 대한 버퍼를 가지고 있다. 각 용량과 트래픽율은 단위 시간당 패킷으로 나타낼 수 있다. $\Lambda = \{\Lambda(s, d)\}$ 를 종단간의 트래픽 매트릭스라고 하자. 시간적으로 변하는 트래픽 형태는 연속적인 트래픽 매트릭스 $\Lambda_k(k = 1, 2, \dots)$ 로 표현될 수 있다. 제어를 효과적으로 하기

위해서는 시간간격 $t_k - t_{k-1}$ ($k=1, 2, \dots$)이 충분해서 제어 알고리즘을 적용하고 망의 트래픽 형태가 다시 바뀌기 전에 의미 있는 상태에 이를 수 있어야 한다.

적용 ELB 알고리즘은 사용자의 요구사항을 만족시키면서 비용 함수를 최소로 하도록 온라인으로 경로와 전송 용량을 수정하여 이를 통신망에 제공하여 준다. 이는 사용자에 의해서 요청되거나 시스템에 의해서 감지된 입력 트래픽 형태에 변화가 생길 때 동작한다. ELB에서는 새로운 통신 경로(s :source, d :destination)에 대한 요구가 발생하면 먼저 가용한 중간 링크(i, j)들이 선택된다. 그리고 나서 ELB의 비용 함수를 근거로 해서 온라인으로 계산되는 $R(i, j, s, d)$ 의 값을 이용하여 (s, d)에 대한 각 링크에 트래픽을 할당하게 된다. 즉 ELB 알고리즘의 목적은 각 링크의 용량에 비례하도록 부하를 분포시키는 것이다. 이 경우 사용되는 비용 함수는 다음과 같다.

$$K = \sum_i \sum_j \frac{1}{2} \left[p(i, j) - \frac{c(i, j)}{r(i)} \right]^2 \quad (2)$$

자신의 비용 함수를 사용하게 되지만 ELB를 표현하기 위해서 몇몇 다른 적절한 미분 함수를 사용할 수도 있다. $c(i, j)/r(i)$ 는 노드 i 의 출력 용량에 관한 링크(i, j)의 부하 용량을 표현한다. 링크(i, j)를 취하는 노드 i 를 떠나는 트래픽의 부분인 $p(i, j)$ 는 제어에 의해 수정되는 양이다. 즉, $p(i, j)$ 는 경로설정 결정 인자인 $R(i, j, s, d)$ 에 의해 수정된다. K 는 링크(i, j)를 흐르는 트래픽의 부분인 $p(i, j)$ 가 상대적인 링크 용량 $c(i, j)/r(i)$ 에 가까워야만 한 다는 것을 나타내는 함수이다.

통신노드쌍 (s, d)를 통과하는 트래픽은 링크 (i, j)가 이 경로에 할당되어 있으면 (i, j)를 사용할 수 있다. 즉, $c(i, j) \neq 0$ 이고 통신노드쌍 (s, d)를 통과하는 트래픽은 링크 (i, j)가 이 경로에 할당되어 있으면 (i, j)를 사용할 수 있다. 즉, $c(i, j) \neq 0$ 이고 $\delta(i, j, s, d) = 1$ 이면 링크 (i, j)를 사용할 수 있다. $\delta(i, j, s, d)$ 는 모든 양종단간 통신 노드쌍에 대해 양종단간의 연결성을 보장하며 어떤 (s, d)쌍에 대해서도 최소한 하나 이상의 경로가 존재하여야 한다는 것을 의미한다. 원천지에서 목적지로의 트래픽은 회귀되지 말아야 한다. 또 부등식 $\lambda(i, j) < c(i, j)$ 를 만족하여야 한다. 이 조건은 모든 링크와 노드가 포화되지 않는 것을 보장한다.

표 1. 사용되는 기호

(i, j)	노드 i 로부터 노드 j 로의 링크
$c(i, j)$	노드 i, j 간의 링크 용량
$\lambda(i, j)$	링크(i, j)에 할당되는 트래픽(단위 시간당패킷)
$\Lambda(i, j)$	노드 s 에서 노드 d 로 보내는 사용자의 트래픽
$\delta(i, j, s, d)$	전송 허가 $\delta(i, j, s, d) = 1$ 이면 s 에서 d 로 전송되는 트래픽이 링크(i, j)를 통과한다. $\delta(i, j, s, d) = 0$ 이면 s 에서 d 로 전송되는 트래픽이 링크(i, j)를 통과하지 못한다.
$R(i, j, s, d)$	s 와 d 를 원천지와 목적지로 하는 패킷이 링크(i, j)를 통과하는 비율
$p(i, j)$	노드 i 를 떠나는 모든 패킷중 j 노드로 들어가는 패킷의 비율

ELB 경로설정은 다음의 변수들과 제약 조건을 가지고 온라인 트래픽 경로설정을 만들고 수정한다. 즉 [1]과 같은 변수들이 주어지고 [4]의 제약 조건하에서 [3]을 이용하여 [2]를 최소화한다.

- [1] 토폴로지, $\delta = \{\delta(i, j, s, d)\}$, 링크 용량, 트래픽 매트릭스 Λ
- [2] 비용 함수: K
- [3] 경로설정 변수: $R(i, j, s, d)$
- [4] ① if $\delta(i, j, s, d) = 0$
 then $R(i, j, s, d) = 0 \quad i, j, s, d \in 1, \dots, N$
- ② if $c(i, j) = 0$
 then $R(i, j, s, d) = 0$ and $p(i, j) = 0$
 $i, j, s, d \in 1, \dots, N$
- ③ $0 \leq R(i, j, s, d) \leq 1$
 if $\sum_k R(k, i, s, d) = 0$
 then $\sum_j R(i, j, s, d) = 0$
 else $\sum_j R(i, j, s, d) = 1$
 $k, i, j, s, d \in 1, \dots, N$

제약 조건 [4]에서 ①과 ②는 경로 선택이 토폴로지와 망의 유용한 용량에 근거한다는 것을 의미한다. ③은 원천지 s 와 목적지 d 를 갖는 패킷이 노드 i 에 도착할 때 다음의 노드 중 하나로 보내진다는 것을 의미한다.

이 적용 알고리즘을 구성하는데 있어서 변수 R 의

비용 함수의 gradient를 계산한다. (S, D)는 특정 양종 단간 통신 노드쌍을 나타내며 (s, d)는 일반적인 쌍이다. 식 (2)로부터 다음과 같은 식을 쓸 수 있다.

$$\frac{\partial N}{\partial R(u, v, S, D)} = \sum_i \sum_j \sigma [p(i, j) - \frac{c(i, j)}{r(i)}] \frac{\partial p(i, j)}{\partial R(u, v, S, D)} \quad (3)$$

위의 계산을 위해서는 $\partial p(i, j) / \partial R(u, v, S, D)$ 가 필요하다. 그리고 다중 경로가 가능하다고 가정한다.

l_1, \dots, l_n 을 일련의 노드라 하고 이 노드들은 연속되는 링크 (s, l_1), (l_1, l_2), ..., (l_n, i)로 구성된다고 하자. 그러면 $p(i, j)$ 는 식(4)에서와 같이 i 에 연결된 노드 중 j 노드로 나가는 패킷들의 합과 각 경로들 중 i 노드로 들어가는 패킷의 전체의 합의 비로 나타내어진다.

$$p(i, j) = \frac{\sum_{(s,d)} \sum_{l_1} \dots \sum_{l_n} R(s, l_1, s, d) R(l_1, l_2, s, d) \dots R(l_n, i, s, d) R(i, j, s, d) \Lambda(s, d)}{\sum_{(s,d)} \sum_{l_1} \dots \sum_{l_n} R(l_n, i, s, d) \dots R(l_1, l_2, s, d)} \times \frac{R(l_n, i, s, d) R(i, j, s, d) \Lambda(s, d)}{R(s, l_1, s, d) \Lambda(s, d)} \quad (4)$$

$p(i, j)$ 에 대한 $R(u, v, S, D)$ 의 미분의 식은 식(5)와 같이 쓸 수 있다.

$$\frac{\partial p(i, j)}{\partial R(u, v, S, D)} = G(i, j, u, v, S, D) - p(i, j)H(i, j, u, v, S, D) \quad (5)$$

여기서 $G(i, j, u, v, S, D)$ 와 $H(i, j, u, v, S, D)$ 는 다음과 같이 정의된다.

$$G(i, j, u, v, S, D) = \begin{cases} 1 & i=u, j=v, s=S, d=D \\ \frac{R(i, j, s, d)}{R(u, v, s, d)} & s=S, d=D, (u, v) \text{는} \\ & (S, i) \text{사이의 라우트상에 있음} \\ 0 & \text{모든 다른 경우} \end{cases}$$

$$H(i, j, u, v, S, D) = \begin{cases} \frac{1}{R(u, v, s, d)} & s=S, d=D, (u, v) \text{는} \\ & (S, i) \text{사이의 라우트상에 있음} \\ 0 & \text{모든 다른 경우} \end{cases}$$

ELB 알고리즘은 트래픽의 변화에 따라 경로결정 매트릭스 $R_k = \{R_k(i, j, s, d)\}$ ($k=1, 2, \dots$)의 일련의 값들을 계산하여 망을 조절한다. 이 과정에서 비용 함수 K_k 를 최소화시킨다. K_k 를 최소화하기 위해 gradient descent 알고리즘을 사용하였으며 이는 절대적인 최소의 값으로 수렴하지는 않지만 비용 함수 K 를 항상 감소시킨다. 변수값들에 대해서 gradient descent 반복 계산을 중지시키는데 사용되는 수용할 수 있는 에러 한계 내에서 국부 최소점(local minimum)에 이르게 된다. 각각의 새로운 입력 형태 Λ_k 에 대해서 경로결정 매트릭스 R_k 의 k 번째 값은 비용 함수 K_{k-1} 의 gradient descent를 근거로 해서 계산된다. 먼저 R_k 를 R_{k-1} 로 한다. 그리고 두개의 연속되는 반복계산에 의한 비용 함수 K_k 가 ϵ 보다 작을 때까지 다음 식(6)을 이용해서 계산한다.

$$R_k(u, v, S, D) \leftarrow R_k(u, v, S, D) - \eta \frac{\partial K}{\partial R(u, v, S, D)} \Big|_k \quad (6)$$

이때 앞에서 정의한 제약 조건들을 만족하여야 한다. $\epsilon > 0$ 은 반복 계산을 중지시키기 위한 에러 한계이며, $\eta > 0$ 은 gradient descent의 속도를 결정하는 상수이다. 큰 값의 η 를 쓰면 gradient descent의 수렴 속도는 빨라지지만 하나 또는 몇 개의 국부 최소점 주위에서 진동할 수 있다. 반면, 작은 η 값은 알고리즘이 천천히 수렴하도록 한다. 여기서는 적용 gradient descent 속도의 개념을 사용한다. 즉, 계산 과정 동안에 조건 $||K|_{k, new} - |K|_{k, old}|| < \epsilon$ 을 만족하면 점차적으로 η 를 감소시키는 것이다.

$\Lambda(S, D) \neq 0$ 인 양종단간 통신 노드쌍을 \simeq 라 할 때 마다 망의 갱신은 원천지 노드에서부터 시작되어야 한다. 어떤 노드 u 에서 경로설정 갱신이 \simeq 되었으면 다음에 갱신되는 노드 v 에 대해서 링크 (u, v) 의 일부분이어야 하며 모든 이웃하는 v 노드들은 이미 갱신되어져 있어야 한다.

다음은 ELB 알고리즘의 적용 과정을 단계적으로 정리한 것이다.

① 입력 값 Λ_k 와 δ_k 를 취하고 $R_k = R_{k-1}$ 로 한다.

② 식 (4)를 이용 $p(i, j)$ 를 계산하고, $K_{k, old}$ 를 계산한다.

③ R_k 를 갱신하기 위해 식 (5)를 이용하여 식 (6)을 반복 계산한다. 그리고 $K_{k, new}$ 를 계산한다. 만약 $||[K]k, new - [K]k, old|| < \epsilon$ 이면 중지한다. 아니면 ②로 간다.

망의 제어 알고리즘에 대한 중요한 부분 중의 하나는 계산의 복잡도이다. 매우 복잡한 알고리즘은 시간이 많이 걸려서 망의 성능을 저하시키며 복잡하고 비싼 프로세서를 필요로 하게 된다.

집중화된 적응 제어 알고리즘에서의 계산은 거의 단계 ②와 ③에서 이루어진다. ②는 알고리즘의 각 반복 계산에 대해 한번씩 계산되어야 한다. ②를 계산하는 것에 대한 복잡도는 망의 토폴로지에 달려 있다. l 을 망에 존재하는 링크의 수라하고, E 는 하나의 양종단간 통신 노드쌍에 대한 다중 경로의 최대수라고 하자. M 은 동작중인 양종단간 통신 노드쌍의 수이고, L 은 하나의 경로에 대한 링크의 최대의 수이다. $L \leq N-1$ 이므로 (2)식의 $p(i, j)$ 의 계산은 $O(EMN)$ 이상의 시간 복잡도를 갖지 않게 된다. 실제로 E 는 작은 정수(2 혹은 3)의 값을 갖게 되므로 (2)의 계산량은 $O(IMN)$ 의 값을 갖게 된다. Gradient Descent에 사용되는 미분식은 $p(i, j)$ 의 계산량과 비슷하므로 ELB 알고리즘은 $O(IMN)$ 의 시간 복잡도를 갖게 되고 이 정도의 계산 시간은 매우 합리적인 수치이다.

Ⅲ. 전송지연을 적용한 적응 ELB 알고리즘

전송지연은 전송매체의 빛의 속도에 의해 발생되며 원천지와 목적지 사이의 거리에 의존하게 된다. 전송지연을 적용하기 위해서는 이 거리를 비용으로 고려한다. 전송지연에 따른 비용 매트릭스를 D_q 라 하자.

양종단간의 노드쌍에 대해 M_a 개의 유용한 경로가 존재한다고 가정하면 경로 q 를 선택하는 확률은 다음과 같다.

$$P_q = \frac{C_0}{D_q}, \quad C_0 = \left(\sum_{q=1}^{M_a} 1/D_q \right)^{-1} \quad (7)$$

식 (7)은 경로의 거리에 반비례하게 패킷을 할당하는 것을 의미한다. 이것은 가장 짧은 경로가 다른 경로보다 더 큰 확률로 선택된다는 것을 의미하며, 일반적인 경로설정보다 훨씬 좋은 성능을 얻을 수 있다.

확률 P_q 값의 결정은 다음과 같다. 먼저 특정 (s, d) 쌍이 정해지면 이에 유용한 다중경로에 대한 홉(hop)수에 반비례하도록 원천지 노드에서부터 각 경로를 선택할 확률 P_q 를 결정한다. 이렇게 원천지 노드에 대한 P_q 값들이 결정되면 각 경로의 다음 노드에 대해서 경로의 분기여부를 확인하고 분기가 되면 각 경로에 따라 다시 목적지까지의 홉수를 계산하여 노드수가 짧은 경로에 높은 경로 선택 확률을 준다. 위와 같이 목적지 노드 전까지에 할당되는 경로선택 확률 P_q 값들을 결정하게 되며 이 절차를 모든 (s, d) 쌍에 대해 수행한다. 여기서 구해진 경로설정 변수를 Q 매트릭스라 하자.

실제 경로 설정시 트래픽량이 적은 경우에는 $Q(i, j, s, d)$ 의 값을 사용하고, 트래픽량이 증가하여 $Q(i, j, s, d)$ 에 따른 각 링크의 용량이 초과하는 경우 ELB 알고리즘에서 이용되는 R 매트릭스와 Q 매트릭스의 각 링크에 대한 $R(i, j, s, d)$ 와 $Q(i, j, s, d)$ 와의 차를 고려하여 R 매트릭스 경로설정 변수 쪽으로 옮겨간다. 이렇게 함으로써 전송지연과 ELB 알고리즘의 이점을 모두 살린 경로 설정을 구현할 수 있다.

이 때 Q 경로설정 변수를 R 경로설정 변수로 접근시킬 경우 Q 변수들과 R 변수들의 차이를 나누는 구간이 너무 많으면 계산량이 많아짐은 물론, 적절한

표 2. 전송지연을 적용한 적응 ELB 알고리즘

- ① 전송 지연 즉 거리 비용을 적용하여 Q 매트릭스를 구한다.
- ② 먼저 Q 매트릭스를 적용하여 전송 용량의 초과 여부를 확인한다. 초과하지 않는 경우 망에 그대로 적용하며 초과할 경우 ③으로 간다.
- ③ ELB 알고리즘을 적용하여 R 매트릭스 값을 구한다.
- ④ 각 $R(i, j, s, d)$ 와 $Q(i, j, s, d)$ 의 차를 구한다.
- ⑤ ④에서 구한 차를 일정 구간으로 나누어 Q 매트릭스에서 R 매트릭스로 그 값을 접근시킨다. 이 과정 중 전송 용량의 초과 여부를 확인하여 초과하지 않는 경우 여기서 구해진 경로 설정 변수를 망에 적용하며 초과 시에는 ⑤를 반복한다.

경로설정 변수값으로 접근하는데 시간이 많이 걸리게 되며, 구간이 너무 적으면 효율적인 경로설정 변수를 찾지 못해 효율이 떨어지게 된다. 이는 망에 인가되는 트래픽의 량에 따라 적절히 잡아 주어야 한다.

다음 표 2에 본 논문에서 제안한 전송지연을 적용한 적용 ELB 알고리즘들을 정리하였다.

이 알고리즘들은 계산 복잡도 면에서 계산량이 크게 증가하지는 않는다. 만약 망내에 존재하는 모든 경로설정 변수를 n 개라 하고 R_k 와 Q_k 값의 차를 m 개의 구간으로 나눈다고 가정하면 이 계산량은 $n \times m$ 이므로 큰 계산량을 요구하지 않는다.

IV. 실험 모델

실험 모델은 패킷(셀)스위칭망과 적용 제어 시뮬레이터로 구성된다. 두 부분은 그림 1에서와 같이 'trigger'라는 것을 통해 상호 동작하며 이 'trigger'는 망의 외부에서 인가되는 트래픽이 망의 상황에 영향을 미치는 정도로 변할 경우 적용 알고리즘들을 동작하게 하는 기능을 수행한다.

그리고 SLAM이라는 망 시뮬레이터를 사용하여 실험하였으며, 트래픽량이 시간적으로 변한다는 조건으로 측정하였다. 이 변화를 측정하기 위해 간단한 트래픽 측정 기능을 시뮬레이터에 부가하였다. 즉, 'trigger'에서 제어 시뮬레이터를 동작시키면 ELB 알고리즘과 여기에 전송지연을 적용한 제안 알고리즘이 동작을 하여 새로운 경로설정 변수를 만들어 내고 이를 망 시뮬레이터로 보낸다. 망 시뮬레이터는 이 경로설정 변수를 망에 적용하여 평균 전송지연을 측정하고 계속해서 트래픽량을 측정하여 'trigger'의 동작여부를 확인한다.

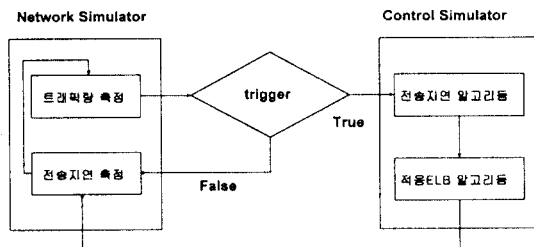


그림 1. 실험 모델

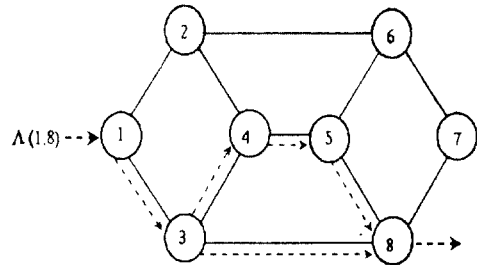


그림 2. 실험망 구조

표 3. 사용 허가 매트릭스

링크	(s, d)	사용 허가 매트릭스			
		(1 8)	(2 7)	(6 3)	(8 2)
(1 2)	1	0	0	1	
(1 3)	1	1	1	0	
(2 1)	0	1	1	0	
(2 4)	0	1	1	0	
(2 6)	1	1	0	0	
(3 1)	0	0	0	1	
(3 4)	1	0	0	1	
(3 8)	1	1	0	0	
(4 2)	0	0	0	1	
(4 3)	0	0	1	0	
(4 5)	1	1	0	0	
(5 4)	0	0	1	1	
(5 8)	1	1	0	0	
(6 2)	0	0	1	1	
(6 5)	1	1	1	0	
(6 7)	1	1	0	0	
(7 6)	0	0	0	1	
(7 8)	1	0	1	0	
(8 3)	0	0	1	1	
(8 5)	0	0	0	1	
(8 7)	0	1	0	1	

표 4. 초기 최단 경로

Source-to-Sink Pair	Route which is Initially Chosen
(1, 8)	1-3-8
(2, 7)	2-6-7
(6, 3)	6-5-4-3
(8, 2)	8-7-6-2

SLAM에서는 각 링크마다 무한대의 버퍼 용량을 가지는 임의의 망 토폴로지를 사용할 수 있다.

그림 2는 본 논문에서 실험한 망의 구조이다. 이 망

은 8개의 노드로 구성되어 있고 통신에 참여하는 노드쌍 (s, d)은 (1,8), (2,7), (6,3), (8,2) 네 개의 경우로 가정하였다. 그리고 표 3에 표시된 이 망에 대한 '사용-허가' 매트릭스 $\delta(i, j, s, d)$ 를 근거로 하였다.

입력 트래픽은 포와송 분포에 따라 발생시킨다. 처음 각 (s, d)쌍에 대해 망이 초기화 될 때는 경로가 표 4에 주어진 하나만이 주어진다. 그러나 입력 트래픽이 계속적으로 발생함에 따라 다른 경로를 사용할 수 있게 된다.

실험에 사용되는 각 링크의 전송 용량은 초당 2.4×10^5 패킷으로 동일하게 가정하였다. 망 전체에 인가되는 트래픽량은 초당 Λ 패킷이며 망에 존재하는 모든 통신 노드쌍에 동일하게 나누어 인가한다. 이 시뮬레이션에서는 4개의 통신 노드쌍이 존재하므로 각 쌍에 초당 $\Lambda/4$ 패킷을 인가한다. 그리고 이 트래픽량 Λ 를 변화시켜가며 각 통신 노드쌍에 대해 총 2×10^6 패킷을 통과시켜 패킷의 평균 전송 지연시간을 측정하였다.

이 시뮬레이션에서는 최종 수신단에서의 패킷 재순서화에 걸리는 시간과 각 노드의 처리 시간은 무시하였다. 이는 종단 시스템의 성능에 중속적이며 망내에서만 전송지연을 측정하기 위함이다. 그리고 이 알고리즘의 결과를 망의 각 노드와 송수신하는데 걸리는 시간 또한 무시하였다. 이는 모든 중앙집중화된 경로설정 기법이 갖는 문제이기 때문이다. 각 노드의 버퍼의 용량은 무한대로 한다. 만일 버퍼의 용량이 유한할 경우 큐잉이 발생하면 패킷의 손실이 발생할 수 있으므로 전체 패킷에 대한 평균 전송지연을 계산할 수 없기 때문이다.

V. 실험 결과 및 고찰

일반적인 망에서 거리 즉, 각 통신 노드쌍의 홉(hop)수에 비례하는 지연을 비용의 개념으로 적용한 알고리즘을 적용 ELB 알고리즘에 결합시킨 새로운 알고리즘에 대한 결과를 컴퓨터 시뮬레이션을 통하여 알아본다. 그림 2의 망에서 통신 노드쌍들에 대한 가능한 경로와 각 경로에 해당하는 홉수는 표 5와 같다. 이 홉수 D_q 는 한 노드에서 그 노드에 연결된 통신 가능한 경로들을 선택하는 확률인 P_q 를 구하는데 사용되어지며 여기서 구해진 각 링크에 대한 경로 설정

표 5. 가능한 경로 및 홉수

(S, D)	가능한 경로수	총 홉수
(1, 8)	1-2-6-5-8	4
	1-2-6-7-8	4
	1-3-4-5-8	4
	1-3-8	2
(2, 7)	2-6-7	2
	2-1-3-8-7	4
	2-4-5-8-7	4
	2-6-5-8-7	4
(6, 3)	6-2-1-3	3
	6-2-4-3	3
	6-5-4-3	3
	6-7-8-3	3
(8, 2)	8-3-1-2	3
	8-3-4-2	3
	8-5-4-2	3
	8-7-6-2	3

표 6. 각 링크에 대한 Q 경로설정 변수

(1, 8)	(2, 7)
Q(1, 2, 1, 8)=0.4	Q(2, 1, 2, 7)=0.2
Q(1, 3, 1, 8)=0.6	Q(2, 4, 2, 7)=0.2
Q(3, 4, 1, 8)=0.25	Q(2, 6, 2, 7)=0.6
Q(3, 8, 1, 8)=0.75	Q(6, 5, 2, 7)=0.25
Q(2, 6, 1, 8)=1.0	Q(6, 7, 2, 7)=0.75
Q(4, 5, 1, 8)=1.0	Q(1, 3, 2, 7)=1.0
Q(5, 8, 1, 8)=1.0	Q(3, 8, 2, 7)=1.0
Q(6, 5, 1, 8)=0.5	Q(4, 5, 2, 7)=1.0
Q(6, 7, 1, 8)=0.5	Q(5, 8, 2, 7)=1.0
Q(7, 8, 1, 8)=1.0	Q(8, 7, 2, 7)=1.0
(6, 3)	(8, 2)
Q(6, 2, 6, 3)=0.5	Q(8, 3, 8, 2)=0.5
Q(6, 5, 6, 3)=0.25	Q(8, 5, 8, 2)=0.25
Q(6, 7, 6, 3)=0.25	Q(8, 7, 8, 2)=0.25
Q(1, 3, 6, 3)=1.0	Q(3, 1, 8, 2)=0.5
Q(4, 3, 6, 3)=1.0	Q(3, 4, 8, 2)=0.5
Q(5, 4, 6, 3)=1.0	Q(5, 4, 8, 2)=1.0
Q(2, 1, 6, 3)=0.5	Q(1, 2, 8, 2)=1.0
Q(2, 4, 6, 3)=0.5	Q(4, 2, 8, 2)=1.0
Q(8, 3, 6, 3)=1.0	Q(6, 2, 8, 2)=1.0
Q(7, 8, 6, 3)=1.0	Q(7, 6, 8, 2)=1.0

매트릭스 Q는 표 6과 같다.

먼저 전송 지연을 나타내는 Q 경로설정 변수 값만을 이용하여 경로를 선택하는 경우와 적응 ELB 알고리즘을 사용하여 경로를 선택하는 경우를 시뮬레이션하여 비교한 결과를 그림 3에 나타내었다. 이 그림

에서 알 수 있듯이 낮은 트래픽 인가율에서는 전송 지연(홉수)만을 고려하여 경로를 선택하는 방법이, 반대로 높은 트래픽 인가율에서는 적응 ELB 알고리즘이 평균지연에 있어서 성능이 우수함을 알 수 있다.

이는 전송 지연만을 고려한 경우가 홉수가 짧은 경로만을 항상 높은 확률로 선택하게 되므로 저부하에서는 병목 현상의 발생 확률이 작기 때문에 성능이 우수하지만 고부하에서는 병목 현상이 자주 발생하기 때문에 성능이 떨어지는 것이다. 반대로 적응 ELB 알고리즘은 망에 인가되는 트래픽을 망 전체의 사용 가능한 각 링크의 전송 용량에 비례적으로 분포시키기 때문에 고부하에서의 병목 현상에는 강하지만 저부하에서는 홉수가 적은 경로를 고려하지 못하게 된다.

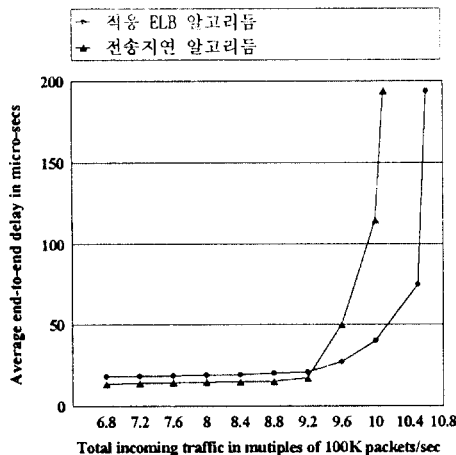


그림 3. 적응 ELB와 전송지연 경로설정 방법과의 성능 비교

다음에는 본 논문에서 제안한 전송지연을 적용한 적응 ELB 알고리즘을 적용한 경우와 기존의 적응 ELB 알고리즘을 적용한 경우를 시뮬레이션하여 비교한 결과를 그림 4에 나타내었다. 그림 4에서 볼 수 있듯이 전송지연을 적용한 적응 ELB 알고리즘을 적용한 결과는 저부하인 경우는 물론 고부하에서도 ELB보다 좋은 성능을 보임을 알 수 있다.

이는 전송지연을 적용한 적응 ELB 알고리즘이 저부하 경우에서의 전송지연을 적용한 방법의 장점을 고부하에서는 적응 ELB 알고리즘의 장점을 둘 다 포함하고 있기 때문이다. 즉, 저부하에서는 병목 현상이

적게 일어나기 때문에 전송지연이 짧은 경로를 선택하는 방법을 사용하여 평균 전송지연을 줄일 수 있고, 고부하에서는 적응 ELB의 R 경로설정 변수들을 참조하여 전송 용량이 넘지 않도록 R 경로설정 변수 값 쪽으로 Q 경로설정 변수를 재조정하기 때문에 망에 인가되는 부하를 적절히 균등하게 분포시키면서도 전송지연이 짧은 경로를 높은 확률로 선택할 수 있게 되므로 전체적인 평균전송지연을 줄일 수 있고 병목 현상도 줄일 수 있게 된다.

동시에 트래픽의 평균 지연이 급격히 증가하는 범위가 ELB 방식과 비슷하므로 병목 제어에 대한 성능도 유지됨을 볼 수 있다.

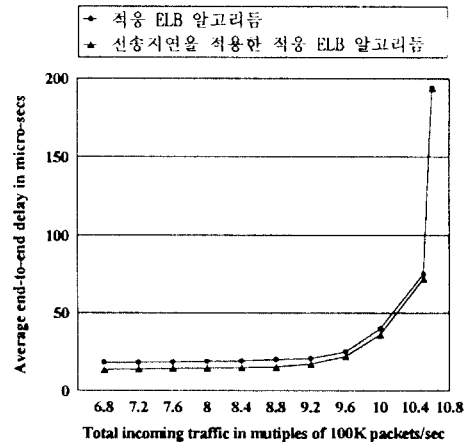


그림 4. 적응 ELB와 전송지연을 적용한 적응 ELB 방법과의 성능 비교

VI. 결 론

모든 경로설정 기법의 궁극적인 목표는 전체적인 평균 지연을 줄이는데 있다. 평균지연을 줄인다는 것은 사용자가 더욱 빠르게 서비스를 제공받을 수 있다는 것을 의미하며, 또한, 더욱 다양한 고품질의 서비스를 제공할 수 있다는 것을 말한다.

본 논문에서는 다중경로가 가능한 고속망에 있어서 ELB 경로설정기법에 거리에 비례하는 전송지연을 적용하여 최단 링크의 사용도를 높이면서 망의 평균지연을 줄이는 경로설정 알고리즘을 제안하고 성

능분석을 실시하였다. 이 기법은 다중경로가 가능한 임의의 망형태에 적용이 가능하다.

이 알고리즘은 기존의 ELB 알고리즘을 사용하는 경로설정 변수가 짧은 경로를 다른 경로보다 더욱 높은 확률로 선택할 수 있도록 갱신시켜 주는 것이다.

시뮬레이션을 통해서 제안된 알고리즘이 저부하에서는 물론, 고부하에서도 기존의 적응 ELB 경로설정 기법보다 그 평균지연에 대한 성능이 우수함을 보였다. 그리고 병목 제어에 대한 성능도 유지됨을 볼 수 있었다.

본 논문에서는 각 노드의 버퍼 용량을 무한대로 가정하였다. 그러나 실제로는 이런 가정이 성립하지 않으므로 유한한 버퍼 용량을 갖는 경우에 대한 연구가 수행되어야 할 것이다. 또한, 거리에 대한 비용을 흡수로 고려하지 않고 실제 매체의 거리에 따른 전송 지연을 적용한다면 더욱 정확하고 효과적인 경로설정 변수를 구할 수 있을 것이다.

참 고 문 헌

1. P. Harshavardhana, S. Dravida, and A. B. Bondi, "Congestion Control for Connectionless Networks via Alternate Routing," *IEEE GLOBECOM 91*, pp. 339-346, 1991.
2. A. Kershenbaum, P. Kermani, G. A. Grover, "MENTOR: An Algorithm for Mesh Network Topological Optimization and Routing," *IEEE Trans. Comm.*, vol. 39, pp. 503-512, 1991.
3. Erol Gelenbe, Xiaowen Mang, "Adaptive Routing for Equitable Load Balancing," *ITC14*, pp. 1465-1474, 1994.
4. G. I. Stassinopoulos, M. G. Kazantzakis, "A Computationally Efficient Iteration Solution of the Multidestination Optimal Dynamic Routing Problem," *IEEE Trans. Comm.*, vol. 39, pp. 1370-1378, 1991.
5. C. Kang, H. H. Tan, "Combined Channel Allocation and Routing Algorithm," *IEEE GLOBECOM 91*, pp. 1690-1695, 1991.
6. E. De Souza, E. Silva, and M. Gerla, "Queueing Network Models for Load Balancing in Distributed

Systems," *J. Parallel and Distributed Computing*, vol. 12, pp. 24-38, 1991.

7. Saewoong Bahk and Magda El Zarki, "Preventive Congestion Control based Routing in ATM Networks," *IEEE* 1994.
8. Z. Wang, D. W. Browning, "An Optimal Distributed Routing Algorithm," *IEEE Trans. Comm.*, vol. 39, pp. 1379-1387, 1991.
9. A. Alan B. Pritsker 「Introduction to Simulation and SLAM II」, 3rd edition, Pritsker, 1986.



주 만 식(Man-Sik Joo) 정회원
 1986년 2월: 연세대학교 전자공학과 공학사
 1995년 2월: 연세대학교 대학원 전자공학과 공학석사
 1995년 3월~현재: 연세대학교 대학원 전자공학과 박사과정

1986년~현재: 한국컴퓨터 재직중
 ※주관심분야: 고속망 트래픽제어



백 이 현(Lee-Hyun Baek) 정회원
 1993년 2월: 연세대학교 전자공학과 공학사
 1995년 8월: 연세대학교 대학원 전자공학과 공학석사
 1995년 8월~현재: 연세대학교 대학원 전자공학과 박사과정

※주관심분야: 네트워크 관리, 고속망 트래픽제어

주 판 유(Pan-Yuh Joo) 정회원
 연세대학교 대학원 전자공학과 박사과정
 한국통신학회 논문지 제21권 10호 참조(pp. 2660-2669)

강 참 언(Chang-Eon Kang) 정회원
 연세대학교 전자공학과 교수
 한국통신학회 논문지 제21권 10호 참조(pp. 2660-2669)