

---

# QoS 라우팅의 LSU 알고리즘 성능 향상 기법

## Performance Improvement of LSU Algorithms in QoS Routing

---

조강홍  
동양공업전문대학

Kang-Hong Cho(khcho@dongyang.ac.kr)

---

### 요약

본 논문에서는 기존에 제시된 QoS 라우팅의 링크 상태 갱신 알고리즘의 성능을 향상시킬 수 있는 플로우 유지 시간을 기반으로 한 LSU 알고리즘을 제안하였다. 기존에 제시된 LSU 알고리즘은 네트워크 트래픽의 통계 정보를 기반으로 하여 LSU 메시지 전송 여부를 결정하는 반면, 제안하는 알고리즘은 플로우 유지 시간을 기반으로 하여 LSU 메시지의 개수를 감소시킬 수 있기 때문에 기존에 제안된 알고리즘에 모두 적용하여 사용할 수 있다. 제안하는 알고리즘은 짧은 시간 안에 반복적으로 발생하는 LSU 메시지의 개수를 감소시키기 위해 플로우의 유지 시간에 대한 통계적 정보를 사용하였다. 알고리즘의 성능 평가를 위해 기존에 제시된 다양한 LSU 알고리즘을 구현하여 본 논문에서 제안하는 알고리즘을 적용하였고 실제 네트워크와 유사한 MCI 네트워크상에서 라우팅 Blocking 확률과 링크 당 평균 LSU 메시지의 개수를 성능 평가 항목으로 하여 시뮬레이션을 수행하여 제시하는 알고리즘의 우수성을 확인하였다.

■ 중심어 : | QoS 라우팅 | LSU | 플로우 | 유지시간 |

### Abstract

This paper has proposed Flow Holding Time based Link State Update(LSU) Algorithm that can improve the performance of the existing LSU algorithms. As the existing LSU algorithms are based on the network traffic information, the proposed algorithm is based on flow holding time so we can minimize the LSU message overhead and has not had a strong influence on the depreciation of QoS routing performance. We apply a flow holding time in order to decrease the number of LSU message. We have evaluated the performance of the proposed model and the existing algorithms on MCI simulation network using the performance metric as the QoS routing blocking rate and the mean update rate per link , it thus appears that we have verified the performance of this algorithm.

■ keyword : | QoS Routing | LSU | Flow | Holding Time |

---

## I. 서론

최근 컴퓨터 기술의 급속한 발전으로 인해 기존의 텍스트 위주의 사용자 환경에서 벗어나 이미지, 그래픽,

오디오 및 비디오 데이터 등을 제공하는 멀티미디어 사용자 환경으로 변화하고 있다. 이런 멀티미디어 서비스의 도입에 따라 화상 전송과 같은 넓은 대역폭을 요구하는 서비스, 더 나아가 영상, 방송과 같은 높은 대역폭

---

접수번호 : #090220-002  
접수일자 : 2009년 02월 20일

심사완료일 : 2009년 03월 18일  
교신저자 : 조강홍, e-mail : khcho@dongyang.ac.kr

과 실시간 전송을 요구하는 서비스가 많아지고 있다.

다양한 QoS 요구 조건을 가지는 사용자 트래픽을 적정 경로를 통해 목적지에 전달하기 위해서는 QoS 라우팅의 사용이 필수적이다. QoS 라우팅이란 적정 경로를 동적으로 결정하는 라우팅이라고 정의할 수 있으며, 여기서 적정 경로란 사용자의 플로우가 요구하는 QoS를 지원할 수 있으면서 서비스 제공자나 경로 사용 비용 등과 같은 정책 제약에 위반되지 않는 경로를 의미한다. QoS 라우팅은 QoS를 요구하는 플로우에 대하여 이러한 적정 경로를 찾아주되 네트워크의 자원 활용률을 극대화하고, 네트워크 엔지니어링과 네트워크에 발생하는 트래픽 부하 간에 일시적인 불일치가 발생하는 경우의 성능 저하를 최소화 하는 것을 목적으로 한다. 즉, 사용자의 플로우의 QoS 요구를 만족시킬 수 있는 경로를 선택하되 그 플로우에 의해 소모되는 네트워크 자원의 양을 절감하고 네트워크상의 부하 분포를 균등하도록 조절함으로써 네트워크 자원 활용률을 극대화해야 한다. 따라서 라우팅 시 고려되어야 할 가장 중요한 내용은 사용자의 QoS 요구 사항에 대한 최적 경로를 선택하는 것과 네트워크 자원을 고르게 활용하여 그 성능을 극대화하는 것이다. 이와 같이 인터넷을 통해 사용자의 QoS 요구 사항에 대한 최적의 적정 경로를 찾는 것이 QoS 라우팅의 가장 큰 이슈로 계속 연구되어 왔다 [1][2].

QoS 라우팅의 요구 조건을 만족시키는 적정 경로를 찾기 위해서 네트워크를 구성하는 라우터는 네트워크를 구성하는 노드와 링크에 대한 정적인 정보뿐만 아니라 링크의 가용대역폭과 같은 상태 정보를 정확히 파악하고 있어야 이를 바탕으로 적정 경로를 파악할 수 있다. 이와 같이 정확한 링크 상태를 반영하기 위해서는 링크의 상태가 변할 때마다 이웃 라우터에게 LSU(Link State Update) 메시지를 전송하여 링크의 현재 상태를 알려야하지만, 과도한 LSU 메시지의 발생은 네트워크 자원을 낭비하고 트래픽 부하가 심할 경우에는 트래픽 혼잡의 원인이 될 수 있다. 반대로 LSU 메시지의 전송이 충분하지 못하면 각 라우터에서는 정확한 링크 상태 정보를 알지 못하고 과거의 데이터를 통해 경로를 선택하기 때문에 신뢰성이 떨어지게 된다. 즉, 정확한 링크

의 상태 반영 빈도와 업데이트 비용 간에는 반비례 관계가 존재한다. 이와 같이 가능한 LSU 메시지의 전송을 줄이면서 QoS 라우팅의 성능을 감소시키지 않기 위한 링크 상태 갱신 기법에 대한 연구 또한 QoS 라우팅의 다른 하나의 이슈로 계속 연구되었다[3][4].

본 논문에서는 링크 상태 갱신 기법에 대한 연구로 기존에 제시되었던 LSU 기법들과 전혀 다른 플로우 유지 시간을 기반으로 하여 QoS 라우팅 성능은 그대로 유지하면서 LSU 메시지의 수를 크게 감소시킬 수 있는 새로운 기법을 제시한다. 제 2 장에서는 기존에 연구되었던 LSU 기법에 대해 소개하고 제3장에서는 제안하는 기법에 대해 설명한다. 제4장에서는 시뮬레이션을 통해 기존의 기법들과 성능 평가 및 성능 비교를 수행하여 제안하는 기법의 우수성을 확인하고 제5장 결론에서는 제안하는 새로운 기법의 의미와 앞으로의 연구 방향에 대해 설명한다.

## II. 관련 연구

QoS 라우팅의 적정 경로를 위해서는 정확한 링크 상태 정보를 기반으로 해야 한다. 이를 위해 링크 상태 정보를 제공하기 위한 다양한 형태의 LSU(Link State Update) 알고리즘이 제안되었다[5-9].

Period Based(PB) LSU 알고리즘은 네트워크를 구성하는 라우터들이 주기적으로 LSU 메시지를 서로 주고받아 네트워크 상태 정보를 유지한다. PB LSU 알고리즘은 LSU 메시지의 트래픽의 양을 조절할 수 있고 쉽게 구현가능하다는 장점을 가지지만 각 링크의 네트워크 상태의 변화를 중요한 시점에 알려주지 못한다는 단점을 가진다. 따라서 일반적으로 그 성능이 좋지 못하다.

Threshold Based(TB) LSU 알고리즘은 사용 가능한 대역폭이 정해진 임계값 이상으로 변화할 경우 LSU 메시지를 전송한다. 한 라우터의 링크  $i$ 에 대한 가장 최근에 갱신된 사용 대역폭이  $bw_i^o$  이고 현재 사용 대역폭의 값은  $bw_i^c$  그리고 임계값이  $th$  경우,

$|bw_i^o - bw_i^c| / bw_i^c > th$ , ( $bw_i^o > 0$ ) 조건에서 LSU 메시지가 발생된다. 이 알고리즘은 사용가능한 대역폭이 작아질수록 좀 더 자세한 정보를 제공하기 때문에 효과적이지만, 네트워크 트래픽 부하가 클수록 다수의 LSU 메시지가 발생하는 문제와 적절한 임계값을 찾기가 쉽지 않다는 단점을 가진다.

Equal Class Based(ECB) LSU 알고리즘은 사용가능한 대역폭을 상수값  $B$ 에 의해  $(0, B)$ ,  $(B, 2B)$ ,  $(2B, 3B)$ , ..., 등으로 동일한 크기의 클래스로 나눈 후 사용가능한 대역폭이 이전과 다른 클래스로 변화될 경우 메시지를 전송한다.

Unequal Class Based(UCB) LSU 알고리즘은 ECB 알고리즘과 유사하게 클래스를 구분하는데, 이 때 상수값  $B$ 와 인자  $f$ 에 의한 등비수열 값으로  $(0, (f+1)B)$ ,  $((f+1)B, (f^2+f+1)B)$ , ..., 와 같이 클래스를 나누어 사용한다. UCB는 ECB에 비해 클래스 사이에 서로 다른 가중치를 두어 좀 더 자세한 링크 상태의 변화를 가능하게 한다.

Dynamic Threshold Based(DTB) LSU 알고리즘은 고정된 임계값 대신에 정해진 주기  $T$  동안에 발생된 LSU 메시지의 수에 따라 임계값을 변화시킨다.  $R_0$ 가 초기의 업데이트 비율,  $\widetilde{R}_k$ 가 마지막  $T$  주기에서의 업데이트 비율이라고 할 때,  $k$  구간에서 사용될 임계값  $th_k$  는

$$\begin{cases} th_{k-1} + \Delta th, & \text{if } \widetilde{R}_k \leq R_0, \\ th_{k-1} - \Delta th, & \text{if } \widetilde{R}_k > R_0, \end{cases}$$

에 의해 결정된다. 이와 같이 임계값의 변화를 통해 LSU 메시지의 개수를 제어할 수 있는 장점을 가진다.

Second-moment Based(SB) LSU 알고리즘은 대역폭  $C$ , 사용 중인 대역폭에 대한 평균값  $\mu$ 와 분산  $\sigma^2$ 을 이용한 stability function  $F(\mu, \sigma^2) = \frac{\sigma^2}{(C-\mu)C}$ 의 값이 임계값( $th$ )보다 클 경우에 LSU 메시지를 전송한다. SB 알고리즘은 통계값을 이용하여 네트워크 트래픽 특성을 적절히

반영하여 안전성을 갖추었으나 역시 고정된 임계값을 사용한다는 단점을 가진다.

Simple-Adaptive(SA) LSU 알고리즘은 LSDB에 저장되어 있는 사용 가능한 대역폭  $B_n$ , 링크의 실제 사용 가능한 대역폭  $\widetilde{B}(t)$ , LSU 메시지 전송을 결정하는 시점의 링크에서 제공 중인 서비스 개수  $\widetilde{N}(t)$

라고 하면  $|B_n - \widetilde{B}(t)| \geq \frac{\widetilde{B}(t)}{\widetilde{N}(t)}$  조건을 만족하면

LSU 메시지를 전송한다.

Statistical Threshold(ST) LSU 알고리즘은 해당 링크를 통해 제공되고 있는 경로 설정 요구의 사용 대역폭들의 통계값을 바탕으로 임계값을 계산하는 방식으로, 현재 링크를 통해 제공되고 있는 경로 설정 요구 대역폭들의 평균을  $\widetilde{m}$ , 요청 대역폭들의 표본 분산을  $\alpha$ , 대역폭 보장 변수를  $\beta$ , 제공되고 있는 서비스의 요청 대역폭들 중 최대값을  $r_{\max}$ 라고 할 경우 임계값은  $th = \max(\widetilde{m} + \beta\alpha, r_{\max})$ 에 의해 결정된다. ST 알고리즘은 임계값을 현재 링크에서 사용 가능한 통계값을 이용하여 다양한 네트워크 환경의 특성에 따라 임계값이 결정되는 장점을 가지고 있지만, 트래픽의 경로 설정 요구 대역폭의 변화가 심할 경우 표본 분산값이 커짐으로써 성능이 떨어지는 단점과 LSU 메시지의 발생 수를 조절할 수 없다는 단점을 가진다.

### III. 제안하는 LSU 알고리즘

기존에 제안된 알고리즘들은 대부분 링크의 가용 대역폭을 기준으로 하여 임계값을 넘었을 경우 LSU 메시지를 전송하도록 되어 있다. 따라서 링크의 상태가 임계값을 넘을 때마다 LSU 메시지를 전송하게 된다. 바꾸어 말하면 네트워크 트래픽의 특성에 따라 링크의 상태가 자주 변하게 되면 LSU 메시지 발생은 크게 증가하게 되고, LSU 메시지의 연속적인 발생을 조절할 수 없게 된다.

예를 들어 [그림 1]과 같이 링크의 상태 변화가 심하

면  $\lambda(t)$ ,  $\lambda(t + \Delta)$  시간 사이에  $LSU(t)$ ,  $LSU(t + 1)$ , ...,  $LSU(t + 5)$  까지 연속적으로 LSU 메시지가 발생된다. 만약  $\Delta$  시간이 허용할 수 있을 정도로 작아서  $\Delta$  시간 안에 다른 플로우의 연결 요구가 없어 링크의 가용 대역폭의 변화 확률이 낮다면 이전에 보냈던 연속적인 LSU 메시지는 의미가 없게 되고 자원의 낭비만 가져오게 된다. 결과적으로 최종적으로 보낸  $LSU(t + 5)$  메시지만 발생해도 QoS 라우팅 성능 상에서는 동일한 결과를 가져오게 될 것이다. 제안하는 알고리즘은 앞에서 제시된 모든 기존의 알고리즘에 추가적으로 적용되어 LSU 메시지를 줄일 수 있다. 만약 QoS 서비스를 지원하기 위한 네트워크의 특성에 따라 플로우 유지 시간을 QoS 라우팅 연결 시 미리 알고 있다고 가정하면 다음에 조건에 따라 LSU 메시지를 발생시킬지 아니면 다음 시점에 발생시킬지 결정하게 된다[10].

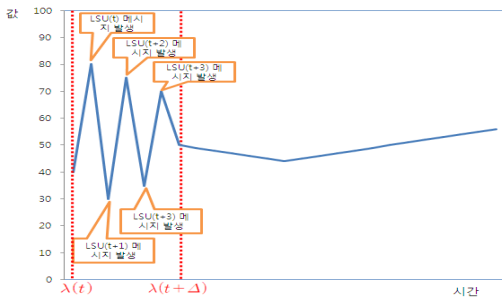


그림 1. LSU 메시지의 발생

현재 시점  $t$  에서 링크  $e_l$  을 사용 중인  $\theta_j$  의 플로우 유지 시간을 가지는  $n$  개의 플로우의 집합을  $flows_i^{e_l}[t, t + \theta_j]$  ( $i \in 1, 2, \dots, n$ ) 이라고 정의하자. 이 플로우들 중 기준시간  $\Delta$  안에 서비스 종료 예정인 플로우가 있으면 현재 시점에서 LSU 메시지를 발생시키지 않고, 다음번 LSU 메시지 발생 여부를 결정하는 시점까지 기다린다. 다음번 시점에서도 동일하게 적용하여 LSU 메시지 발생 여부를 결정하면 되는데, 만약 각 시점에서  $\Delta$  시간 안에 계속적인 플로우가 존재할 경우 LSU 메시지가 발생되지 않아 링크 상태

정보가 부정확하게 되므로 일정 횟수  $c$  이상 넘지 않도록 제한을 둔다.

본 논문에서는 기존에 제시된 다양한 알고리즘에 제안하는 알고리즘을 적용할 수 있고 그 내용은 다음과 같다. 여기서  $th$  는 기존에 제시된 각 알고리즘이 계산한 임계값을 의미하며, 이 임계값을 넘으면 그 후에 다음과 같이 적용할 수 있다.

```

if(기존에 제시된 알고리즘에서 LSU 메시지 발생될 경우)
{
  for(j = 0; j < n; j++)
    if((∃ any flowsiel[t, t + θj] ∈ (t + θj < Δ))
      No update LSU message;

  if((all flowsiel[t, t + θj] ∈ (t + θj > Δ))
    Update LSU message;
}
    
```

일반적으로 플로우 유지 시간을 알지 못하는 경우에 이를 위해 네트워크에서 서비스되는 통계적인 플로우 유지 시간 값을 계산하여 플로우의 서비스 종료 시간을 추측하여 사용할 수 있다. 네트워크상에서 제공된 플로우 유지 시간의 평균을  $\mu$ , 분산을  $\sigma^2$  라고 할 경우에 플로우 유지 시간은  $\mu - \beta\sigma/n \leq \tilde{\theta} \leq \mu + \beta\sigma/n$  범위로 추정할 수 있다. 여기서, 95% 신뢰도일 경우  $\beta$ 를 1.96, 99% 신뢰도일 경우  $\beta$ 는 2.58로 유지 시간을 계산할 수 있다. 계산된 플로우 유지 시간을 이용하면 앞에서 설명한 것과 마찬가지로 LSU 메시지 발생 여부를 판단할 수 있다. 현재 시점에서 각 플로우의 추정된 유지시간  $\tilde{\theta}$  를 이용해서 계산된 각 플로우의 종료 시간이 기준 시간  $\Delta$  안에 존재하면 LSU 메시지를 발생시키지 않고 LSU 메시지 발생 시점까지 기다린다. 역시 마찬가지로 각 시점에서  $\Delta$  시간 안에 계속적인 플로우가 존재할 경우 LSU 메시지가 발생되지 않아 링크 상태 정보가 부정확하게 되므로 일정 횟수  $c$  이상 넘지 않도록 제한을 둔다.

#### IV. 실험 및 결과

본 논문에서 성능 평가를 위해 자바를 기반으로 한 라우팅 시뮬레이터를 구현하였고 [그림 2]와 같이 미국 백본 네트워크와 유사하게 구성되어 QoS 라우팅 성능 평가에 대표적으로 사용되는 MCI 네트워크 토폴로지 구성하였다[1][2][5]. MCI 네트워크 토폴로지는 19개의 노드와 77개의 OC-3(155Mbit/s)와 T3(45Mbit/s)의 선로들로 구성되어 있다.

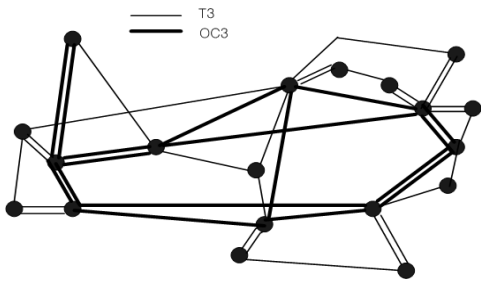


그림 2. 시뮬레이션에서 사용한 MCI 네트워크 토폴로지

송신 노드와 수신 노드는 두 노드가 같은 경우를 제외하고 균일한 확률로 랜덤하게 선택되도록 하였고 1~5 Mbit/s 사이의 CBR(Constant Bit Rate)트래픽을 균일 분포(uniform distribution)에 따라 랜덤하게 발생하였고, 이 때 플로우 유지 시간은 평균 1200초 또는 2400 초 값을 가지고 지수 분포(exponentially distribution)에 따라 발생시켰다. 각 알고리즘의 성능을 확실히 구분하기 위하여 네트워크의 트래픽 부하 트래픽 도착율에 따라 0.1에서 0.2까지 변화시켜서 지수 분포에 따라 발생시켰다.

요청된 트래픽 요구 대역폭을 만족시키는 적정 경로를 찾기 위한 QoS 라우팅 알고리즘은 일반적으로 사용되고 적정 경로 중 비용이 가장 낮은 경로를 찾는 Dijkstra 알고리즘을 사용하였다.

본 논문에서 제안하는 알고리즘의 성능을 평가하기 위한 성능 평가 항목으로 일반적으로 QoS 라우팅 알고리즘의 성능을 평가하기 위해 사용되는 Blocking 확률,  $P_{block}$ 과 링크 당 LSU 메시지 수,  $R_{update}$ 를 사용하였다.  $N_{block}$ 이 경로 설정 요구가 받아들여지지 않아

Blocking된 횟수이고  $N_{totalrequest}$ 는 전체 경로 설정 요구 횟수 일 경우  $P_{block}$ 는 다음과 같다.

$$P_{block} = \frac{N_{block}}{N_{totalrequest}} \quad (1)$$

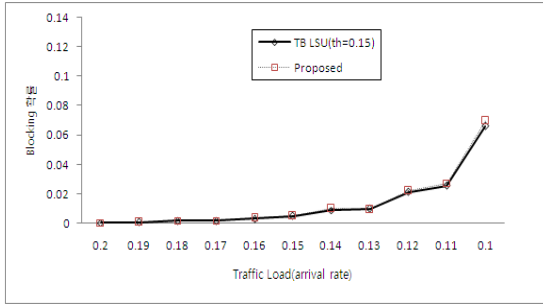
예를 들어 만약 전체 플로우의 개수( $N_{totalrequest}$ )가 100개이고 이중 Blocking 된 플로우의 개수( $N_{block}$ )가 10개라고 하면 Blocking 확률( $P_{block}$ )은 0.1이 된다. 또한  $N_{update}$ 가 전체 LSU 메시지 발생 횟수이고  $N_{link}$ 는 네트워크를 구성하는 링크의 총 개수 일 경우  $R_{update}$ 는 다음과 같다.

$$R_{update} = \frac{N_{update}}{N_{link}} \quad (2)$$

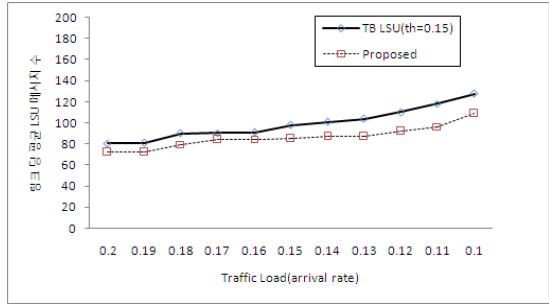
예를 들어 만약 전체 링크의 개수( $N_{link}$ )가 그림2와 같이 77개이고 LSU 메시지의 발생 횟수( $N_{update}$ )가 1000개 라고 가정하면 링크당 LSU 메시지의 수 ( $R_{update}$ )는 약 12.987 정도가 된다.

위의 Blocking 확률과 링크 당 LSU 메시지의 발생 횟수를 성능 평가 항목으로 하면 QoS 라우팅 알고리즘의 Blocking 확률의 큰 성능 저하 없이 링크 당 LSU 메시지 수가 감소된 결과를 보일 수 있다.

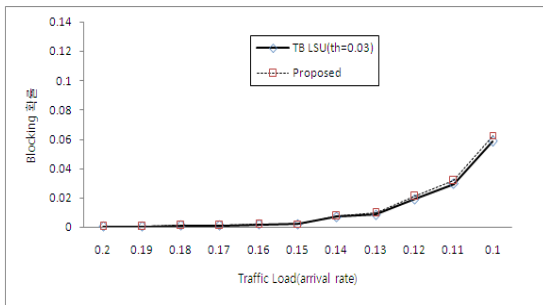
[그림 3]은 기존에 제안된 LSU 알고리즘에 본 논문에서 제안하는 알고리즘을 적용하여 트래픽 부하에 따른 Blocking 확률과 링크별 평균 LSU 메시지 수를 보여주고 있다. LSU 알고리즘의 성능을 좀 더 분명히 구분하기 위해 트래픽 부하를 0.2에서 0.1까지 단계적으로 증가시켰다. [그림 3]의 (a) ~ (j)의 그림은 기존에 제시된 알고리즘에 본 논문에서 제안하는 알고리즘을 각각 적용한 결과를 보여준다. (a)는 기존의 임계값 기반의 TB LSU(th=0.15) 알고리즘과 이 알고리즘에 제안하는 알고리즘을 적용한 후 Blocking 확률을 나타내고 있는데 크게 차이가 없음을 볼 수 있다. 반면 (b)의 그림을 보면 제안하는 알고리즘을 적용한 후 링크 당 평균 LSU 메시지의 수는 크게 감소한 것을 확인할 수 있다. 즉 제안하는 알고리즘을 적용한 결과 QoS 라우팅의 큰 성능 저하 없이 링크 당 LSU 메시지 수가 크게 감소된 것을 뚜렷하게 확인할 수 있다. [그림 3]의 (c) ~ (j)의



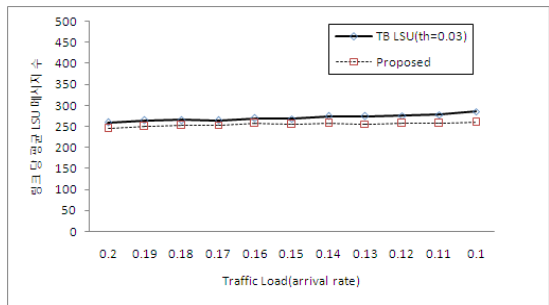
(a) 제안된 TB LSU(th=0.15)의 Blocking 확률



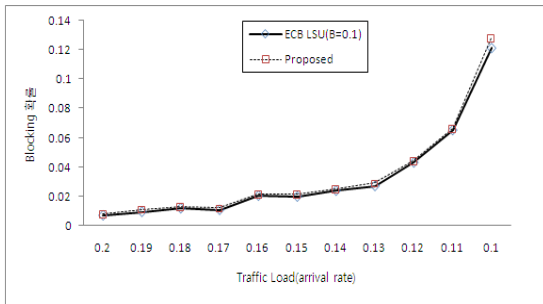
(b) 제안된 TB LSU(th=0.15)의 링크별 평균 LSU 메시지 수



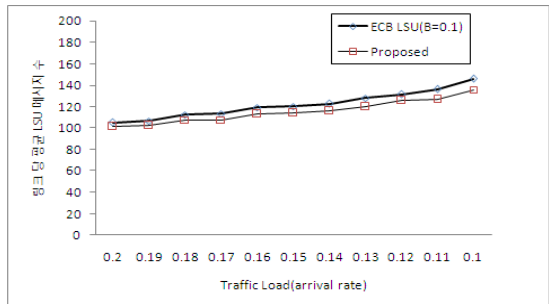
(c) 제안된 TB LSU(th=0.03)의 Blocking 확률



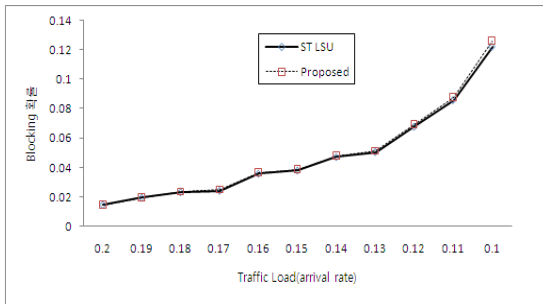
(d) 제안된 TB LSU(th=0.03)의 링크별 평균 LSU 메시지 수



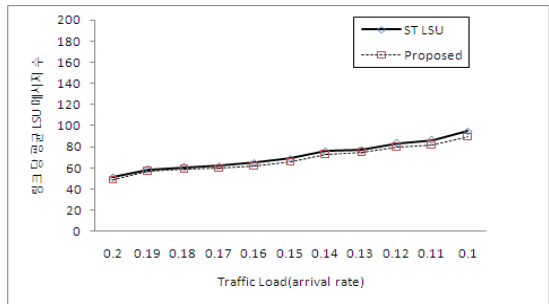
(e) 제안된 ECB LSU(B=0.1)의 Blocking 확률



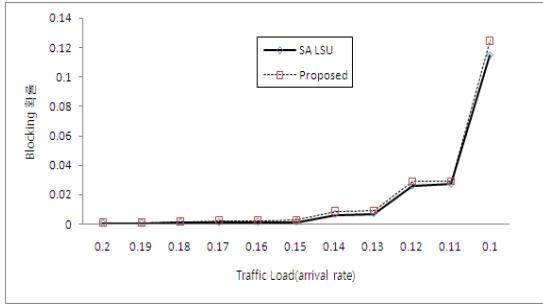
(f) 제안된 ECB LSU(B=0.1)의 링크별 평균 LSU 메시지 수



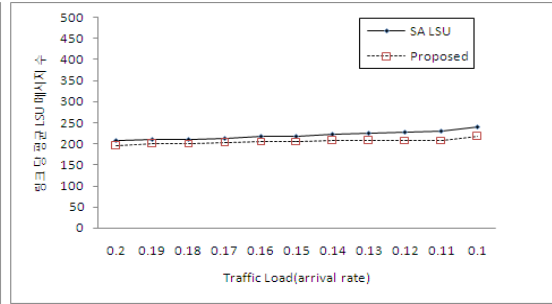
(g) 제안된 ST LSU의 Blocking 확률



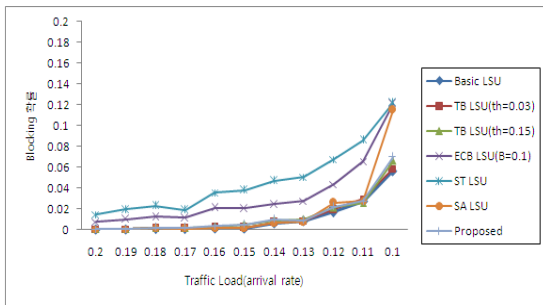
(h) 제안된 ST LSU의 링크별 평균 LSU 메시지 수



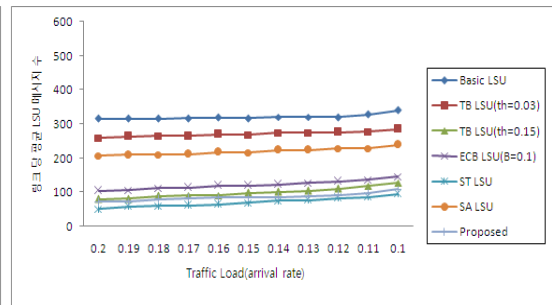
(i) 제안된 SA LSU의 Blocking 확률



(j) 제안된 SA LSU의 링크별 평균 LSU 메시지 수



(k) 트래픽 부하에 따른 Blocking 확률



(l) 트래픽 부하에 따른 링크별 평균 LSU 메시지 수

그림 3. 제안하는 알고리즘의 성능 평가

결과를 보면 TB LSU(th=0.03), BCB LSU(B=0.1), ST LSU, SA LSU 알고리즘에서도 정도의 차이가 있지만 모든 알고리즘에서 Blocking 확률의 큰 변화 없이 링크 당 평균 LSU 메시지의 수가 크게 줄어든 것을 볼 수 있다.

[그림 3]의 (k)와 (l)은 기존에 제시된 알고리즘과 (a) ~ (j) 중 가장 좋은 성능을 보인 Proposed TB LSU(th=0.15)과의 성능 차이를 확인하기 위한 그림이다. 여기서 Basic LSU 알고리즘은 링크의 상태가 변할 때마다 LSU 메시지를 보내는 방식으로 평균 LSU 메시지 수가 가장 높지만 링크 상태의 변화를 즉각적으로 알 수 있기 때문에 Blocking 확률의 성능 상으로 가장 좋고 이상적이기 때문에 기준 값으로 볼 수 있다. 보는 바와 같이 LSU 알고리즘에 따라 Blocking 확률이 어느 정도 차이가 나는 것을 볼 수 있는데, ST LSU 알고리즘과 SA LSU 알고리즘과 같이 고정 임계값이 아니라 트래픽의 특성에 따라 임계값이 변하는 적응형 알고리즘이 오히려 고정 임계값을 사용하는 TB LSU 알고리

즘에 비해 성능이 좋지 않은 것을 볼 수 있다. 이 이유는 고정 임계값 알고리즘의 경우 시뮬레이션을 통해 네트워크 특성에 맞도록 실험한 통해 얻은 최적의 임계값을 사용하였고 반대로 적응형 알고리즘은 변화가 심한 네트워크 트래픽 특성을 제대로 파악하지 못했기 때문이다. 또 한 가지는 Blocking 확률을 링크별 평균 LSU 메시지 수의 결과를 같이 살펴보아야 하는데 ST LSU 알고리즘은 링크별 평균 LSU 메시지 수가 가장 작지만 이것은 트래픽의 특성을 제대로 파악하지 못해 LSU 메시지가 적게 발생되었기 때문이다. 따라서 LSU 메시지가 적게 발생되었기 때문에 Blocking 확률이 크게 높아진 것을 볼 수 있다. SA LSU 알고리즘은 Blocking 확률에서는 어느 정도 좋은 결과를 보여주었지만, 평균 LSU 메시지 수가 높게 나타난 것을 알 수 있다. 이것은 트래픽의 특성에 따라 영향을 받을 수 있기 때문에 좀 더 다양한 환경에서의 시뮬레이션이 추가적으로 필요한 것으로 여겨진다.

결과적으로 통계적인 플로우 유지 시간을 기반으로

LSU 메시지 발생을 제어하는 본 연구의 알고리즘을 기존의 알고리즘에 적용한 결과 기존의 알고리즘에서 QoS 라우팅 성능의 저하 없이 LSU 메시지의 수를 크게 감소시킴으로써 제안하는 알고리즘의 우수성을 확인하였다.

## V. 결론

본 논문에서는 기존의 다양한 LSU 알고리즘에 대해 시뮬레이션을 통해 성능 평가를 수행하였고, 제안하는 알고리즘의 적용을 통해 QoS 라우팅 성능이 크게 저하되지 않으면서 LSU 메시지 발생을 약 10~20% 정도 크게 줄일 수 있음을 확인하였다. 네트워크의 트래픽 통계 정보를 통해 임계값을 불변 또는 적응시켜 LSU 메시지 발생 여부를 판단하는 기존의 LSU 알고리즘과 달리 플로우의 유지 시간을 기반으로 연속적으로 발생하는 LSU 메시지를 줄여 네트워크의 자원 낭비를 크게 줄일 수 있었다.

성능 평가를 위해 시뮬레이션을 진행하던 중 임계값, 트래픽 발생 변수값과 같은 실험 결과에 영향을 미치는 여러 인자들이 존재하여 이에 따라 약간의 결과 차이가 나타나는 것을 볼 수 있었다. 추가적으로 좀 더 정확한 성능 평가를 위해 다양한 형태의 네트워크를 토폴로지를 구성하는 것과 VBR, CBR 등의 트래픽 특성을 혼합하여 적용하는 것이 필요하겠고 시뮬레이션 결과에 중요한 영향을 미치는 트래픽의 특성 및 종류를 좀 더 다양화해서 좀 더 정확하게 실험하는 것이 필요하다고 생각된다.

## 참고 문헌

[1] Q. Ma and P. Steenkiste, "On path selection for traffic with bandwidth guarantees," ICNP , pp.191, Fifth International Conference on Network Protocols (ICNP'97), 1997.

[2] G. Apostolopoulos and G. roch, "QoS Routing: A

Performance Perspective," in proceedings of ACM SIGCOMM 1998.

- [3] G. Apostolopoulos, G. Roch, and Sanjay Kamat, "Improving Routing Performance Under Inaccurate Link State Information," roceedings of the 16th International Teletraffic Congress, 1999.
- [4] D. Lorenz and A. Orda, "QoS Routing in Networks with Uncertain Parameters," ACM Transactions on Networking, Vol.6, pp.768-778, 1998.
- [5] 이진주, 정민영, 이태진, 추현승, "대역폭 보장을 위한 통계적 임계값 기반의 링크 상태 갱신 알고리즘", 정보과학회논문지, 제35권, 제5호, 2008.
- [6] Z. Miao, Z. Huiling, O. K. L. Victor, M. Zhengxin, "A Stability-Based Link State Updating Mechanism for QoS Routing," IEEE International Conference on Communications, Vol.1, pp.33-37, 2005.
- [7] M. T. Eva, M. B. Xavier, S. L. Sergio, D. P. Jordi, "The Prediction Approach in QoS Routing," Communications, 2006. ICC IEEE International Conference on Vol.3, Issue , pp.1020-1025, 2006(6).
- [8] A. Ariza, E. Casilari, and F. Sandoval, "QoS Routing with adaptive updating of link states," Electronic Letters, Vol.37, pp.604-606, 2001.
- [9] S. H. Choi, M. H. Jung, M. Y. Chung, M. J. Yang, T. I. Kim, and J. H. Park, "Simple-Adaptive Link State Update Algorithm for QoS Rougint," LNCS 3991, Part I, pp.969-972, 2006.
- [10] S. Xun, Gustavo de Veciana, "Predictive routing to enhance QoS for stream-based flows sharing excess bandwidth," Computer Networks, Vol.42, pp.65-80, 2002.



저 자 소 개

조 강 홍(Kang-Hong Cho)

정회원



- 1997년 2월 : 성균관대학교 정보공학과 학사
- 1999년 2월 : 성균관대학교 전기전자 및 컴퓨터공학부 석사
- 2003년 2월 : 성균관대학교 전기전자 및 컴퓨터공학부 박사

▪ 2003년 3월 ~ 현재 : 동양공업전문대학 조교수

<관심분야> : 트래픽 분석, 보안, QoS 라우팅