

큐 상태 정보와 예측을 기반으로 한 흐름제어기법

A Flow Control based on Queue Dynamics and Estimate Information

서 주 하*

Seo, Ju-Ha

정 부 영**

Jung, Boo-Young

유 현 희***

Ryu, Hyun-Hee

Abstract

In this paper, We propose a new flow control scheme based on Queue Dynamics and it's Estimated Information in order to achieve higher throughput and network efficiency using control of the best-effort traffic. The feedback Information gives a result that compare queue length with queue threshold. Traffic changes at the time when queue length is cross over under the queue thresholds. The performance of the purposed scheme has been analyzed mathematically and we verify efficiency of the proposed method by means of simulation.

1. 서 론

네트워크의 다양한 응용을 위해 데이터를 송신자로부터 수신자까지 보다 신뢰성 있게 전송하고 고속망 환경에서 망 자원 사용의 효율성과 전송 성능을 높이기 위한 흐름제어의 중요성이 더욱 커지고 있다[1,2]. 흐름제어 기법은 노드와 노드사이에서 트래픽을 제어하는 노드간 제어(link-by-link)[3,4]와 데이터의 전송자와 수신자 사이의 데이터 트래픽을 제어하는 양단간 제어(end-to-end) [5,6]가 있다.

노드간 제어는 제어정보가 이웃한 노드 사이에 교환되므로 양단간 제어에 비하여 제어정보의 전달 지연이 작기 때문에 망 환경 변화에 빨리 적응할 수 있다. 제어 방법에 따른 분류로는 전송되는 데이터의 양을 제어하는 윈도우 기반(window-based) 제어와 전송되는 데이터의 속도를 제어하는 전송률 기반(rate-based) 제어기법이 있다. 윈도우 기반의 흐름

제어 기법은, 통신하고자 하는 호스트사이에 서로 동의한 윈도우라 불리는 일정한 크기 또는 동적으로 변하는 크기를 갖는 버퍼를 설정하고, 이 버퍼를 가능한 효율적으로 이용하여 오버플로우 없는 데이터의 전송을 목표로 한다. 이 기법의 단순성으로 인하여 TCP를 포함한 많은 통신 프로토콜에서 윈도우 기법을 이용하고 있다[8,9]. 그러나 버퍼의 상태를 교환하여야 하는 윈도우 기법은 통신 경로상에 많은 중간노드가 존재할 수 있는 인터넷과 같은 multi-hop 통신 환경에서는 부적절할 수 있다.

중간 노드에서 흐름제어 기능을 제공하기 위해서 여러 가지 방법이 제안되고 있다. 그 예로, 인터넷 환경에서 제안된 예약 프로토콜의 일종인 RSVP가 있다 [10]. 예약 프로토콜은 각 통신 노드의 통신 자원인 대역폭, 데이터 전송 비율 등을 미리 할당하는 방법으로 몇 가지 부담을 가지고 있다. 첫째, 각 통신경로의 데이터 흐름을 조절하기 위하여 개개의 데이터 패킷을 스케줄링을 하여야 한다. 둘째, 이러한 방법은 통신자원의 낭비를 초래한다. 예를 들면, 대량의 데이터를 간헐적으로 전송하는 경우 통신 경로상에 있는 노드들은 송신자가 데이터를 전송하지 않음에도 불구하고 기법상의 결점으로 인하여 통신자원을 계속

* 강원대학교 컴퓨터정보통신공학과 교수, 공학박사

** 강원대학교 산업대학원 컴퓨터공학과 석사과정

*** 강원대학교 컴퓨터정보통신공학과 석사과정

점유하여 통신자원의 낭비를 초래한다. 따라서 통신망의 모든 데이터 흐름에 대하여 통신자원을 예약하는 것은 바람직하지 않다. 또 다른 문제점은 중간 노드에서 데이터를 축적방식에 의하여 전달하기 때문에 예기치 못한 데이터의 전달 지연이 발생한다. 일반적으로 중간 노드들은 통신망의 폭주로 인한 데이터의 손실을 방지하기 위하여 필요 이상의 버퍼를 준비한다. 그러나 통신망의 속도가 빨라짐에 따라 과도한 버퍼링에 의한 지연은 데이터의 전달 지연을 예측하기 어렵게 한다. 따라서 손실된 데이터의 재전송을 위한 timeout 값의 결정을 어렵게 하는 윈도우를 기반으로 한 종단간 흐름제어는 비효율적이라고 할 수 있다.

전송률 기반 제어는 전송되는 데이터의 속도를 직접 제어하므로 고속망 환경에서 효율적인 흐름제어 기법으로 채택되고 있다[1,4,5,7]. ATM 포럼에서도 셀 손실율의 보장을 요구하는 ABR(Available Bit Rate) 트래픽을 제어하기 위한 방법으로써 윈도우 기반 제어와 전송률 기반 제어 사이에 많은 논란을 거듭한 끝에 94년 9월에 전송률 기반 제어 방식을 채택하였다.

최선형 트래픽은 UBR(Unspecified Bit Rate) 트래픽으로써 QoS(Quality of Service)요구가 명시되지 않기 때문에 QoS요구가 명확한 보장형 트래픽이 사용하고 남은 여분의 망 자원을 이용하여 전송 서비스를 받는다. 따라서 망 및 노드 자원과 트래픽의 상태에 관한 제어정보에 의하여 전송환경에 맞게 가장 효율적으로 전송되어야 한다. 제어정보에 의하여 트래픽 전송률을 제어하는 흐름제어 기법은 망 상태에 관한 정보를 얻는 기능과 얻어진 정보에 기초하여 전송률을 제어하는 기능으로 구성된다. 본 논문에서는 수신측에서의 임계값을 낮은쪽에서 높은쪽으로 또는 높은쪽에서 낮은쪽으로 넘을 때를 알리는 큐 정보를 피드백하면 전송측은 전달받은 상태정보에 근거하여 데이터의 전송속도를 조절하는 전송률 기반 흐름제어 기법을 제안한다.

2장에서는 전송률 제어 기법 3장에서는 제안된 기법을 설명하였고 4장에서는 수학적 해석으로 제안한 기법의 성능을 측정하기 위한 모델을 제시하고 중요한 파라미터를 해석하였다. 5장으로 성능분석으로 시뮬레이션 모델과 결과를 분석하고 마지막 6장으로 본 연구의 결론을 기술하였다.

2. 전송률 제어기법

지금까지 제안된 전송률 제어기법의 대부분은 망의

트래픽 혼잡을 데이터가 지나는 경로상에 존재하는 노드의 평균 큐 길이 Q 에 의해 결정하였다[5,6,11]. 트래픽 혼잡이 발생한 노드는 데이터 패킷의 한 비트를 이용하여 전송측에게 큐의 상태정보를 전달한다. 큐의 평균 길이가 임계값(threshold) Q_{TH} 보다 크면 1 작으면 0을 나타낸다. 노드는 데이터 패킷의 비트값이 1이면 비트값을 변경하지 않으며 패킷의 비트값이 0이고 큐의 상태정보가 1이면 상태정보값을 패킷의 비트값으로 복사하여 다음 노드로 전송한다. 전송측은 수신측에서 전달된 상태정보에 따라서 전송률을 증가시키거나 감소시킨다. 이와 같이 평균 큐 길이를 이용하여 망의 트래픽 혼잡을 파악하는 방법은 다음과 같은 결점을 갖고 있다.

- Q 의 값이 Q_{TH} 보다 크면서 큐의 길이가 감소하고 있는 상황에서도 전송측에서 전송률을 더욱 감소시킨다.
- Q 의 값이 Q_{TH} 보다 작으면서 큐의 길이가 증가하고 있는 상황에서 전송측에서 전송률을 더욱 증가시킨다.

위의 두 가지 이유 때문에 전송률의 최대값 λ_{max} 와 λ_{min} 의 차이가 커지게 되어 망 사용의 효율을 감소시킨다.

따라서 본 논문에서 큐에 대한 정보를 이용하여 송신측에서 수신측의 큐의 길이를 예측하여, 이 예측한 값을 기반으로 전송률을 조절하는 기법을 제안하였다.

3. 제안된 기법

제안된 전송률 제어 기법은 네트워크의 전송률을 네트워크의 상태에 따라 조절하기 위해서 큐의 정보를 이용한다. 수신기의 처리 능력을 알 수 있는 방법은 수신기의 버퍼(큐)의 오버플로우 및 언더플로우의 여부로 알 수 있다. 오버플로우 시는 패킷의 손실을 유발하고 언더플로우시는 큐의 활용을 제대로 못하는 상황이 발생한다. 따라서 이 논문에서는 큐 길이가 Q_{th} 보다 커지는 순간과 작아지는 순간에 발생하는 큐 상태정보를 피드백하여 데이터의 손실을 최소화하고 큐의 활용을 최대화하는 방안을 제안하고자 한다.

망은 하나의 큐와 전파 지연에 의해서 다음과 같은 형태로 모델링된다.

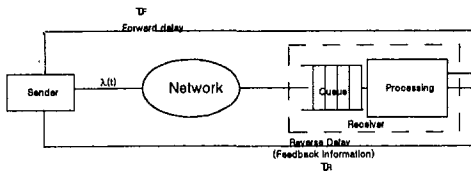


그림 3 모델링 구조

- Q_{th} : 큐에서의 임계값
- Q_s : 큐 상태를 나타내는 feedback 정보
- $q(t)$: 시간 t 에서의 큐의 길이
- $\lambda(t)$: 시각 t 일때의 전송자에서의 데이터 전송률
- μ : 노드에서의 처리율
- τ_F : Sender에서 큐까지의 전파지연시간
- τ_R : 큐에서 receiver를 지나 sender까지의 지연시간

송신기에서 데이터를 보낼 때 Q_s 의 값은 '0'이며 큐의 상태에 따라 Q_s 의 값이 좌우된다.

Q_s 의 값은 시간 t 에서 큐 길이가 Q_{th} 보다 작아지는 순간 피드백 정보를 '0'으로 하여 전송률을 증가시키고 큐 길이가 Q_{th} 보다 커지는 순간 피드백 정보를 '1'이 되게하여 전송률을 감소시킨다.

- (1) $Q_s = '1'$ if $q(t) < Q_{th} \rightarrow q(t) \geq Q_{th}$
- (2) $Q_s = '0'$ if $q(t) > Q_{th} \rightarrow q(t) \leq Q_{th}$

Q 비교	전송률
$Q > Q_{th}$	증가
$Q < Q_{th}$	감소

표1. 큐의 상태에 따른 변화

네트워크에서의 모든 연결은 왕복 전파 지연 시간이 생기게 된다. 송신기에서 큐까지는 큐의 상태에 대한 정보를 알기 위해 보내는정방향 지연(forward delay)가 생기고 여기서 다시 큐의 현재 상태를 나타내는 정보를 받아 송신기에게 피드백 되는 역방향지연(reverse delay)가 생긴다. 송신기는 이 정보를 기반으로 네트워크의 흐름을 원활히 하기 위해 전송률을 조절하게 된다.

따라서 송신기에서는 피드백 정보, 즉 큐 길이와 임계값에 대한 상태 정보를 τ_R 만큼 전파지연된 후에 받게된다. 즉 송신기에서 받은 시간의 τ_R 만큼 이전의

수신기 큐 상태를 나타내게 된다.

따라서 송신기가 피드백을 받는 시간을 t_R 라 하면

$$q(t - \tau_R) \cong Q_{th} \quad (3)$$

라 할 수 있다.

또 송신기가 이 값을 받아 즉시 전송률을 조정했을 경우, 변경된 전송률이 수신기에 영향을 미치는 시간은 τ_F 후가 된다.

따라서 송신기에서 왕복 전파 지연시간이 포함되는 값 $(t - \tau_R)$ 부터 $(t + \tau_F)$ 까지의 전송률은 일정하다고 볼 수 있으며, 이 기간의 큐 길이의 변화를 estimate 할 수 있다.

전송률은 다음과 같이 제어한다.

i) 피드백에 의한 값 $Q_s = '1'$ 인 경우 큐의 길이가 임계값보다 큰 경우이므로 전송률을 감소시킨다. 수신기에서는 forward delay τ_F 후에 영향을 받는다.

$t_1 + \tau_F$ 후의 예측된 큐의 길이를 $q(t_1 + \tau_F)_{estimate}$ 라 하면 송신기 측에서의 전송률 감소는 τ 에 비례하거나 $q(t_1 + \tau_F)_{estimate}$ 의 함수이거나 일정한 비율로 할 수 있다.

τ 는 다음과 같이 정의된다.

$$\tau = \tau_F + \tau_R$$

ii) $Q_s = '0'$ 인 경우 큐의 길이가 임계값보다 작은 경우이므로 전송률을 증가시킨다. 증가는 $q(t_0 + \tau_F)_{estimate}$ 의 함수로 한다.

iii) 만약 큐가 Overflow될 경우 이를 알리는 NACK를 받는다면 전송률을 1/2로 감소시킨다.

iv) Queue empty(Underflow)되면 전송률을 증가시킨다.

4. 수학적 해석

본 장에서는 망에 의해 연결된 전송자와 수신자 사이의 단일연결을 모델링하여 해석하였다. 망은 하나의 병목노드 및 전파지연에 의해서 그림 2에서와 같이 모델링하였다.[7,11]

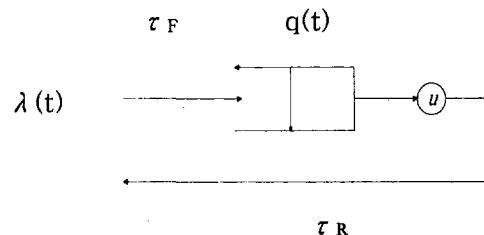


그림 4. 단일연결 모델

- $q(t)$: 시각 t 일 때의 병목노드에서의 큐길이
- $\lambda(t)$: 시각 t 일때의 전송자에서의 데이터 전송률

- μ : 병목노드에서의 처리율
- τ_F : 전송자에서 병목노드까지의 전파지연
- τ_R : 병목노드에서 수신자를 거쳐 전송자까지의 피드백 정보 전파지연

본 연구에서 제안된 메커니즘은 식 (1-2)로 정의된다.

$$\frac{dq(t+\tau_F)}{dt} = \begin{cases} 0 & \text{if } q(t+\tau_F)=0 \text{ and } \lambda(t)-\mu < 0 \\ \lambda(t)-\mu & \text{otherwise} \end{cases} \quad (1)$$

$$\lambda(t) = \begin{cases} a[q(t+\tau_F)_{estimate} - Q_{th}] + b & \text{if } q(t-\tau_R) < Q_{th} \\ c[q(t+\tau_F)_{estimate} - Q_{th}] + d & \text{if } q(t-\tau_R) \geq Q_{th} \end{cases} \quad (2)$$

시스템은 초기 상태, 준비 상태, 안정 상태의 세 가지 상태로 이루어진다.

초기 상태($0 \leq t \leq t_0$)는 식 (3)에 의해서 정의된다.

$$\begin{cases} \lambda(t) = \alpha & \alpha; \text{초기에만사용} \\ q(t) = 0 \end{cases} \quad (3)$$

준비 상태($t_0 \leq t \leq t_1$)는 그림 2에서 첫번째 사이클에 해당하는 구간으로 준비 상태가 시작되는 시점에서 큐는 아직 비어있다.

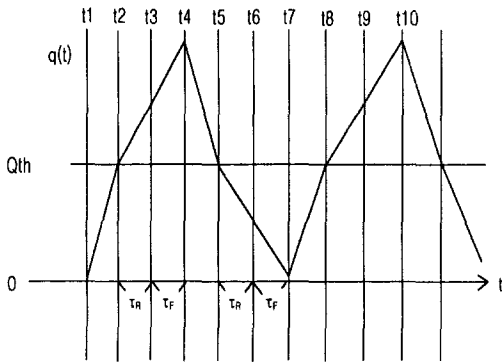


그림 3. 각 구간별 상태

- $t_1 = 0$: 큐에 도착시작
- $t_2 = Q_{th}$ 를 넘는 시점 \uparrow : $q(t_2) = Q_{th}$ (포함)
- $t_3 = t_2 + \tau_R$: $Q_S = 1$ 을 송신기가 받는 시점(전송률 감소)
- $t_4 = t_3 + \tau_F$: 변화된 전송률이 수신기영향
- $t_5 = Q_{th}$ 를 넘는 시점 \downarrow : $q(t_5) = Q_{th}$ (포함)
- $t_6 = t_5 + \tau_R$: $Q_S = 0$ 을 송신기가 받는시점(전송률 증가)
- $t_7 = t_6 + \tau_F$: 증가된 전송률이 수신기에 영향
- $t_8 = Q_{th}$ 를 넘는 시점 \uparrow : $Q_S = 1$ 발생

4.1 구간별 수학적 해석

여기서는 각 구간별 수학적으로 해석하였다.

시간 t_0 일때의 큐의 길이는 0으로 시작하며

$$t_0 = 0 \quad q(t_0) = 0$$

전송자에서의 전송률이 수신자에게 영향을 미치는 시점은 전파지연시간 τ_F 가 흐른 후이다.

$$t_1 = \tau_F$$

이 때의 큐의 길이는

$$q(t) = \int_{t_2}^t (\lambda(t-\tau_F) - \mu(t)) dt + Q_{th}$$

$$t_2 = Q_{th} \text{를 넘는 시점 } \uparrow \quad q(t_2) = Q_{th}$$

큐의 길이가 Q_{th} 와 교차하여 생긴 상태정보가 전송자에게 전달되는 시점은 전파지연시간 τ_R 이 흐른 후이며

$$t_3 = t_2 + \tau_R$$

이 때의 전송률은

$$\lambda(t) = \mu(t) - \frac{q(t) - Q_{th}}{\tau}$$

변화(감소)시킨 전송률이 큐에 영향을 미치는 시점은 전파지연 시간 τ_F 후이며

$$t_4 = t_3 + \tau_F$$

$$t_4 \text{ 이후} : q(t) = \int_{t_4}^t (\lambda(t-\tau_F) - \mu(t)) dt + Q_{th}$$

$$t_5 = Q_{th} \text{를 넘는 시점 } \downarrow \quad q(t_5) = Q_{th}$$

$$t_5 \text{ 이후} : q(t) = \int_{t_5}^t (\lambda(t-\tau_F) - \mu(t)) dt + Q_{th}$$

$$t_6 = t_5 + \tau_R$$

$$\lambda(t) = \mu(t) - \frac{q(t) - Q_{th}}{\tau}$$

$$t_7 = t_6 + \tau_F$$

$$t_7 \text{ 이후} : q(t) = \int_{t_7}^t (\lambda(t-\tau_F) - \mu(t)) dt + Q_{th} + q(t_7)$$

$$t_8 = Q_{th} \text{를 넘는 시점 } \uparrow \quad q(t_8) = Q_{th}$$

그 이후로는 안정상태가 되어 위의 상태를 반복하게 된다.

$$t_9 = t_3 \sim t_8 \text{ 반복}$$

5. 성능 분석

여기서는 4장에서 제안된 모델과 수학적 해석을 근거로 컴퓨터를 이용한 모의 실험을 하였으며 제안된 흐름제어기법에 대하여 평가한다.

먼저 이 논문에서 사용되는 몇 개의 매개변수에 대한 조절을 하면서 모의 실험을 하였다. 모의 실험에서

사용한 매개 변수는 다음과 같다.

- λ_0 (초기 전송률 평균) :
 7packets/msec (Poisson분포)
 ⇒ 도착간격은 Exponential분포
- τ (전파지연) : 0.1msec
- μ (처리율) : 5 packets/msec
- Buffer size : 20
- Q_{th} (임계값) : 6 ~ 15

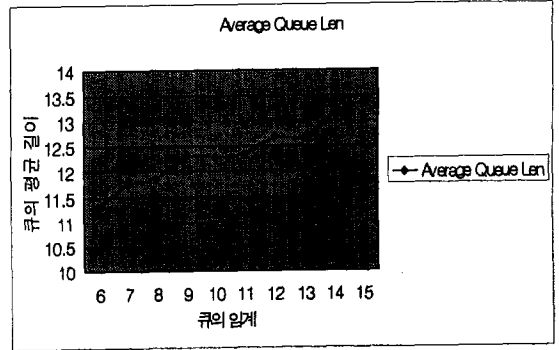


그림5. 큐의 임계값에 따른 평균 큐 길이

초기 전송률이 처리율보다 크게 설정한 이유는 큐의 길이가 Q_{th} 를 넘게 하기 위함이며 모델을 간단히 하기 위해서 전파 지연은 0.1msec로 일정하다고 가정하였다.

모의 실험을 위해 사건 기반의 모의 실험언어인 SMPL을 사용하였으며 그 블록도는 다음과 같다.

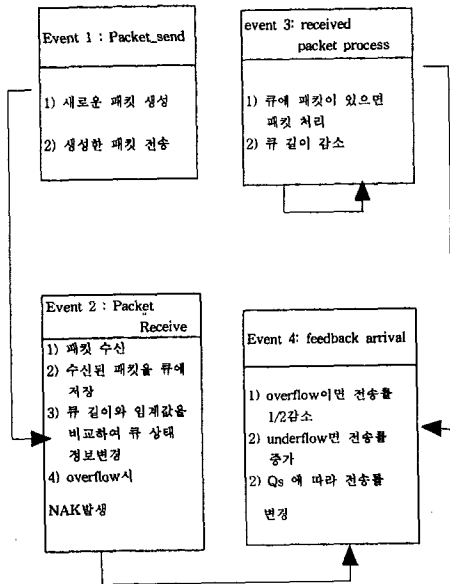


그림4. 모의 실험 블록도

그림 5는 큐의 임계값을 변화시키면서 큐의 평균 길이를 관찰한 것이다. 큐의 길이에 따라 큐의 평균 길이가 변화하였으며 임계값이 13일 때 큐의 평균 길이가 가장 임계값에 근사함을 알 수 있다. 그러나 임계값에 근사하다고 좋은 것은 아니며 패킷 손실률이 얼마나 되는가를 고려하여야 한다.

그림 6은 큐의 임계값에 따른 평균 전송률을 관찰한 것이다. 임계값이 커질수록 전송률이 비례해서 증가한다는 결론을 얻을 수 있다. 그러나 임계값이 너무 클 경우 오버플로우 신호로 인해 전송률을 1/2로 감소시키는 경우가 많이 발생하기 때문에 오히려 전송률이 감소한다.

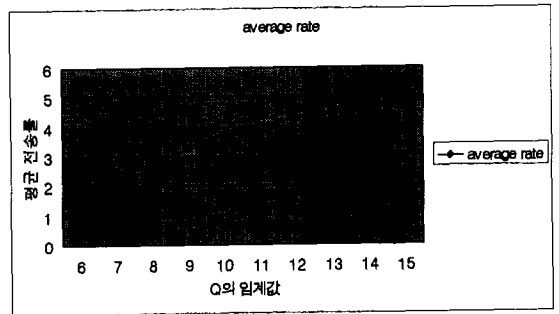


그림6. 큐의 임계값에 따른 평균 전송률

그림 7은 Q_{th} 에 따른 패킷의 손실률을 나타낸 것이다. 임계값이 작을 경우 오버플로우의 가능성이 적어 패킷의 손실이 적지만 임계값이 커짐에 따라 오버플로우의 경우가 많아짐으로서 패킷의 손실률이 커짐을 알 수 있다.

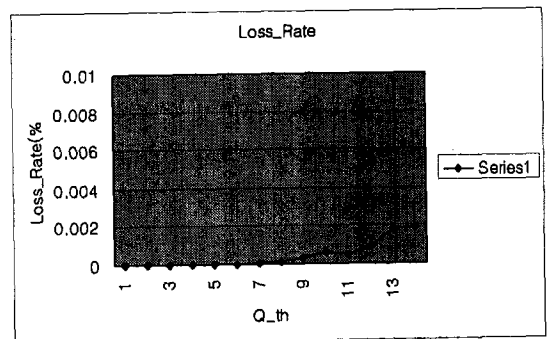


그림 7. 각 $Q_{th}(1-15)$ 까지의 loss rate

시뮬레이션 결과 그림 8에서 보는바와 같이 임계값을 변화시킬 때 거의 일정한 처리율을 보이고 있어 제안된 방법이 매우 효율적인 방법임을 알 수 있었다.

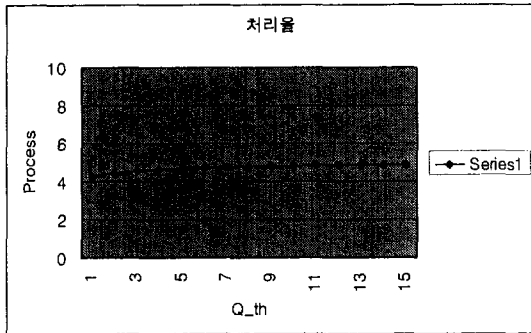


그림 8. 임계값의 변화에 따른 처리율

6. 결 론

본 논문에서는 네트워크 환경에서 큐의 상태에 근거하여 최선형 트래픽의 전송률을 직접 제어하여 망 이용의 효율성과 전송 성능을 향상시켜 자원의 활용을 높이는 방안을 제안하였다. 제안된 방식은 수신기에서 버퍼의 오버플로우, 언더플로우, 큐 길이의 관계를 근거로 하여 전송률을 제어하고 큐 길이가 Q_{th} 통과시 계속 발생하는 같은 신호는 무시하고 큐 상태정보 Q_{th} 값이 변화하는 순간에만 전송률을 제어하여 지나친 전송률의 감소나 증가를 방지하였다. 전송자는 피드백 정보에 기반하여 망 상황에 알맞는 수학적 해석에 근거한 양으로 데이터의 전송률을 조정한다. 이러한 결과로서 제안된 기법은 기존의 방법에 비해 수신자의 전송률, 버퍼, 대역폭 등의 망 자원을 효율적으로 이용한다는 사실을 알 수 있으며 수학적 해석 및 시뮬레이션 등을 통하여 증명하였다. 신뢰성 있는 데이터 전송에 있어서 보다 연구해야 할 과제로는 보다 복잡한 멀티캐스트 등에서 전파지연시간에 의한 큐 상태 변화에 민감하지 못한 부분이나 오버플로우시의 적절한 전송률의 감소, 실시간 통신에서의 흐름 제어와 유선과 무선이 통합된 네트워크에서의 망 자원을 확보하고 전송률을 조정하는 부분에 대한 연구가 필요하다.

참 고 문 헌

[1] J. J. Bae, and T. T. Suda, "Survey of Traffic Control Schemes and Protocols in ATM Networks." Proc. of the OEEE. vol. 79, no. 2, pp. 170-189. Feb. 1991.
 [2] C. M. Aras, J. F. Kurose, D. S. Reeves. and H.

Schulzrinne, "Real-Time Communication in Packet-Switched Networks," Proc. of the IEEE, vol. 82. no. 1. pp. 122-139, Jan. 1994.
 [3] H. T. Kung, R. Morris, T. Charuhas, and D. Lin, "Use of Link-by-link Flow Control in Maximizing ATM Network sPerformance:Simulation Results." Proc. IEEE Hot Interconnects Symposium '93. Palo Alto, USA, Aug. 1993.
 [4] P. P. Mishra and H. Kanakia, "A Hop by Hop Rate - based Congestion Control Scheme," Proc. ACM SIGCOMM '92, pp. 112-123, Aug. 1992.
 [5] N. Yin and M. Hluchyj, "On Closed-Loop Rate Control for ATM Cell Relay Networks," Proc. IEEE INFOCOM '94, pp. 99-109, June. 1994.
 [6] K. K. Ramakrishnan and R. Jain, "A Binary Feedback Scheme for Congestion Avoidance in Computer Networks with a Connectionless Network Layer," Proc. ACM SIGCOMM '88, pp. 303-313, AUG. 1988.
 [7] C. A. Eldrge, "Rate Controls in Standard Transport Layer Protocols," ACM Computer Communication Review. vol. 22, no. 3, pp, 106-120, July 1992.
 [8] Defense Advanced Research Projects Agency, "Transmission Control Protocol," RFC-793, Sep-tember, 1981.
 [9] V. Jacobson, "Congestion Avoidanced and Control," Proceedings of ACM Sigcom, Computer Communication Review, vol. 18, No. 4, pp. 314-329, 1988.
 [10] Lixia Zhang, Stephen Deering, Deborah Estrin, Scott Shenker, Daniel Zappala, "RSVP:A New Resource ReSer Vation Protocol," IEEE Network, pp. 8-17, September 1993.
 [11] J. C. Bolot and A. U. Shanker, "Dynamical Behavior of Rate-Based Flow Control Mechanisms." ACM Computer Communications Review, vol. 20, no. 2, pp. 35-49, April 1990.
 [12] M. Yamamoto, Y. Sawa, S. Fukatsu. H. Ikeda, "NAK-based Flow Control Scheme for Reliable Multicast Communications", IEEE, 1998
 [13] 서주하, 유소영, "신뢰성 있는 멀티캐스트 통신에서는 NACK피드백을 이용한 흐름제어", 한국정보과학회 봄 학술발표논문집, 1999