

論文 92-29A-11-1

DQDB망 소거노드 알고리즘

(An Erasure Node Algorithm for DQDB Networks)

全炳千*, 金大榮**

(Byung Chun Jeon and Dae Young Kim)

要約

DQDB 망에서는 목적지 노드에서 읽혀진 슬롯을 제거하는 소거노드를 사용할 수 있다. 소거 노드의 주요한 기능은 하위노드에서 사용된 재생슬롯과 동일한 수의 request를 제거하는 것이다. 우선 순위를 가지는 망에서는 추가적으로 다음과 같은 두 가지, 즉 (1) 재생된 슬롯이 어느 우선순위의 세그먼트 전송에 사용되는지 정확하게 추측하고 (2) 제거하고자 하는 우선순위의 request를 얼마나 빨리 찾느냐 하는 것을 해결하여야 한다. 본 논문에서는 request의 제거방식의 두 가지 소거노드 알고리즘을 제안하였다. 이 알고리즘은 저부하에서 request를 과다하게 제거하나 최고 부하에서는 최적제거 특성을 얻을 수 있다. 시뮬레이션을 통하여 저부하시에는 기존의 알고리즘과 유사한 액세스 지연을 나타내며 고부하시에는 보다 개선된 성능을 보임을 확인하였다.

Abstract

In the DQDB networks, erasure nodes may be allocated on the bus and can release slots which already had been read by destination nodes. An important function of the erasure node must be to eliminate as many requests as erased slots used by downstream nodes. In the networks with priorities, following two additional issues must be solved : (1) accurately to guess for which priority the erased slots is to be used, and (2) to find in a short period a request of that priority to cancel. In this paper, we propose two erasure node algorithms based on elimination of the requests. In these algorithms, the requests are eliminated excessively at low load but optimum elimination can be achieved at full load. Simulation results show that these two algorithms have similar access delay to that of existing algorithms at low load and better performance at high load.

I. 서론

IEEE 802.6 위원회에서 MAN(Metropolitan Area

Network) 표준으로 결정된 DQDB(Distributed Queue Dual Bus) 망은 두개의 단방향 버스를 가지며, 버스 양끝의 헤드노드는 일정길이의 슬롯을 발생하고 여러 노드가 이 슬롯을 이용하여 세그먼트를 전송하도록 한다. DQDB망은 버스의 전달지연이 하나의 슬롯 전송 시간보다 짧을 때 모든 노드에서 발생한 세그먼트는 하나의 큐에 넣어져 발생된 순서대로 전송되는 것과 같은 효과를 낼 수 있으므로 액세스 지연시간이 짧으며, 버스의 대역폭을 거의 100% 사용할 수 있는 장점을 가지고 있다. 또한 3가지 우선순위의 패킷 데이터와 등시성 데이터 전송

*正會員, 韓國電子通信研究所, ATM 시스템 研究室 (ETRI, ATM System Section)

**正會員, 忠南大學校 情報通信 工學科 (Dept. Infor. Comm. Eng., Chungnam Nat'l Univ.)

接受日字 : 1992年 1月 6日

을 위한 기능을 갖추고 있어 추후 다중 매체 통신망으로 사용될 것으로 예상된다.^[1,12]

따라서 DQDB망에서는 목적지 소거 방식과 같이 버스의 이용률을 증가시키면서도 망의 지연이 크게 증가하지 않도록 특정노드에서만 슬롯을 재생하도록 하는 소거노드 방식을 채택하였다. 즉 보통의 노드는 지연없이 다음 노드로 슬롯을 통과시키며 슬롯의 목적지가 자신의 노드이면 다음 슬롯의 PSR(Previous Slot Received)비트를 1로 바꾸어 이전 슬롯이 사용된 슬롯임을 표시하며, 소거노드는 1 슬롯 길이만큼의 FIFO를 가지고 지나가는 슬롯을 지연시키면서 PSR비트를 감시하여 PSR 비트가 1이면 FIFO에 저장된 이전 슬롯의 busy 비트를 0으로 만들어 전송한다. 이와같은 방법은 망에 연결된 일부의 노드만이 소거노드로 동작하므로 전달지연이 크게 증가하지 않으면서도 버스의 이용률을 높일 수 있으며, 모든 노드가 슬롯 재생 기능을 가지지 않아도 되는 장점이 있다. 따라서 소거노드의 수에 따른 버스의 이용률 증가 및 소거노드의 최적위치를 정하기 위한 연구와 소거노드를 사용하였을 경우 데이터 분포에 따른 버스 이용률 증가에 대한 연구가 이루어졌다.^[5,6,7,8]

소거노드에서 발생된 빈 슬롯에 의해 확장된 대역폭을 모든 노드가 공정하게 사용하도록 하기 위하여 소거노드가 빈 슬롯을 발생시킬 때마다 지나가는 request를 제거하거나 negative request를 전송하는 방법이 제안되었다.^[6,9,10,11] 그러나 이러한 방법은 소거노드에 의하여 증가된 대역폭을 모든 노드가 공정하게 나누어 사용하지 못하도록 높은 부하가 가해질 경우 노드간 액세스 지연의 차이가 나타난다.

본 논문에서는 소거노드가 망에 미치는 효과와 기존 소거노드 알고리즘에서 액세스지연의 불공정성이 나타나는 원인을 분석하고, 이를 보완한 두가지의 소거노드 알고리즘을 제안하며 시뮬레이션을 통하여 그 성능을 비교, 분석하였다.

II. 소거노드의 효과

소거노드가 1개, 2개, 그리고 모두 소거노드로 동작하는 경우의 매체이용률이 최대가 되는 소거노드의 최적위치와 최대 매체 이용률에 대하여 알아본다. DQDB에서 세그먼트의 전송방법은 다음과 같이 나눌 수 있다.

첫째, 특정 노드에서 발생된 세그먼트는 목적지 노드의 위치에 상관없이 양쪽버스에 모두 전송하는 방법으로, 목적지 노드의 상대적 위치를 알 수 없을 때 또는 망에 가해지는 부하가 적어 전송효율이 낮아도 되는 경우에 사용될 수 있다.

둘째, 발생된 세그먼트를 목적지 노드로 향하는 버스에만 전송하는 방법으로 첫째 방법에 비하여 전송효율이 2배로 높아진다.

모든 노드에서 발생하는 세그먼트의 양이 동일하고 목적지가 모든 노드로 고르게 분포되어 있다면 첫번째 방법은 발생된 모든 세그먼트를 양쪽의 노드로 전송하여야 하므로 노드의 위치에 상관없이 동일한 양의 세그먼트를 전송하지만, 둘째 방법은 노드의 위치에 따라 버스로 전송하는 세그먼트의 양이 달라진다. 소거노드가 1개와 2개일 경우에는 목적지노드로 향하는 버스에만 세그먼트를 전송하는 방법, 모든 노드가 소거노드로 동작하는 경우에는 두가지 방법에 대하여 소거노드의 최적위치 및 매체이용률에 대하여 알아본다.

1. 소거노드가 1개 일때

발생된 세그먼트를 목적지 노드의 위치로 향하는 버스에만 전송할 경우 버스로 전송되는 세그먼트의 transmission density function은

$$f_{tx} = 2\lambda(1-x) \quad 0 \leq x \leq 1 \quad (1)$$

reception density function은

$$f_{rx} = 2\lambda x \quad 0 \leq x \leq 1 \quad (2)$$

가 된다.

2는 소거노드가 없을 경우에 한쪽 버스의 최대 처리율을 나타낸다. <그림 1>과 같이 k점에 소거노드가 위치한다고 가정했을 때 사용된 슬롯의 분포는

$$f_{div} = \int_0^x f_{tx} dx \quad 0 < x \leq k \\ = \int_0^x f_{tx} dx - \int_0^x f_{rx} dx \quad k < x \leq 1 \quad (3)$$

소거노드의 위치 k가 최적점이라면 $F_{d(k)}=1$, 및 $F_{d(1)}=1$ 이 되어야 한다. 즉 k점에 위치한 소거노드에서 재생시킨 빈 슬롯이 k점 이후의 노드에서 모두 사용될때 k점이 최적의 소거노드 위치가 된다. 따라서 <그림 1>과 같이 소거노드에서 재생된 빈 슬롯의 수를 나타내는 빗금친 부분 B와 소거노드 이후의 노드들이 사용하는 빈 슬롯의 수를 나타내는 빗금친 부분 A는 동일한 면적을 가져야 한다.

$$\int_0^x f_{rx} dx = \int_0^x f_{tx} dx \quad k \quad (4)$$

(4)에 (1)과 (2)를 대입하면 $k=0.5$ 가 된다. 따라서 소거노드가 1개일때 최적위치는 망의 중앙부임을 알 수

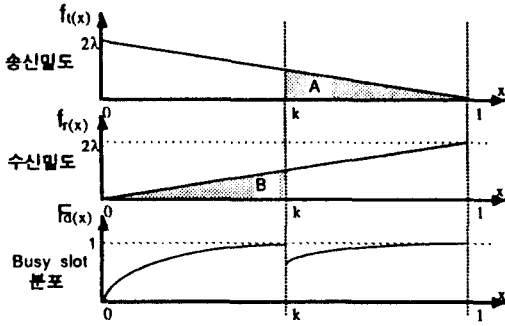


그림 1. 소거노드가 1개일때 busy slot의 분포
Fig. 1. Busy slot distribution in the DQDB network with one erasure node.

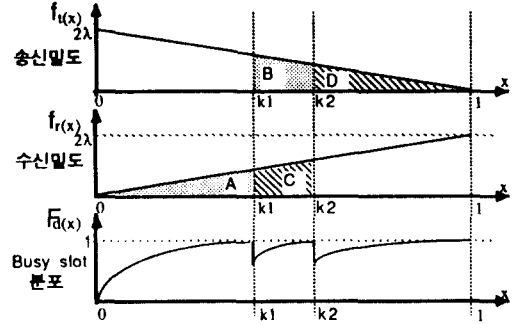


그림 2. 소거노드가 2개일때 busy slot의 분포
Fig. 2. Busy slot distribution in the DQDB network with two erasure nodes.

있다.

$$f_{d(k)} = \int_0^{0.5} f_{t(x)} dx = 1 \quad (5)$$

(5)에 (1)을 대입하면 $\lambda=1.333$ 이 된다. 즉 소거노드 하나를 사용하면 소거노드가 없는 경우에 비하여 이용률이 33.3% 증가됨을 알 수 있다.

(2) 소거노드가 2개 일때

소거노드가 k_1, k_2 에 있을 경우 사용된 슬롯의 분포는 아래의 (6)식과 같으며, <그림 2>와 같이 $F_{d(k_1)}=1, F_{d(k_2)}=1, F_{d(1)}=1$ 이 되어야 최대의 매체이용률을 얻을 수 있다.

$$\begin{aligned} F_{d(x)} &= \int_0^x f_{t(x)} dx \quad 0 < x \leq k_1 \\ &= \int_0^x f_{t(x)} dx + \int_0^{k_1} f_{r(x)} dx \quad k_1 < x \leq k_2 \\ &= \int_0^x f_{t(x)} dx + \int_0^{k_1} f_{r(x)} dx + \int_{k_1}^{k_2} f_{r(x)} dx \quad k_2 < x \leq 1 \end{aligned} \quad (6)$$

위의 조건이 만족될 경우 <그림 2>의 k_1 에 위치한 소거노드가 재생하는 빈 슬롯(빗금부분 A)은 k_1 과 k_2 사이에 위치한 노드들이 모두 사용하므로 $F_{d(k_2)}=1$ 이 되며, k_2 에 위치한 소거노드가 재생하는 빈 슬롯(빗금부분 C)은 k_2 이후의 노드들이 모두 사용한다. 즉 빗금부분 A의 면적과 빗금부분 B의 면적의 크기는 동일하며 또한 빗금부분 C의 면적과 빗금부분 D의 면적의 크기는 동일하다. 즉

$$\int_0^{k_1} f_{r(x)} dx = \int_{k_1}^{k_2} f_{t(x)} dx \quad (7)$$

$$\int_{k_1}^{k_2} f_{r(x)} dx = \int_{k_2}^1 f_{t(x)} dx \quad (8)$$

위식에 (1)과 (2)를 대입하면 $k_1=0.414, k_2=0.586$

을 얻을 수 있다. 소거노드가 2개일때 매체 이용률은 (5)와 동일한 방법으로 얻어진다.

$$F_{d(k)} = \int_0^{0.414} f_{t(x)} dx = 1 \quad (9)$$

즉 $\lambda=1.523$ 이 된다.

단지 두개의 소거노드를 사용하므로써 소거노드를 사용하지 않은 경우에 비하여 매체 이용률을 52.3% 향상시킬 수 있다.

3. 모든 노드가 소거노드로 동작하는 경우

모든 노드가 소거노드로 동작할 경우 즉 목적지 소거방식으로 동작할 경우 노드에서 발생된 모든 세그먼트를 목적지 노드의 위치에 관계없이 양쪽 버스에 전송할때 Transmission density function은

$$f_{t(x)} = \lambda \quad (10)$$

목적지 소거방식으로 동작할때 Reception density function은

$$f_{r(x)} = \lambda x \quad 0 \leq x \leq 1 \quad (11)$$

가 된다.

따라서 버스점유 분포는 다음과 같이 표시된다.

$$F_{D(x)} = \int (f_{T(x)} - f_{R(x)}) dx = \lambda x - 0.5\lambda x^2 \quad (12)$$

<그림 3> (가)에서와 같이 버스의 종단에서 버스 점유율이 1일때 최대의 이용률을 나타내므로 $F_{D(1)}=1$ 을 (12)에 대입하면 $\lambda=2$ 가 됨을 알 수 있다. 즉 모든 노드가 목적지 노드로 동작할때 2배의 버스 이용률을 가진다. 그러

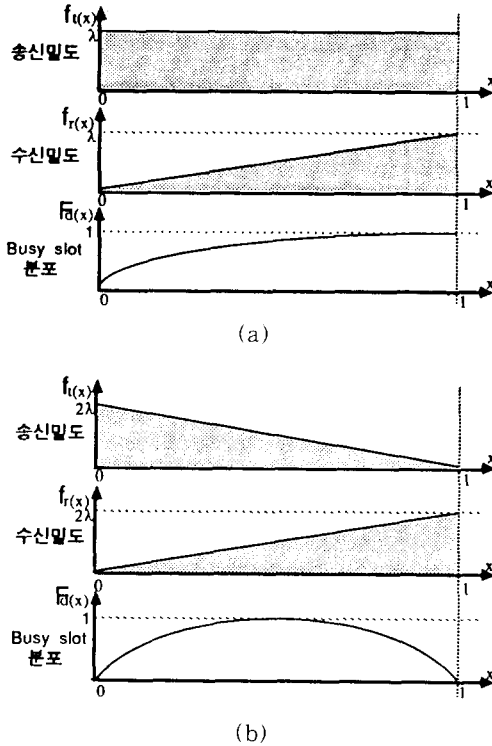


그림 3. 목적지 소거 방식으로 동작할 때의 Busy Slot 분포
 (a) 세그먼트를 양쪽버스 모두로 송신할 경우
 (b) 세그먼트를 목적지 노드로 향하는 버스에
 만 송신할 경우

Fig. 3. Busy slot distribution in the DQDB network with destination release.

나 이 방법은 동일한 세그먼트를 양쪽의 버스로 전송하므로 망 전체의 최대 이용률은 2가 된다.

발생된 세그먼트를 목적지 노드의 위치로 향하는 버스에만 전송할 경우, 전송된 모든 세그먼트는 목적지에서 소거되므로 버스 종단에서의 사용된 슬롯의 점유율은 0이 된다. 이 경우의 transmission density function은 (1)과 동일하며, reception density function은 (2)와 동일한 값을 가진다. 따라서 버스 점유 분포 $F_{D(x)}$ 는

$$F_{D(x)} = \int (f_{T(x)} - f_{R(x)}) dx = 2\lambda x(1 - x) \quad (13)$$

이 된다.

(13)에서 버스 점유 분포가 최대가 되는 점, 즉 $x=0.5$ 의 점에서 $F_{D(0.5)} = 1$ 을 대입하면 $\lambda=2$ 가 된다. 즉 최대 버스 이용률이 2가 되며 망전체의 이용률은 4가 됨을 알 수 있다. <그림 3> (나)는 목적지 노드로 향하는 버스에만 세그먼트를 전송할 경우의 버스 transmission density, reception density 및 버스 점유 분포를 나타낸다.

III. 기존 소거노드 알고리즘의 문제점

1. Request 제거 방법

우선 순위가 없는 경우 소거노드는 버스 A에 $PSR=1$ 인 슬롯을 만나면 PSR 비트를 0으로 바꾸고, 버퍼에 저장된 이전 슬롯의 busy 비트를 0으로 만든 다음 SR(slot release) 계수기를 증가시킨다. 그러나 RQ 계수기와 CD 계수기의 값이 모두 0일 경우에는 SR 계수기를 증가시키지 않는다. SR 계수기의 값이 0보다 크면 버스 B에 지나가는 슬롯의 request 비트가 1인 슬롯을 찾아 request 비트를 0으로 바꾼 후 계수기를 1 감소시킨다. 여러 우선 순위를 가지는 망에서 소거노드는 SR 계수기를 우선순위별로 가지고 있으며, 빈 슬롯을 발생시킬 때마다 $p=\max\{i : RQ(i) + CD(i) > 0\}$ 우선순위의 SR 계수기를 증가시킨다. 여기서 i 는 우선순위를 나타내며 모든 우선순위의 RQ 계수기와 CD 계수기의 값이 0일 경우 어느 우선순위의 SR 계수기도 증가시키지 않는다. SR 계수기의 값이 0보다 크면 버스 B에 지나가는 해당 우선순위의 request를 제거한다.^[11]

실제 소거노드는 일반 노드의 기능에 소거기능을 추가한 형태이므로, 소거노드가 전송할 세그먼트를 가지고 있으며 그 세그먼트의 우선순위 h 가 p 보다 높다면 소거노드에서 만들어진 빈 슬롯은 소거노드에서 우선순위 h 의 세그먼트를 전송하는데 사용될 것이므로 제거할 $i=\max\{p, h\}$ 우선순위의 request를 제거하는 방법을 제안하였다.^[9] 위와 같은 방법은 모두 우선순위의 RQ 계수기와 CD 계수기의 값이 0일 경우^[11], 또는 RQ 계수기와 CD 계수기가 모두 0이고 소거노드가 전송할 세그먼트를 가지고 있지 않을 경우^[9], 빈 슬롯을 발생하여도 하위노드에서 사용되지 않을 것으로 가정하여 request를 제거하지 않지만, 실제에 있어서는 소거노드에서 발생된 빈 슬롯이 다음 노드로 진행되는 동안 하위쪽에 위치한 노드에서 세그먼트가 발생되면 빈 슬롯은 세그먼트의 전송에 사용되므로 request의 제거가 불안정하게 된다.

이와같은 문제점을 보완하기 위하여 <그림 4>와 같이 소거노드에서 빈 슬롯을 발생할 때마다 그때의 시간을 큐에 넣고, 큐의 다른쪽 끝에서는 경과된 시간을 측정하여 timeout되면 큐에서 제거하며, timeout되지 않았으면 버스 B에 지나가는 request를 제거한다.^[10] 여기에서 timeout값은 소거노드에서 다음 노드까지 전달 지연의 2배 이내의 값으로 하여, 소거노드에서 빈 슬롯을 발생시킨 순간에 $RQ=CD=0$ 이라도 빈 슬롯이 다음 노드에서 사용되면 request가 제거된다. 이 방법은 소거노드에서 빈 슬롯을 발생시킨 순간 $RQ > 0$ 또는 $CD > 0$ 인 경우 시간의 지연에 관계없이 request를 제거하여야 하나 timeout 시간내에 request가 도착하지 않으면 제거되지

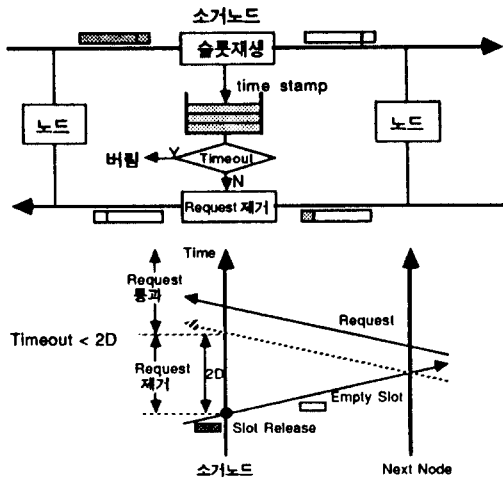


그림 4. Yokotani의 소거노드 알고리즘
Fig. 4. Yokotani's erasure node algorithm.

못한다.

2. Negative Request를 사용하는 방법

<그림 5>와 같이 슬롯의 액세스 제어 영역에 reserve request 비트를 할당하고, 소거노드가 빈 슬롯을 재생시킬 때마다 버스 B에 지나가는 negative request 비트를 1로 바꾸되 동일 슬롯내에서 request 비트와 negative request 비트가 동시에 존재하게 될 경우에는 두 비트를 모두 0으로 클리어한다. 각 노드는 버스 B에 negative request가 지나갈 때마다 $CD > 0$ 이면 CD 계수기를 1 감소시키고, $CD=0$ 이고 $RQ > 0$ 이면 RQ 계수기를 1 감소시키고, $RQ=CD=0$ 이면 negative request 비트를 0으로 만든다. 즉 버스 B에 지나가는 negative request는 버스 A에 지나가는 빈 슬롯과 동일한 효과를 발생시키며, $CD=RQ=0$ 인 경우 소거노드에서 빈 슬롯을 발생시켜도 하위 노드에서 빈 슬롯의 요구가 없으므로 사용되지 않은 것으로 가정하여 negative request가 상위쪽 노드에 영향을 미치지 않도록 제거하는 방법이다.¹⁶⁾

이 알고리즘에서 negative request는 빈 슬롯과 동일한 기능을 수행하므로 여러 우선순위를 가진 망에 적용할 경우 모든 우선순위의 CD 계수기와 RQ 계수기에 동일한 영향을 주도록 하고 모든 우선순위의 CD 계수기와 RQ 계수기가 0인 경우에 negative request가 제거되도록 한다.

하나의 슬롯에 negative request와 request가 동시에 존재하게 될 경우 노드들에게 주는 영향이 서로 상쇄되므로 두 비트를 모두 제거한다. DQDB 프로토콜은 송신 세그먼트와 동일한 수의 request가 발생하므로 소거

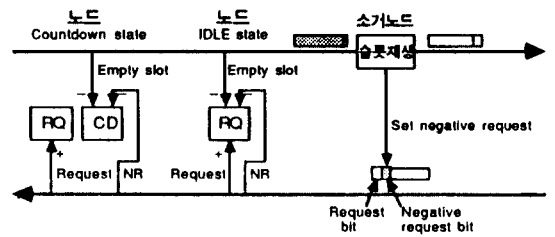


그림 5. Negative Request를 사용한 소거노드 알고리즘
Fig. 5. Erasure node algorithm using negative requests.

노드에 의하여 대역폭이 증가된 만큼 request를 전달하는 대역폭도 증가되어야 하므로 request와 negative request를 상쇄시키는 것은 타당하다. 그러나 여러 우선순위를 가지는 망에서 동일 슬롯내에 request와 negative request가 동시에 존재할 경우, request는 우선순위를 가지고 있으므로 우선순위를 가지지 않은 negative request와 그대로 상쇄시킬 수 없게 된다. 따라서 특정 우선순위의 트래픽이 버스 대역폭보다 크게 발생될 경우, 소거노드에 의하여 대역폭이 증가되어 이를 수용할 수 있어도 세그먼트와 동일한 갯수로 발생되는 request는 버스 대역폭 이상을 수용할 수 없으므로 노드간 대역폭 및 액세스지연에 불균형이 초래된다.

IV. 본 논문에서 제안한 소거노드 알고리즘

이상적인 소거노드 알고리즘이 되기 위하여는 소거노드에서 발생시킨 빈 슬롯들 중 세그먼트의 전송을 위하여 사용된 슬롯의 수와 제거된 request의 수가 동일하여야 하며, 또한 빈 슬롯을 발생시킨 소거노드는 신속하게 request를 제거하여야 한다. 여러 우선순위를 가지는 경우, 소거노드에서 발생된 빈 슬롯이 세그먼트 전달에 사용되면 사용된 우선순위의 request를 제거하여야 한다. 그러나 소거노드에서 발생된 빈 슬롯이 세그먼트 전달에 사용되었는지 또한 어느 우선순위 세그먼트 전달에 사용되었는지를 정확하게 측정하기는 불가능하므로 이에 근사적인 추정값으로 request를 제거하여야 한다.

본 논문에서는 소거노드에서 빈 슬롯이 발생되면 일단 사용되는 것으로 가정하여 <그림 6>과 이 SR 계수기를 증가시키고, SR 계수기의 값이 0보다 크면 버스 B에 지나가는 request를 제거하고 SR 계수기를 감소시킨다.¹³⁾ SR 계수기의 값이 SR_MAX 값을 초과하지 않도록 한다.

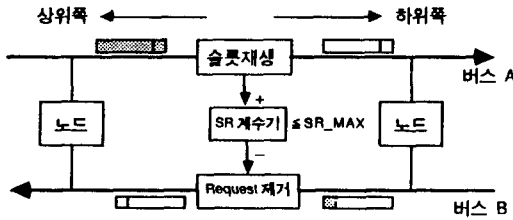


그림 6. 제안된 알고리즘
Fig. 6. Proposed algorithm.

본 알고리즘의 타당성을 입증하기 위하여 <그림 7>과 같이 소거노드에서 하위노드쪽으로 전달되는 빈 슬롯과 소거노드가 하위노드로부터 수신하는 request의 수에 따라 망의 상태를 3가지로 분류하고 각각의 동작에 대하여 고찰해 본다. 소거노드는 일반 노드와 같이 세그먼트의 송수신 기능과 함께 사용된 슬롯을 소거하는 기능이 추가된 노드를 의미하나 <그림 7>의 소거노드는 사용된 슬롯을 제거하는 기능만을 가진 노드로 가정한다.

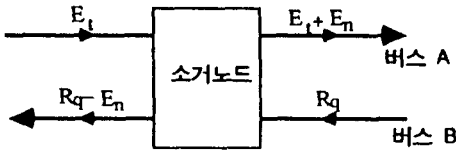


그림 7. 소거노드에서의 빈 슬롯과 request의 흐름
Fig. 7. Flow of empty slots and requests.

- (가) $E_n \leq R_q, E_t + E_n \gg R_q$
- (나) $E_n \leq R_q, E_t + E_n \geq R_q$
- (다) $E_n > R_q, E_t + E_n > R_q$

E_n : 단위 시간당 소거노드에서 발생시키는 빈 슬롯의 평균 갯수

E_t : 단위 시간당 소거노드가 상위 노드들로부터 수신하는 빈 슬롯의 평균 갯수

R_q : 단위 시간당 소거노드가 하위 노드들로부터 수신하는 request의 평균 갯수

상태 (가)는 소거노드에서 발생하는 빈 슬롯의 수가 하위노드로부터 수신하는 request의 갯수보다 적으며, 소거노드에서 하위노드로 전달하는 빈 슬롯의 수가 하위노드로부터 수신하는 request의 수보다 클 경우를 의미한다. 이는 하위노드들에서 발생된 request보다 하위노드로 전달되는 빈 슬롯이 더 많으므로 망에 가해지는 부

하가 적거나 소거노드의 위치가 최적위치로 부터 상위쪽으로 벗어나 있음을 나타낸다. 이때 소거노드는 소거노드에서 발생시킨 빈 슬롯만큼의 request를 제거하고 나머지 request를 상위쪽으로 전달한다. 이상적인 소거노드일 경우 발생된 빈 슬롯중 하위노드들에서 사용된 슬롯의 갯수만큼의 request를 제거하여야 하나, 상태 (가)와 같이 낮은 부하에서는 발생된 빈 슬롯중 일부의 슬롯만이 사용될 것이므로 소거노드는 최적의 경우보다 더 많은 수의 request를 제거하게 되어 상위쪽과 하위쪽 노드들간에 액세스지연의 불균형이 초래될 수 있다. 그러나 이 경우 낮은 부하상태이기 때문에 이러한 불균형은 매우 적을 것으로 판단된다.

상태 (나)는 상태 (가)에서 부하가 증가되어 소거노드가 하위쪽으로부터 수신되는 request의 수가 하위쪽으로 전달되는 빈 슬롯의 수와 거의 같은 정도가 된 경우를 의미한다. 이 경우에도 소거노드는 소거노드에서 발생시킨 빈 슬롯만큼의 request를 제거한다. 부하가 증가할수록 소거노드에서 발생시킨 빈 슬롯이 하위쪽 노드들에서 사용되는 비율이 높아지며, $E_t=0, E_n=R_q$ 인 경우 소거노드에서 발생시킨 빈 슬롯이 하위쪽에서 모두 사용되며 동일한 수의 request가 제거되므로 소거노드는 최적의 상태로 동작하게 된다.

상태 (다)는 소거노드에서 발생하는 빈 슬롯의 수가 하위노드로부터 수신되는 request의 수보다 많은 경우이며 이는 부하가 불균등하게 분포되었거나 소거노드의 위치가 최적위치로부터 하위쪽으로 벗어나 있음을 의미한다. 이 경우 하위쪽에서 벗어나 있음을 의미한다. 이 경우 하위쪽에서 발생된 request의 수보다 전달되는 빈 슬롯의 수가 많으므로 하위쪽에는 낮은 부하가 가해진 것처럼 동작하며 하위쪽에서 발생된 request는 소거노드에서 모두 제거된다. 이 경우 소거노드의 RS 계수기가 무한정 증가하는 것을 방지하기 위하여 SR 계수기의 값이 SR_MAX와 같으면 소거노드에서 빈 슬롯을 발생하여도 계수기를 증가시키지 않는다.

제안된 알고리즘은 낮은 부하시에는 최적값보다 더 많은 request가 제거되나 저부하 상태이므로 이에 의한 액세스 지연의 불균형은 무시할 수 있을 정도로 적으며, 부하가 증가할수록 최적상태에 가까워져 최대 부하일때 최적의 성능을 발휘한다. 따라서 이 알고리즘은 망의 대역폭을 증가시켜 더 높은 처리율을 얻기 위하여 사용하는 소거노드의 본래 목적에 부합되는 알고리즘임을 알 수 있다.

이와같은 알고리즘을 3단계의 우선순위를 가진 망에 적용하기 위한 두가지 방법을 제안한다.

1. 우선순위별 제거 방식

소거노드에서 빈 슬롯을 발생하면 첫번째에 도착하는

request를 제거하는 방법으로, 소거노드에서 발생된 빈 슬롯은 하위쪽 노드들이 발생한 request들중 가장 높은 우선순위에 해당하는 세그먼트 전송에 사용될것이므로 높은 우선순위의 request가 우선적으로 소거노드에 도착하도록 하고 이를 소거하는 방법이다. 즉 request의 제기시 빈 슬롯이 사용될 확률이 높은 우선순위를 선택하여 제거하는데 중점을 둔 방법이다. 이 방법에서는 <그림 8>과 같이 액세스 제어 영역의 request 비트로서 2비트만을 사용하고 00일 경우에는 request가 없는 경우 01,10,11일 때에는 각각 우선순위 1,2,3의 request임을 나타낸다. 각 노드에서는 우선순위로 REQ_i(i는 우선순위를 나타냄) 계수기를 가지고 있는데 이는 버스에 실어보낼 request의 수를 가지고 있다.

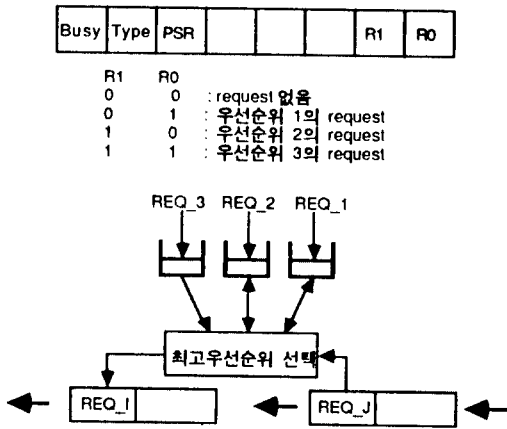


그림 8. 액세스제어 영역과 request 전달 동작
Fig. 8. Access control field and request transfer action.

각 노드는 버스 B에 지나가는 슬롯에 request가 실려 있지 않을 경우, 슬롯에 $\max\{i:REQ_i>0\}$ 우선순위의 request를 실어주고 해당 우선순위의 계수기를 1 감소시킨다. 슬롯에 request가 실려있을 경우, $\{i:request(i)\} \geq \max\{i:REQ_i>0\}$ 이면 request를 그대로 통과시키며, $\{i:request(i)\} < \max\{i:REQ_i>0\}$ 이면 request를 $\max\{i:REQ_i>0\}$ 우선순위로 바꾸어 주고 해당 우선순위의 REQ_i 계수기를 1 감소시키며, 본래 슬롯에 실려있던 request의 우선순위에 해당하는 REQ_P, $P=\{i:request(i)\}$, 계수기를 1 증가시킨다. 하나의 슬롯은 하나의 request만을 전달할 수 있으며, 모든 노드는 슬롯에 실려있는 request의 우선순위보다 높은 request가 발생하면 높은 우선순위로 교환하므로 슬롯의 request 영역에는 그 슬롯이 통과한 노드에서 발생된 request 중 가

장 높은 우선순위의 request가 실리게 된다.

소거노드는 <그림 6>과 같이 동작하되 SR 계수기의 값이 0보다 크면 버스 B에 지나가는 request를 우선순위에 상관없이 제거한다. 따라서 소거노드에서는 하나의 SR 계수기를 사용하면서도 슬롯을 재생할 때마다 자동적으로 가장 높은 우선순위의 request가 제거된다.

2. 먼저 도착하는 request 제거방식

DQDB 표준의 액세스 제어영역을 그대로 사용하며 소거노드의 RS 계수기 값이 0보다 크면 먼저 도착하는 request를 우선적으로 제거하는 방법으로 신속성에 중점을 둔 방법이다. 소거노드는 <그림 6>과 같이 동작하되 여러 우선순위를 가지는 경우에도 하나의 SR 계수기만을 사용하며, SR 계수기의 값이 0보다 크면 우선순위에 상관없이 버스 B에 지나가는 request를 제거한다. 하나의 슬롯에 여러 우선순위의 request가 동시에 존재할 경우 높은 우선순위의 request부터 RS 계수기가 허용하는 수만큼 제거한다.

DQDB에서는 송신된 세그먼트의 수와 request의 수는 동일하므로 우선순위별 request의 분포는 망에 가해진 우선 순위별 부하의 분포의 크기를 나타낸다. 이러한 특징 때문에 재생된 슬롯 갯수만큼의 request를 우선순위에 상관없이 제거하여도 제거된 request의 우선순위별 분포는 망에 가해진 우선순위별 부하의 분포와 동일하게 되며, 슬롯의 재생과 request의 제거 사이의 지연이 최소화된다. 또한 소거노드에서 재생된 슬롯이 재사용되어 전달하는 세그먼트의 우선순위 분포는 망에 가해진 우선 순위별 부하의 분포와 동일하게 된다. 따라서 이 방법은 특정지점에서는 우선순위 동작이 적절하게 이루어지지 않는 것처럼 보이나 재생된 슬롯이 전달하는 우선순위별 세그먼트의 수와 제거하는 우선순위별 request의 수의 평균값은 동일하게 되어 결과적으로 우선순위별 제거방법과 동일한 특성을 나타낸다.

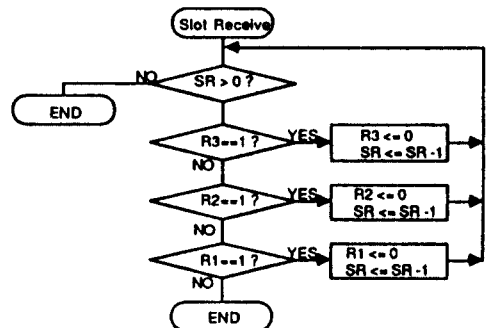


그림 9. 소거노드의 동작
Fig. 9. Erasure node action.

V. 시뮬레이션

본 논문에서 제안한 우선순위별 제거 방식(Eras-P)과 먼저 도착하는 request 제거방식(Eras-F)의 두가지 알고리즘의 성능과 RQ 계수기 또는 CD 계수기의 값이 0보다 클 경우에만 request를 제거하는 방법(RC) 및 negative request를 사용하는 방법(NR)의 성능을 시뮬레이션을 통하여 비교하였다. 4가지 알고리즘을 동일한 조건에서 시뮬레이션을 수행하여 액세스 시간, 즉 세그먼트가 발생한 순간부터 빈 슬롯을 사용하여 버스로 전송될때까지의 평균 시간을 비교하였다. 시뮬레이션은 SLAM-II를 사용하였으며 시뮬레이션 조건은 아래와 같다. 패킷의 발생은 모든 노드에서 동일하게 발생하나 한쪽 버스에 대해서만 시뮬레이션을 수행하였으므로 실제 세그먼트 송신밀도는 <그림 1>과 같은 분포를 가진다. 우선순위가 없는 경우와 2단계 우선순위를 가지는 경우에 대하여 성능을 측정하였으며, 2단계 우선순위를 가지는 경우 두 우선순위는 동일한 크기의 트래픽을 발생하는 것으로 가정하였다. 부하의 크기는 1 슬롯시간에 망에 가해지는 세그먼트 갯수의 평균값을 백분율로 나타내었다.

- 노드의 수 : 11(모든 노드는 동일간격으로 분포)
- 망의 길이 : 5Km
- 망의 속도 : 150Mbps
- 노드의 BWB_MOD 값 : 8
- 패킷의 발생 : Poission 분포
- 패킷의 길이 : 8 세그먼트 /패킷
- 버퍼의 크기 : 20 세그먼트
- RS_MAX : 10

1. 낮은 부하시의 액세스 지연

30% 부하일때 우선순위가 없는 경우를 <그림 10>, 2단계의 우선순위의 경우를 <그림 11>에 나타내었다. <그림 10>은 1개의 소거노드를 가질 때 우선순위가 없는 경우의 액세스지연을 나타낸 것이다. 제안된 두 알고리즘은 우선순위에 상관없이 거의 동일한 액세스지연을 나타내고, 저부하에서는 IV장의 상태 (가)와 같이 최적값에 비하여 더 많은 request를 제거하므로 소거노드의 상위쪽 노드는 액세스지연이 낮으며 하위쪽 노드는 높은 경향을 보이거나 그 효과는 비교적 적음을 알 수 있다. <그림 11>은 높은 우선순위 세그먼트와 낮은 우선순위 세그먼트를 동일하게 발생시킬 때의 액세스 지연을 나타낸 것으로 높은 우선순위 세그먼트의 액세스지연은 알고리즘에 상관없이 거의 동일하며 낮은 우선순위 세그먼트도 액세스지연의 차이가 별다른 차이를 보이지 않았다. 따라서 제안된 알고리즘(Eras-P, Eras-F)은 낮은 부하에서도 기존의 알고리즘과 유사한 성능을 보임을 알 수 있다.

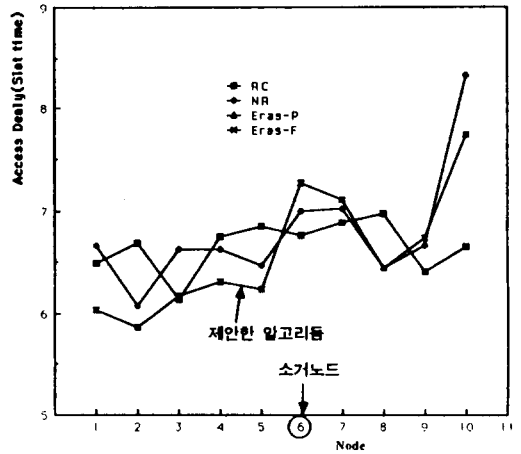


그림 10. 30% 부하, 우선순위가 없는 망에서의 액세스지연
Fig. 10. Access delay at 30% load, no priority.

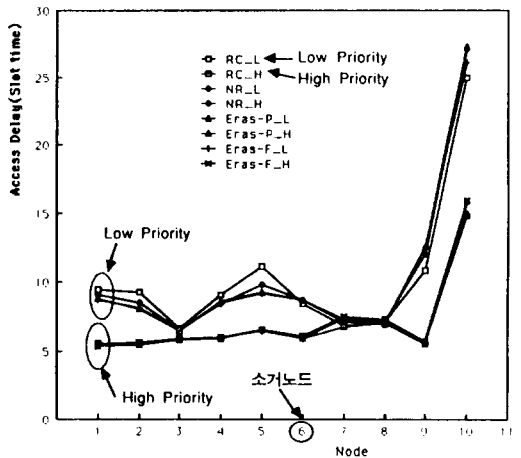


그림 11. 30% 부하, 2단계 우선순위 망에서의 액세스지연
Fig. 11. Access delay at 30% load, two priorities.

2. 높은 부하시의 액세스 지연

IV장의 상태 (나)와 같이 망에 높은 부하가 걸려 있을 경우 성능을 비교한 것으로 1개의 소거노드를 사용하고 110%의 부하가 가해졌을 경우 우선순위가 없을때의 액세스지연을 <그림 12>에, 2단계의 우선순위 망의 액세스지연을 <그림 13>에 나타내었다. 2개의 소거노드를 사용하고 110%의 부하가 가해졌을 경우의 액세스지연을 <그림 14>, <그림 15>에 나타내었다. 제안된 두 알고리즘은 거의 동일한 지연을 가지며, 소거노드의 위치에 따른 액세스지연의 변화가 기존 알고리즘에 비하여 적음을 알 수

있다. 이는 소거노드가 최적점에 가깝게 위치하고 있기 때문에 재생된 슬롯 갯수와 제거된 request 갯수가 평형을 이루기 때문이며, 기존 알고리즘에 비하여 보다 개선된 특성을 보이고 있다. 2단계의 우선순위를 가지는 경우에 있어서 높은 우선순위 세그먼트의 지연은 알고리즘에 상관없이 유사한 지연을 가지나 낮은 우선순위 세그먼트의 지연은 본 논문에서 제안한 알고리즘이 가장 우수한 특성을 보이며 negative request를 사용하는 방법(NR), request를 제거하는 방법(RC) 순으로 액세스지연이 커진다.

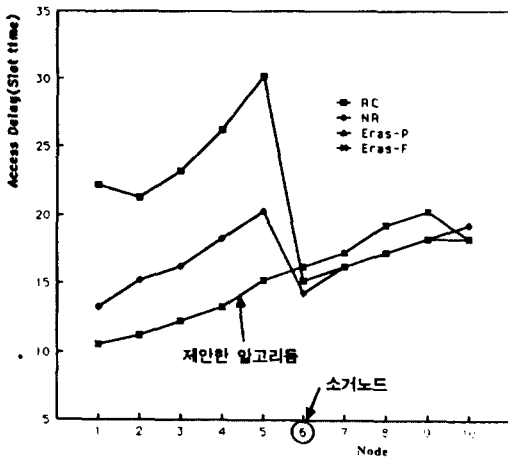


그림 12. 110% 부하, 우선순위가 없는 망에서의 액세스지연
Fig. 12. Access delay at 110% load, no priority.

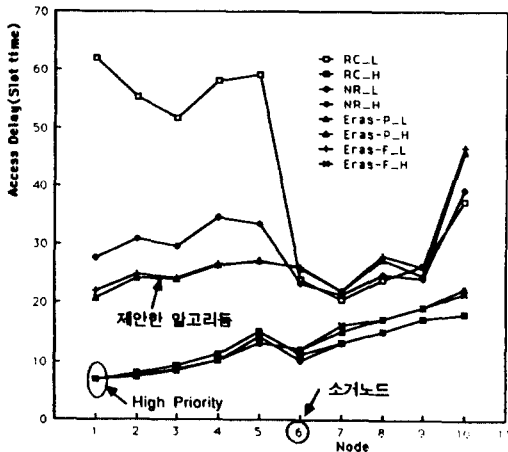


그림 13. 110% 부하, 2단계 우선순위 망에서의 액세스지연
Fig. 13. Access delay at 110% load, two priorities.

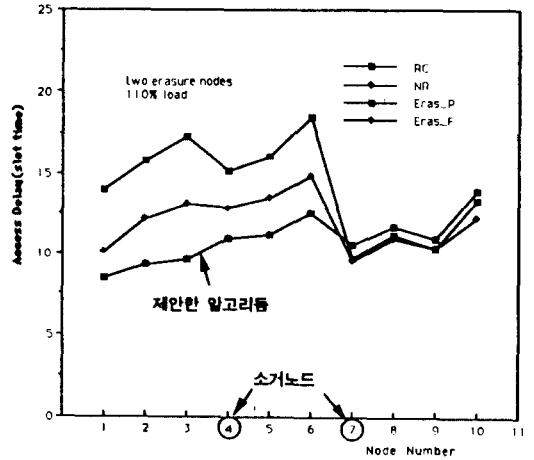


그림 14. 110% 부하 우선순위가 없는 망에서의 액세스지연
Fig. 14. Access delay at 110% load, no priority.

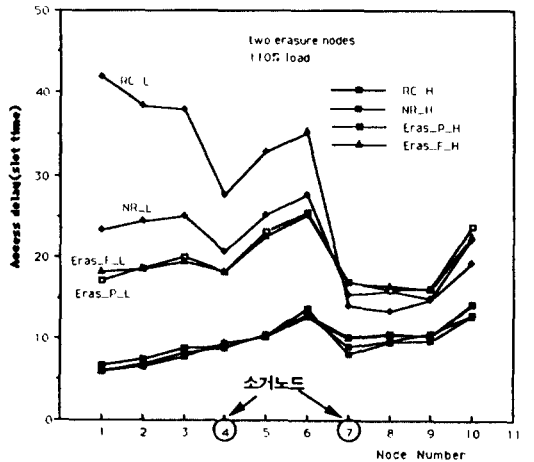


그림 15. 110% 부하, 2단계 우선순위 망에서의 액세스지연
Fig. 15. Access delay at 110% load, two priorities.

3. 최적위치가 아닐 경우의 액세스 지연

IV장의 상태 (다)와 같이 소거노드의 위치가 최적위치로부터 하위쪽으로 벗어나 있기 때문에 소거노드에서 발생시키는 빈 슬롯의 수가 하위쪽으로부터 수신하는 request의 수보다 많은 경우의 액세스지연을 <그림 16>, <그림 17>에 나타내었다.

$$\sum_{i=1}^k f_{ni} > \sum_{i=k}^{10} f_{ni}$$

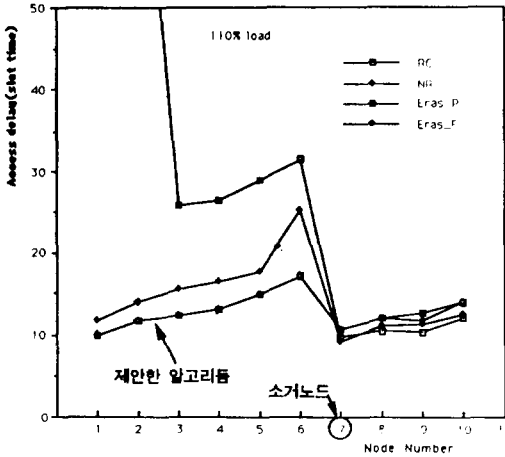


그림 16. 110% 부하, 우선순위가 없는 망에서의 액세스지연
Fig. 16. Access delay at 110% load, no priority.

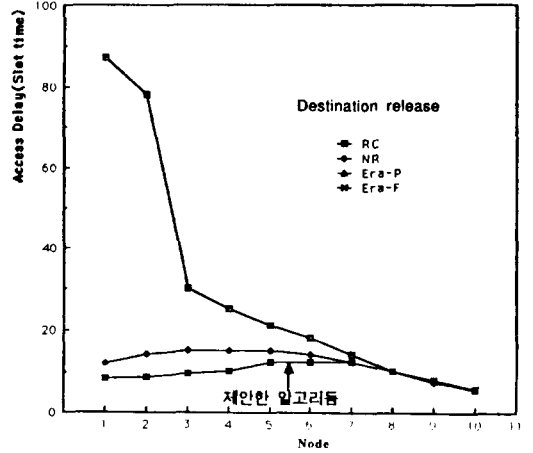


그림 18. 130% 부하, 우선순위가 없는 망에서의 액세스지연
Fig. 18. Access delay at 130% load, no priority.

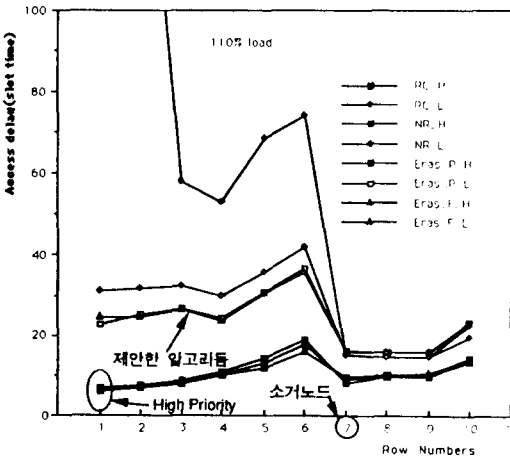


그림 17. 110% 부하, 2단계 우선순위 망에서의 액세스지연
Fig. 17. Access delay at 110% load, two priorities.

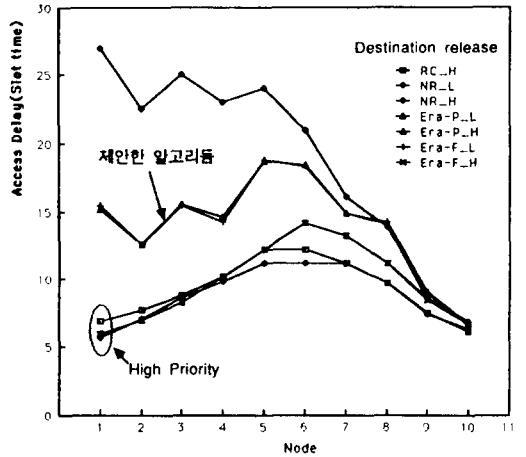


그림 19. 130% 부하, 2단계 우선순위 망에서의 액세스지연
Fig. 19. Access delay at 130% load, two priorities.

즉 <그림 16>, <그림 17>은 위의 조건을 만족하는 점에 소거노드를 위치시켰을 때의 액세스지연을 나타낸 것으로 이 경우에도 제한한 알고리즘이 가장 낮은 값을 보이고 있다. 소거노드가 최적점에 가깝게 위치한 <그림 12>, <그림 13>, <그림 14>, <그림 15>에서는 소거노드의 액세스지연이 이전 노드에 비하여 별다른 차이를 보이지 않고 있으나, 이 경우에는 재생되는 슬롯갯수가 제거되는 request 갯수에 비하여 많으므로 소거노드의 액세스지연이 이전 노드에 비하여 작은 특성을 보인다.

4. 목적지 소거방식으로 동작할 경우의 액세스지연
모든 노드가 소거노드로 동작할 경우, 즉 목적지 소거방식으로 동작할 경우의 액세스지연을 <그림 18>, <그림 19>에 나타내었다. 모든 노드가 소거노드로 동작하면 IV장의 상태 (가), (나), (다)의 모든 상태가 망에 동시에 존재하는 경우가 된다. 역시 본 논문에서 제한한 방식이 낮은 액세스지연을 가지며, request를 제거하는 방식의 낮은 우선순위(RC_L) 세그먼트의 액세스지연은 <그림 19>의 범위를 크게 벗어나 생략하였다.

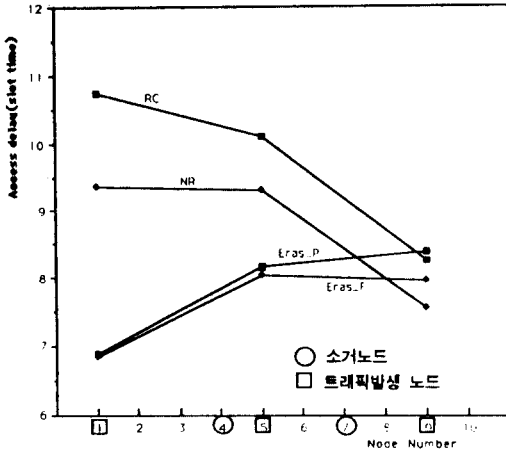


그림20. 비균일 부하시의 액세스지연(no priority)
Fig. 20. Access delay at nonuniform traffic (no priority).

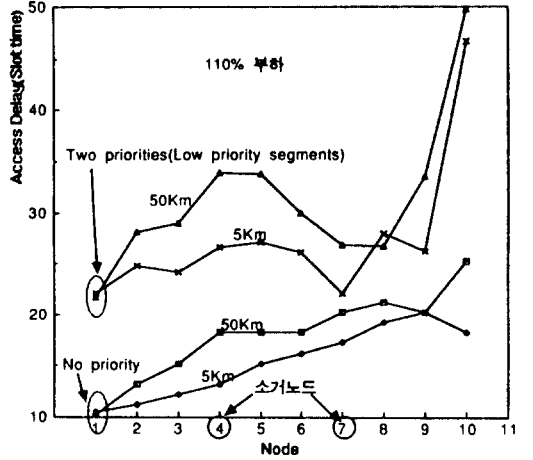


그림22. 망의 길이의 변화에 따른 액세스지연의 변화
Fig. 22. Variance of access delay with different bus length.

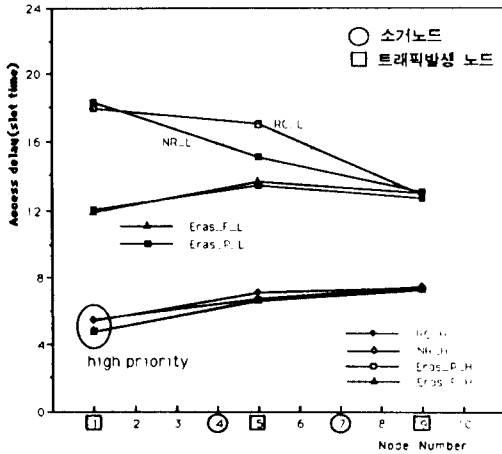


그림21. 비균일 부하시의 액세스지연(two priority)
Fig. 21. Access delay at nonuniform traffic(two priorities).

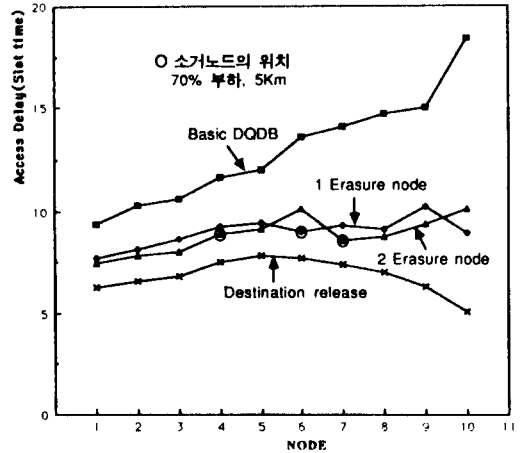


그림23. 소거노드의 수에 따른 액세스지연의 변화
Fig. 23. Variance of access delay with different numbers of erasure node.

5. 불균형 부하시의 노드별 액세스 지연

11개의 노드들중 3개의 노드, 즉 1번, 5번, 9번 노드만이 동일한 크기의 트래픽을 발생시키며 발생된 트래픽의 총량은 110%가 되도록 했을때의 트래픽 발생노드의 액세스지연 특성을 <그림 20>, <그림 21>에 나타내었다. 이때 4번, 7번 노드는 소거노드로 동작한다.

6. 망의 길이 및 소거노드 수의 변화에 따른 액세스지연의 변화

제안된 Eras-F 알고리즘의 망 길이에 따른 액세스지연의 변화를 <그림 22>에, 소거노드 수에 따른 액세스지연의 변화를 <그림 23>에 나타내었다.

<그림 22>에서는 2개의 소거노드를 가지는 망에 110%의 부하를 가하고 망의 길이가 5Km, 50Km일 때의 우선순위가 없는 경우 및 2단계 우선순위중 낮은 우선순위 세그먼트의 액세스지연을 나타내었다. 망의 길이가 길어질수록 액세스지연이 증가하는데, 이는 망이 길어질수록 노드간에 request 및 빈 슬롯의 전달시간이 길어져

완전한 분산큐로 동작하지 못하기 때문이다.

〈그림 23〉에서는 70%의 부하일때, 소기노드가 없는 경우, 1개인 경우, 2개인 경우, 모든 노드가 소기노드로 동작하는 경우의 액세스지연의 변화를 나타낸 것으로, 소기노드의 수가 증가할수록 액세스지연이 감소된다. 그러나 소기노드가 없는 경우에서 1개로 증가될 경우의 액세스지연은 크게 줄어든다, 소기노드의 수가 증가할수록 그 효과는 작아짐을 알 수 있다. 이는 II장에서 나타낸바와 같이 1개의 소기노드를 사용하였을 경우 이용률이 33%, 2개를 사용하였을 경우 52%, 모든 노드가 소기노드로 동작하면 100% 증가되므로 소기노드가 증가할수록 액세스지연의 감소효과가 작아진다.

VI. 결 론

본 논문에서 제안한 두 가지의 소기노드 알고리즘(Eras-P, Eras-F)은 동일한 특성을 나타내며, $RQ > 0$ 또는 $CD > 0$ 일 때에만 request를 제거하는 방법(RC)이나 negative request를 사용하는 방법(NR)에 비하여 낮은 액세스지연을 나타내며, 또한 모든 경우에서 NR이 RC에 비하여 낮은 액세스지연을 가짐을 확인하였다.

소기노드 알고리즘의 성능은 소기노드에 의하여 증가된 대역폭을 모든 노드에게 얼마나 공정하게 분산시켜 주느냐에 따라 결정된다. RC에서 소기노드의 RQ 계수기와 CD 계수기의 값이 모두 0일 경우에는 빈 슬롯을 재생했을 경우에도 request를 제기하지 않으므로 헤드에 가까운 노드는 더 많은 수의 request를 받게되어 빈 슬롯을 사용할 기회가 적어지므로 액세스 지연이 커진다. NR의 경우 소기노드에서 발생된 빈 슬롯과 negative request의 갯수가 동일하므로 소기노드에 의하여 증가된 대역폭을 모든 노드에 완전하게 분산시켜 주는 것처럼 보이나, negative request는 CD 계수기 또는 RQ 계수기의 값이 0보다 클 경우에만 계수기의 값을 감소시키므로 상위노드에게 소기노드의 효과를 제대로 분산시켜주지 못하여 상위쪽 노드의 액세스지연이 하위쪽 노드에 비하여 커진다. 따라서 RQ 및 CD의 상태와 상관없이 request를 제거하는 방식은 재생되는 슬롯의 갯수와 제기되는 request 갯수가 평형 상태를 이루므로 기존 알고리즘에 비하여 개선된 성능을 보이며, 특히 소기노드가 최적점에 위치한 경우 소기노드의 액세스지연은 이전 노드와 별다른 차이를 보이지 않는다. 또한 액세스지연 분포는 소기노드가 없는 DQDB와 같이 망의 끝으로 갈수록 증가되는 특성을 나타낸다.

본 논문에서 제안한 두 알고리즘은 우선순위 적용방식이 다름에도 불구하고 동일한 효과를 나타내고 있는데, 이는 우선순위에 따라 구분하여 request를 제거할 필요가

없음을 보여준다. 따라서 액세스제어 영역을 변화시키지 않으며 간단하게 구현될 수 있는 Eras-F가 유용한 소기노드 알고리즘이 될것으로 사려된다. 그러나 소기노드에서 발생하는 빈 슬롯의 수가 하위쪽 노드로부터 받는 request의 수보다 많을 경우, RS 계수기는 RS_MAX 값을 유지하게 되므로 부하가 크게 변화될 경우, 하위쪽 노드의 액세스지연이 순간적으로 크게 증가될 우려가 있다. 따라서 부하의 상태에 따라 RS_MAX의 값을 동적으로 결정하는 방법에 대한 연구가 이루어져야 할것이다.

參 考 文 獻

- [1] IEEE Project 802. 6, Proposed Standard, Distributed Queue Dual Bus Metropolitan Area NetWork, *Doc. no. P802.6/D12*, Feb. 1990.
- [2] H. Ohnishi, N. Morita & S. Suzuki, "ATM Ring Protocol and Performance," *ICC'89*, pp. 394-398, June 1989.
- [3] Specification of the Asynchronous Transfer Mode Ring(ATMR) Protocol, *Japaness National Body, Version A1.1*, July 1991.
- [4] I. Cidon & Y. Ofek, "Metaring-a full-duplex ring with fairness and spatial reuse," *INFOCOM'90*, pp. 969-981, 1990.
- [5] Y. Gong, M. Paterakkis, "Performance analysis of a flexible protocol achieving user fairness in high-speed dual-bus networks with destination release," *INFOCOM'91*, pp. 479-485, 1991.
- [6] M. W. Garrett, San-Qi Li, "A study of slot reuse in dual bus multiple access networks," *INFOCOM'90*, pp. 617-629, 1990.
- [7] M. A. Rodrigues, "Erasure node : performance improvements for the IEEE 802.6 MAN," *INFOCOM'90*, pp. 636-644, 1990.
- [8] A. E. Kamal, "Efficient multi-segment message transmission with slot reuse on DQDB", *INFOCOM'91*, pp. 869-878, 1991.
- [9] M. Zukerman, P. Potter, "A protocol for erasure node implementation within the DQDB framework," *GLOGECOM'90*, pp. 1400-1404, Dec. 1990.
- [10] T. Yokotain, H. Sato, S. Nakatsuka, "Proposed erasure node algorithm in DQDB," *IEEE 802.6 Contribution 802E 90/49*, 1990.

- [11] L. Susanna, D. Luciani, R. Pignatelli, "The balanced erasure : A mechanism for slot reuse in DQDB protocol," *ICC'91*, pp. 1350-1354, June 1991.
- [12] M. Zukerman, P. Potter, "The DQDB protocol and its performance under overload traffic conditions," *Computer Network and ISDN System*, vol. 20, 1990.
- [13] 김광택, 한기준, "DQDB에서의 분산 Erasure 노드 프로토콜", 한국통신학회 91 하계 학술발표회 논문지, pp. 67-70, 1991년 8월.
- [14] 전병천, 김대영, "DQDB망 소거노드 알고리즘들의 성능 비교", 한국통신학회 하계종합학술발표회 발표, 1991년 11월.

 著 者 紹 介



全 炳 千 (正會員)

1984年 충남대학교 전자공학과(학사). 1986年 충남대학교 전자공학과(석사). 1992年 8월 충남대학교 전자공학과(박사). 1987年-1988年 국방과학연구소, 1988年 현재 한국전자통신연구소 선임연구원.



金 大 榮 (正會員)

1975年 서울공대 전자공학과(학사). 1977年 KAIST 전기 전자공학과(석사). 1983年 KAIST 전기 전자공학과(박사). 1978年-1981年 서독 RWTH Aachen, UNI Hannover. 1987-1988年 미국 University of California Davis 분교. 1983년-1991년 충남대학교 전자공학과. 1992년-현재 충남대학교 정보통신공학과 부교수