

트래픽 편중화에 근거한 다중채널 TDM 프로토콜

正會員 白 善 旭* 正會員 崔 陽 熙* 正會員 金 宗 相*

Multi-Channel TDM Protocol based on Traffic Locality

Seonuck Paek*, Yanghee Choi*, Chongsang Kim* *Regular Members*

要 約

각 노드에게 타임슬롯을 정적으로 할당하는 TDM 프로토콜은 동작이 간단하고 고부하에서의 처리율이 높다는 잇점때문에 TDM 프로토콜을 다중채널 고속통신망에 적용하는 연구가 활발히 이루어지고 있다. 그러나, 다중채널 TDM 프로토콜에서 저부하에서의 지연시간이 필요이상 크고 트래픽의 편중화에 동적으로 적용하지 못하는 것은 해결되어야 할 문제점이다. 본 논문에서는 이러한 문제점을 완화할 수 있는 새로운 TDM 프레임 구조를 제안한다. 제안된 구조에서 상호 트래픽이 많은 노드들 사이에는 많은 양의 타임슬롯이 할당되며 그렇지 않은 노드들 사이에는 상대적으로 적은 양의 타임슬롯이 할당됨으로써 지연시간과 처리율이 개선된다. TDM 프레임 설계시 고려해야 할 사항을 처리율 최적화와 저부하에서의 지연시간 최적화 관점에서 분석하였다. 제안된 구조의 지연시간 분석을 위한 근사해석모델을 개발하였으며 그 타당성은 시뮬레이션에 의해 검증하였다.

ABSTRACT

Since TDM protocol can be easily implemented and show high throughput at heavy load, the researches on the multi-channel high-speed network based on TDM access control have been getting more attention than ever. TDM-type multi-channel network, however, has disadvantages of excessive delay at light load and inadaptibility to traffic skewing. In this paper, we propose a new multi-channel TDM structure as an approach to alleviate the previous problems. In the proposed TDM structure, time slots are allocated proportional to the traffic flow pattern among the nodes, thus delay and throughput performance are improved. Design principles of TDM frame are discussed considering traffic locality and the number of available channels. Approximate analytic models for delay evaluation are developed and verified by simulations.

I. 서 론

최근에 광통신 기술과 WDM(Wavelength Division

Multiplexing) 및 고속 소자 기술의 발달로 인해 하나의 물리적 회선에 여러 개의 논리적 채널을 둘 수 있게 되어, 다중 채널을 이용한 고속 통신망 연구가 활발히 이루어지고 있다[1][2]. 특히, WDM star coupler 형태의 광통신망 구조는 신호의 손실이 적고 신

* 서울대학교 컴퓨터공학과
Dept. of Computer Eng., Seoul National Univ.
論文番號: 94-30

되도가 높다는 특성 때문에 현재 연구가 집중되고 있다[3][4]. 현재 다중 채널 통신망에 대한 연구는 이용가능한 채널의 수, 각 노드가 가질 수 있는 송/수신 인터페이스의 수 및 조절가능성과 제어채널의 존재 여부에 따라 다양한 프로토콜이 제안되고 있다[1][2]. 일반적으로 채널을 정적으로 사용할수록 구현은 쉬운 반면 트래픽 흐름의 변화에 적응하기 곤란하므로 대역폭 낭비를 초래한다. 반면에 동적인 알고리즘일수록 구현은 어려운 반면 채널의 이용효율은 개선된다[1][2].

다중채널의 매체 접근 제어 프로토콜 중에서 각 노드에 채널과 타임슬롯을 정적으로 할당하는 TDM 방식은 제어가 쉽고 송/수신 인터페이스가 간단하면서 고부하에서 처리율이 높다는 장점때문에 많은 연구가 이루어지고 있다: SA(Source Allocation), DA(Destination Allocation), SDA(Source Destination Allocation), AMTRACE(A Multibus TRAIIn Communication), WTDM(Wavelength Time Division Multiplexing), I-TDMA(Interleaved Time Division Multiplexed Access)[5][6][7][8]. 예를 들어, SDA[5]는 가장 정적인 고정 할당 TDM 프로토콜로서 각 노드가 이용할 수 있는 채널과 타임슬롯이 각 목적지 노드에 대해 미리 결정되어 있다. 표 1에 4개의 노드가 2개의 채널을 이용하여 SDA프로토콜에 따라 통신할 때의 채널/타임슬롯 할당예가 나타나 있다.

표 1. SDA 채널/타임슬롯 할당표($N=4, b=2$)
Table 1. Channel/time slot allocation of SDA($N=4, b=2$)

		slot t					
		1	2	3	4	5	6
node i	1	2		3		4	
	2	3		4		1	
	3		4		1		2
	4		1		2		3

표 1의 각 행은 송신 노드 번호를 나타내고 각 열은 타임슬롯 번호를 나타낸다. 그리고 표 1의 각 항목은 수신 노드의 번호를 나타낸다. 같은 타임슬롯에 패킷이 동시에 전송되는 '채널충돌'을 방지하기 위해서 각 열에는 0이 아닌 항목의 수가 이용가능한 채널의 수보다 더 크지 않도록 한 것이다. 또한, 한 목적지에 수신인터페이스의 수 이상으로 패킷이 도착하는 '과잉도착' 현상을 방지하기 위해서 각 열에는 같은 목적

지 번호가 두번 이상 나타나지 않도록 하였다. SDA와 같은 고정할당 TDM 방식에서는 일반적으로 TDM 프레임을 구성하고, TDM 프레임내에서 자신에게 할당된 타임슬롯만을 이용할 수 있게 된다. N 개의 노드가 b 개의 채널로 SDA 방식으로 통신한다면, 채널충돌과 과잉도착 현상이 없도록 각 사용자노드에 타임슬롯을 할당해야 한다는 제약 조건으로부터 필요한 TDM 싸이클 주기의 최소값은 $\lceil N(N-1)/b \rceil$ 가 된다. SDA와 같은 고정할당 TDM 계열의 프로토콜들은 빈 채널 선택이나 채널충돌 방지, 과잉도착 방지 등과 같은 부담을 제거할 수 있으며, 이러한 제어와 관련된 대역폭 낭비가 없어 채널 수의 증가에 따른 망의 대역폭 증가를 기대할 수 있다[5][6][7][8].

그러나, 고정 할당 방식의 다중채널 TDM 프로토콜들은 일반적으로 다음과 같은 문제점을 지닌다. 첫째, 부하가 없을 때 도착한 패킷이라도 지정된 타임슬롯까지 기다려야 하므로 저부하에서의 지연시간이 필요이상 크다. 고부하에서는 대기지연(queueing delay)이 지연시간의 주요 요인이지만, 저부하에서는 대기지연보다는 TDM 싸이클의 주기가 지연시간의 주요한 원인이므로 저부하에서의 지연시간 성능은 TDM 싸이클 주기가 커짐에 따라 더 나빠진다. 둘째, 통신망 구성에 참여하는 노드의 수가 점점 많아짐에 따라 TDM 싸이클의 주기가 점점 커진다. 특히, 이용가능한 채널의 수가 한정되어 있을 때 싸이클 주기는 SDM의 경우 $O(N^2)$ 로 증가한다. 따라서 임의의 두 노드 사이에서 사용될 수 있는 대역폭은 상대적으로 $1/O(N^2)$ 로 적어져서 양단간의 성능은 그만큼 저하된다[9]. 정적 할당 알고리즘의 이러한 성능저하 현상을 완화시키기 위한 시도로서 동적 알고리즘인 SWIFT(Store and forward With Integrated Frequency and Time switching)가 제안되었지만[9], 그 알고리즘은 다른 노드의 상태가 한 타임슬롯이내에 알려질 수 있는 환경에서만 적용할 수 있으므로, 노드 사이의 전파지연시간(propagation delay)이 전송 지연시간(transmission delay) 보다 큰 고속 LAN/MAN 환경에서는 효과를 기대하기 어렵다. 셋째, 대규모 통신망의 트래픽 유형은 일반적으로 특정 노드들 사이에서만 통신량이 집중되는 트래픽의 편중화 현상(traffic localization)이 현저하다[10][11]. 따라서 TDM 싸이클을 구성하는 타임슬롯 중에서 일부는 빈번하게 사용되어 병목현상을 발생시킬 소지가 있는 반면, 일부는 거의 사용되지 않아서 낭비되는 현상이 나타난다[9].

본 논문에서는 비교적 대규모의 교환기나 멀티프로세서 시스템의 상호연결망 및 LAN/MAN 환경에 적용할 수 있는 새로운 다중채널 TDM 프로토콜을 제안하고 TDM 구조 설계시 고려해야할 원칙들에 대해 고찰한다. 제안된 TDM 프로토콜의 목적은 트래픽 편중화 현상이 강한 노드들 사이의 통신에 좀 더 많은 전송 기회를 주어 처리율과 지연시간을 개선시키며 상호통신량이 적은 노드들 간에는 전송기회를 상대적으로 적게 주어 대역폭의 낭비를 줄이는 것이다. 또한, TDM 싸이클 주기를 가능한 한 단축하여 저부하에서의 과도한 지연시간을 개선하고자 하는 것이다.

제안된 프로토콜에서 그룹간의 트래픽 편중화 현상이 심한 노드들은 하나의 군(group)으로 묶고, 이에 따라 통신망은 몇 개의 군으로 구성된다. 이와 같이 트래픽 편중현상을 TDM 프레임 구조에 반영함으로써 TDM 방식의 일반적인 단점인 저부하에서의 과도한 지연시간과 트래픽 흐름에 능적으로 적응하지 못하는 문제를 완화할 수 있음을 본 논문에서 확인하고자 한다.

II 절에서는 제안된 TDM 프레임 구조 및 채널/타임슬롯 할당 알고리즘에 대해 기술하고 III 절에서는 제안된 알고리즘의 성능 평가를 위한 해석적 모델을 기술하며 IV 절에서는 제안된 방식과 기존의 방식과의 성능 비교 결과를 제시한다.

II. 프로토콜 모델

2.1 통신망 구성

N 개의 노드가 동일한 대역폭을 갖는 $b(1 \leq b \leq N)$ 개의 채널을 통해 통신하는 다중채널 통신망을 생각하자. 각 채널은 동일한 크기의 타임슬롯으로 나누어지고 데이터는 타임슬롯 크기의 패킷 단위로 전송된다고 가정하자. 또한, 각 사용자노드는 하나의 송신 인터페이스와 하나의 수신인터페이스를 갖고 있어서 b 개의 채널 중에서 하나의 채널을 선택하여 송신하고 마찬가지로 하나의 채널로 수신할 수 있다. 비교적 큰 규모의 통신망에서 특정 노드들끼리만 서로 빈번하게 통신하는 트래픽 편중화 현상은 널리 알려진 사실이므로, 본 논문에서는 노드들간의 트래픽 편중화 현상에 따라 N 개의 노드를 몇개의 군으로 분할하도록 제안한다[10][11]. N 개의 노드가 트래픽 편중화 현상에 따라 한 군마다 l_i 개씩 g 개의 군으로 분할된다고 가정한다($i=1, \dots, g, \sum_{i=1}^g l_i = N$). 군의 크기

가 1일 때는 따로 분할할 필요가 없으므로 $2 \leq l_i \leq N-2(i=1, 2, \dots, g)$ 인 경우만 생각하고 또한, $g=1$ 이면 하나의 군으로만 이루어진 통신망이므로 $2 \leq g \leq \lceil N/2 \rceil$ 일 때만 생각한다. 서로 다른 군 사이의 통신을 위해 각 군마다 하나씩 모두 g 개의 게이트웨이를 고려한다. 게이트웨이는 하나의 송신인터페이스와 수신인터페이스를 갖도록 한다.

사용자노드와 게이트웨이의 채널/타임슬롯 접근 알고리즘으로서의 정적 할당 TDM 방식 중에서 가장 고정적인 SDA를 채택한다[5]. 물론, 다른 정적 할당 TDM 알고리즘에도 제안된 방안이 적용될 수 있지만 구현과 성능평가의 관점에서 가장 간단한 SDA를 채택한 것이다. 각 송/수신 인터페이스가 각 타임슬롯에 선택해야 할 채널이 SDA 프로토콜에 따라 미리 결정되어 있으므로 고도의 정밀도를 요하는 하드웨어를 사용하지 않고 현재의 FDM 기술이나 WDM 기술을 이용하여 쉽게 구현할 수 있다[6]. 또한, 게이트웨이와 각 사용자노드들의 채널 접근 알고리즘이 동일하므로 게이트웨이를 별도로 두지 않고 각 군내의 한 노드가 게이트웨이의 기능을 담당하도록 구현할 수도 있다.

채널이 20 개일 때, 20 개의 노드가 4 개씩 5 개의 군으로 분할될 경우에 본 논문에서 제안한 분할에 의한 방식의 재구성 방안을 적용한 예를 표 2에 나타내었다.

표 1에서처럼 각 행은 송신 사용자노드나 송신 게이트웨이를 의미하며, 각 열은 타임슬롯을 나타낸다. 표 2의 각 항목은 목적지를 의미한다. 채널충돌이나 과잉도차 현상을 방지하기 위해서 한 열에 0이 아닌 항목의 수가 b 보다 많지 않도록 하고, 또한 한 열에 같은 목적지가 두번 이상 나타나지 않도록 하였다. 표 2에서 사용자노드 $n_{1,1}$ 은 같은 군에 속한 노드 $n_{1,2}, n_{1,3}, n_{1,4}$ 로 4 타임슬롯마다 한 번씩 내부패킷을 전송할 수 있다. 반면에 노드 $n_{1,1}$ 로부터 다른 군에 속한 노드 $n_{5,4}$ 로의 패킷전송은 타임슬롯 1에 게이트웨이 G_1 으로 전송되고 타임슬롯 8에 게이트웨이 G_5 로 전송되며 마지막으로 타임슬롯 12에 $n_{5,4}$ 로 전달됨으로써 이루어진다. 부하가 적을 때에는 12 타임슬롯마다 한 번씩 다른 군에 속한 노드로 패킷을 전송할 수 있다. 한편, 망을 분할하지 않고 모든 노드 사이에 SDA 방식으로만 타임슬롯을 할당하면 TDM 싸이클 주기는 19가 되어 19 타임슬롯마다 한번씩 특정 노드로 전송할 수 있다.

본 논문에서는 제안된 분할방식에 따라 구성된 통

표 2. H-SDA 채널/타임슬롯 할당표($l = 4, g = 5, b = 20$)

Table 2. Channel/time slot allocation of H-SDA($l = 4, g = 5, b = 20$)

		slot l											
		1	2	3	4	5	6	7	8	9	10	11	12
node i / gateway G_i	$n_{1,1}$	G_1	$n_{1,2}$	$n_{1,3}$	$n_{1,4}$		$n_{1,2}$	$n_{1,3}$	$n_{1,4}$		$n_{1,2}$	$n_{1,3}$	$n_{1,4}$
	$n_{1,2}$	$n_{1,3}$	G_1	$n_{1,4}$	$n_{1,1}$	$n_{1,3}$		$n_{1,4}$	$n_{1,1}$	$n_{1,3}$		$n_{1,4}$	$n_{1,1}$
	$n_{1,3}$	$n_{1,4}$	$n_{1,1}$	G_1	$n_{1,2}$	$n_{1,4}$	$n_{1,1}$		$n_{1,2}$	$n_{1,4}$	$n_{1,1}$		$n_{1,2}$
	$n_{1,4}$	$n_{1,2}$	$n_{1,3}$	$n_{1,1}$	G_1	$n_{1,2}$	$n_{1,3}$	$n_{1,1}$		$n_{1,2}$	$n_{1,3}$	$n_{1,1}$	
	$n_{2,1}$	G_2	$n_{2,2}$	$n_{2,3}$	$n_{2,4}$		$n_{2,2}$	$n_{2,3}$	$n_{2,4}$		$n_{2,2}$	$n_{2,3}$	$n_{2,4}$
	$n_{2,2}$	$n_{2,3}$	G_2	$n_{2,4}$	$n_{2,1}$	$n_{2,3}$		$n_{2,4}$	$n_{2,1}$	$n_{2,3}$		$n_{2,4}$	$n_{2,1}$
	$n_{2,3}$	$n_{2,4}$	$n_{2,1}$	G_2	$n_{2,2}$	$n_{2,4}$	$n_{2,1}$		$n_{2,2}$	$n_{2,4}$	$n_{2,1}$		$n_{2,2}$
	$n_{2,4}$	$n_{2,2}$	$n_{2,3}$	$n_{2,1}$	G_2	$n_{2,2}$	$n_{2,3}$	$n_{2,1}$		$n_{2,2}$	$n_{2,3}$	$n_{2,1}$	
	$n_{3,1}$	G_3	$n_{3,2}$	$n_{3,3}$	$n_{3,4}$		$n_{3,2}$	$n_{3,3}$	$n_{3,4}$		$n_{3,2}$	$n_{3,3}$	$n_{3,4}$
	$n_{3,2}$	$n_{3,3}$	G_3	$n_{3,4}$	$n_{3,1}$	$n_{3,3}$		$n_{3,4}$	$n_{3,1}$	$n_{3,3}$		$n_{3,4}$	$n_{3,1}$
	$n_{3,3}$	$n_{3,4}$	$n_{3,1}$	G_3	$n_{3,2}$	$n_{3,4}$	$n_{3,1}$		$n_{3,2}$	$n_{3,4}$	$n_{3,1}$		$n_{3,2}$
	$n_{3,4}$	$n_{3,2}$	$n_{3,3}$	$n_{3,1}$	G_3	$n_{3,2}$	$n_{3,3}$	$n_{3,1}$		$n_{3,2}$	$n_{3,3}$	$n_{3,1}$	
	$n_{4,1}$	G_4	$n_{4,2}$	$n_{4,3}$	$n_{4,4}$		$n_{4,2}$	$n_{4,3}$	$n_{4,4}$		$n_{4,2}$	$n_{4,3}$	$n_{4,4}$
	$n_{4,2}$	$n_{4,3}$	G_4	$n_{4,4}$	$n_{4,1}$	$n_{4,3}$		$n_{4,4}$	$n_{4,1}$	$n_{4,3}$		$n_{4,4}$	$n_{4,1}$
	$n_{4,3}$	$n_{4,4}$	$n_{4,1}$	G_4	$n_{4,2}$	$n_{4,4}$	$n_{4,1}$		$n_{4,2}$	$n_{4,4}$	$n_{4,1}$		$n_{4,2}$
	$n_{4,4}$	$n_{4,2}$	$n_{4,3}$	$n_{4,1}$	G_4	$n_{4,2}$	$n_{4,3}$	$n_{4,1}$		$n_{4,2}$	$n_{4,3}$	$n_{4,1}$	
	$n_{5,1}$	G_5	$n_{5,2}$	$n_{5,3}$	$n_{5,4}$		$n_{5,2}$	$n_{5,3}$	$n_{5,4}$		$n_{5,2}$	$n_{5,3}$	$n_{5,4}$
	$n_{5,2}$	$n_{5,3}$	G_5	$n_{5,4}$	$n_{5,1}$	$n_{5,3}$		$n_{5,4}$	$n_{5,1}$	$n_{5,3}$		$n_{5,4}$	$n_{5,1}$
	$n_{5,3}$	$n_{5,4}$	$n_{5,1}$	G_5	$n_{5,2}$	$n_{5,4}$	$n_{5,1}$		$n_{5,2}$	$n_{5,4}$	$n_{5,1}$		$n_{5,2}$
	$n_{5,4}$	$n_{5,2}$	$n_{5,3}$	$n_{5,1}$	G_5	$n_{5,2}$	$n_{5,3}$	$n_{5,1}$		$n_{5,2}$	$n_{5,3}$	$n_{5,1}$	
G_1					G_2	G_3	G_4	G_5	$n_{1,1}$	$n_{1,2}$	$n_{1,3}$	$n_{1,4}$	
G_2					G_3	G_4	G_5	G_1	$n_{2,1}$	$n_{2,2}$	$n_{2,3}$	$n_{2,4}$	
G_3					G_4	G_5	G_1	G_2	$n_{3,1}$	$n_{3,2}$	$n_{3,3}$	$n_{3,4}$	
G_4					G_5	G_1	G_2	G_3	$n_{4,1}$	$n_{4,2}$	$n_{4,3}$	$n_{4,4}$	
G_5					G_1	G_2	G_3	G_4	$n_{5,1}$	$n_{5,2}$	$n_{5,3}$	$n_{5,4}$	

신망을 H-SDA(Hierarchical SDA)라 하고, 모든 사용자 노드들이 게이트웨이 없이 SDA 알고리즘에 따라 통신하는 방식을 F-SDA(Flat SDA)라 부르기로 한다. 또한, 어떤 사용자노드에 도착한 패킷의 목적지가 그 사용자노드와 같은 군에 속하면 패킷을 '내부 패킷'이라 하고 그렇지 않으면 '외부패킷'이라 한다. 표 2에서 내부패킷의 전송주기는 4인 반면 외부패킷의 TDM 주기는 12이다. 반면에 F-SDA의 TDM 사이클주기의 최소값은 19이다. 즉 트래픽 편중현상을 반영한 통신망 분할에 의해 TDM 사이클 주기를 단축할 수 있어 저부하에서의 지연시간은 짧아진다. 반면에 H-SDA에서 게이트웨이로의 송/수신 및 게이트웨이 상호간의 통신을 위해 타임슬롯을 할당해야 하므로 통신망의 최대처리율은 저하된다. 즉, 저부하에서의 지연시간과 고부하에서의 처리율은 길항관계(trade-off)에 있다.

F-SDA에서는 N 개의 노드가 $N-1$ 개의 노드에 대해 한 번씩 전송기회를 가지므로 F-SDA TDM 프레임 구성하는데 필요한 타임슬롯 수의 최소값을 B_f 라 하면 $B_f = N(N-1)$ 이 되고 이것을 b 개의 채널 위에 할당했을 때의 TDM 사이클 주기를 T_f 라 하면 $T_f = \lceil N(N-1)/b \rceil$ 이 된다. H-SDA의 TDM 사이클은 다음과 같은 부분으로 구성된다.

- B_i : 군 i 에 속한 사용자노드 상호 간의 통신을 위해 필요한 슬롯 수.
- B_{i_c} : g 개의 게이트웨이 상호간의 통신을 위해 필요한 슬롯 수.
- B_{iG} : 사용자노드들로부터 게이트웨이 방향으로 패킷을 전송하기 위해 필요한 슬롯 수.
- B_{iI} : 게이트웨이들로부터 사용자노드 방향으로 패킷을 전송하기 위해 필요한 슬롯 수.

2.2 H-SDA 타임슬롯 계산

이 때, 위의 각 부분은 다음과 같이 계산된다.

$$B_i = l_i(l_i - 1), i = 1, \dots, g$$

$$B_g = g(g - 1)$$

$$B_{iG} = B_{iL} = \sum_{i=1}^g l_i = N.$$

그러므로 H-SDA의 TDM 프레임의 구성하는데 필요한 전체 타임슬롯 수의 최소값을 B_h 라 하면, B_h 는 채널의 수 b 가 1 일 때의 TDM 싸이클 주기가 되는데, 다음과 같이 계산된다:

$$\begin{aligned} B_h &= \sum_{i=1}^g B_i + B_g + B_{iG} + B_{iL} = \sum_{i=1}^g l_i(l_i - 1) + g(g - 1) + 2N \\ &= \sum_{i=1}^g l_i^2 + g(g - 1) + N. \end{aligned}$$

B_h 와 B_f 의 비를 $R_{h/f}$ 라 정의하면 $R_{h/f}$ 는 1보다 작음을 알 수 있다:

$$R_{h/f} \equiv \frac{B_h}{B_f} = \frac{\sum_{i=1}^g l_i^2 + g(g - 1) + N}{N(N - 1)} < 1$$

$$(3 \leq g \leq \lfloor N/2 \rfloor \text{ or } g = 2, 4 < N).$$

즉, 본 논문의 제안과 같이 노드들을 군으로 분할하여 계층적으로 망을 재구성함으로써 TDM 프레임의 구성하는데 요구되는 타임슬롯 수를 줄일 수 있다. 군의 수 g 가 주어질 경우에 $R_{h/f}$ 가 최소값을 갖는 것은 각 군의 크기가 같을 때, 즉 $l_i \approx \lfloor N/g \rfloor (i = 1, 2, \dots, g)$ 일 때이다. 이렇게 각 군의 크기가 비슷한 규모로 되도록 하는 것은 크기가 다른 군보다 상대적으로 작은 몇 개의 군을 서로 통합함으로써 가능하다. 또한, 교환기나 멀티프로세서 시스템처럼 구성이 자유로운 환경이라면, 군의 크기가 같도록 자유롭게 구성할 수 있다. 따라서, 본 논문에서는 각 군의 크기가 같은 경우만 고려한다. 각 군이 l 개씩 같은 수로 구성된 경우를 가정하면, $R_{h/f}$ 는 다음과 같다:

$$R_{h/f} = \frac{l(l + 1)g + g(g - 1)}{lg(lg - 1)} < 1. \quad (1)$$

만일 l 이 g 보다 크다면 대략 g 배 정도 H-SDA가 F-SDA 보다 타임슬롯이 적게 든다. 또한, $R_{h/f}$ 는 $g \approx \sqrt[3]{N^2/2}$ 일 때 최소값을 갖는다:

$$\begin{aligned} R_{h/f} &\approx \frac{3 + \epsilon}{\sqrt[3]{4(\sqrt[3]{N^2} - \epsilon)}} = \frac{1}{O(\sqrt[3]{N^2})} \\ (0 < \epsilon < 1, \quad 0 < \epsilon < 1). \end{aligned}$$

그러나 TDM 싸이클의 주기는 실제로는 이용가능한 채널 수 $b(1 \leq b \leq N)$ 에 의해 결정되는데, 표 2의 예에서 알 수 있듯이 $R_{h/f}$ 와 TDM 싸이클 주기의 비는 조금 다를 수 있다.

다음에는 H-SDA의 처리율을 분석한다. F-SDA 방식에서 각 노드는 $N - 1$ 개의 노드로 직접 1홉 만에 패킷을 전송할 수 있으므로 타임슬롯의 이용효율은 $N(N - 1)/B_f = 1$ 이 된다. 반면에, H-SDA에서는 F-SDA와 달리 외부패킷은 3홉을 가지므로 외부패킷 총 개에 쓰이는 타임슬롯 중 일부는 낭비되는 것으로 간주된다. 따라서 처리효율의 관점에서 보아 채널의 이용효율은 저하된다. H-SDA에서 각 노드는 한 번의 TDM 싸이클에 자신과 같은 군에 속한 $l - 1$ 개의 노드에 하나의 내부패킷을 전송할 수 있다. 한편, 외부패킷의 전송을 보면, 먼저 각 게이트웨이에는 자신이 담당하는 l 개의 사용자노드로부터 하나씩 모두 l 개의 패킷이 도착하고 이것을 $g - 1$ 개의 다른 게이트웨이로 전송한다. 또한, $g - 1$ 개의 다른 게이트웨이로부터 오는 패킷을 자신이 담당하는 l 개의 사용자노드로 전송한다. 그러므로 한 번의 TDM 싸이클에 하나의 게이트웨이를 통해 전송될 수 있는 외부패킷의 총 수는 $\min\{l, g - 1\}$ 이 되고 하나의 노드가 보낼 수 있는 외부패킷의 수를 $\min\{1, (g - 1)/l\}$ 이 된다. 즉, $l < g - 1$ 일 때는 각 사용자노드는 하나의 외부패킷을 한 TDM 싸이클에 전송할 수 있는데, 이 경우에는 한 게이트웨이에서 다른 게이트웨이로 보낼 수 있는 $g - 1$ 개의 타임슬롯 중에서 l 개만이 사용되므로 $(g - 1) - l$ 개의 슬롯은 구조적으로 낭비된다. 반면에 $l > g - 1$ 일 때는 사용자노드들에서 게이트웨이로 가는 l 개의 슬롯중에서 $g - 1$ 개 밖에 쓰일 수 없으므로 $l - (g - 1)$ 개의 패킷은 구조적으로 낭비된다고 볼 수 있다. 한편, $g \approx l + 1 \approx \sqrt{N}$ 일 때는 전송한 바와 같은 구조적 낭비는 없다. H-SDA의 타임슬롯 이용효율 Eff 는 다음과 같다:

$$\begin{aligned} Eff &\equiv \frac{N \times ((l - 1) + \min\{1, (g - 1)/l\})}{B_h} \\ &= \frac{N^2 - (1 - \min\{1, g(g - 1)/N\})gN}{N^2 + gN - g^2 + g^3}. \quad (2) \end{aligned}$$

즉, 최대처리효율 관점에서 군의 수 g 가 작을수록 타임슬롯의 효율은 증가하며 군의 수가 1 일 때는 F-SDA의 경우로서 최대처리효율은 1이 된다. 만일, g 가 작고 l 이 충분히 크면, 식 (2)에서 $Eff \approx 1$ 이므로 낭비는 거의 없다고 볼 수 있다.

지금까지 살펴본 바와 같이 TDM 사이클 주기와 처리율사이에는 서로 길항관계(tradeoff)가 있음을 알 수 있다. 즉, 식 (1)과 (2)로부터, 균의 수가 $g < \sqrt[3]{N^2/2}$ 의 범위에서 커질 수록 TDM 사이클의 주기는 단축되어 저부하에서의 지연시간 특성이 향상되는 반면, 망의 최대처리율(capacity)은 감소한다. 반면에 균의 수가 적어지면 처리용량은 점점 커지지만 저부하에서의 지연시간은 점점 길어진다. 즉, H-SDA 방식으로 망의 설계할 때 두개의 최적화 기준이 있음을 알 수 있다.

한편, 타임슬롯의 구조적 낭비가 없는 경우, 즉 $g \approx l+1 \approx \sqrt{N}$ 일 때는 $R_{eff} = 1/O(\sqrt{N})$ 가 되어 TDM 주기가 \sqrt{N} 배 정도 단축되고, $Eff = (g-1)^2/(g^2-1)$ 가 되어 g 가 클 경우 이용효율은 1에 가까워진다. 즉, 구조적 낭비가 없으면서 TDM 프레임 주기를 가장 많이 단축할 수 있으므로 멀티프로세서 시스템이나 교환기와 같이 통신망 분할을 임의로 할 수 있는 경우에는 최적의 구성이라고 판단된다.

2.3 H-SDA TDM 프레임 구성

H-SDA에서 요구되는 타임슬롯 수 B_k 를 b 개의 채널위에 할당하여 H-SDA 채널/타임슬롯 할당표(TDM 프레임)를 구성함에 있어 먼저 고려할 점은 SDA의 제약조건에 따라 '채널충돌'과 '과잉도착'이 없도록 하는 것이다. 또 하나의 고려사항은 저부하에서 외부패킷을 최초 송신 사용자노드에서 최종 목적지 사용자노드까지 가능한 빨리 전송할 수 있도록 하는 것이다. 이를 위해 각 균의 모든 사용자노드로부터 게이트웨이까지의 전송이 끝난 직후에 바로 송신 게이트웨이에서 목적지 게이트웨이로의 전송이 이루어지도록 H-SDA TDM 프레임에 타임슬롯을 할당한다. 마찬가지로 모든 송신 게이트웨이에서 목적지 게이트웨이로의 전송이 끝난 직후에 바로 목적지 게이트웨이에서 최종 목적지 노드로 전송이 이루어지도록 타임슬롯을 배치한다. 즉, B_{iL}, B_{iC}, B_{iR} 의 순서로 b 개의 채널위에 타임슬롯을 할당하되 그들 간에 간격이 없도록 한다. 이렇게 함으로써, 외부패킷이라도 부하가 적을 경우에 한 번의 H-SDA TDM 사이클 내에 최종목적지까지 전달이 될 수 있도록 하였다. 그리고 균 i 의 내부통신을 위한 타임슬롯 B_i 들은 외부패킷 전송을 위한 타임슬롯을 먼저 할당한 후에 TDM 프레임의 처음부터 살펴보아 아직 할당되지 않은 타임슬롯에 할당한다. 만일 채널 수가 적다면 B_{iL} 의 앞이나 B_{iR} 의 뒤에 배치하며, 만일 채널수가 충분하다면

B_{iL}, B_{iC}, B_{iR} 과 중복되는 시간대에 빈 채널에 할당될 수 있다.

표 2는 전송한 사항들을 고려한 채널/타임슬롯 할당 알고리즘으로부터 얻은 것이다. 표 2에서는 B_{iL}, B_{iC}, B_{iR} 와 중복되는 시간대에 B_i 들이 12타임슬롯안에 3번씩 같은 구조로 할당되었다. 만일 이용가능한 채널 수가 적다면 사이클 주기는 12보다 길어지고, 또한 B_{iL}, B_{iC}, B_{iR} 과 B_i 들은 서로 다른 시간대에 할당되게 된다. 그리고, 저부하에서 외부패킷의 지연시간을 최적화하기 위해서 모든 사용자노드들이 자신의 게이트웨이로 패킷을 전송한 직후에 게이트웨이들간의 통신이 이루어지도록 하였다. 즉, G_1 로부터 다른 게이트웨이로의 전송은 균 1의 모든 노드가 G_1 으로 전송한 직후에 이루어지도록 하였다. 마찬가지로 모든 게이트웨이들 간의 통신이 끝난후에 게이트웨이로부터 목적지 사용자노드로의 전송이 이루어지도록 구성하였다. 이 예에서 보는 것처럼 H-SDA와 F-SDA의 실제 TDM 사이클 주기의 비는 R_{eff} 와는 달리 SDA의 제약 조건과 채널 수로부터 결정된다. H-SDA에서 같은 균에 속한 사용자노드에 대해서는 4 타임슬롯마다 한 번씩 전송기회를 갖는 반면에 다른 균에 속한 사용자 노드들에 대해서는 12 타임슬롯마다 한 번씩의 전송기회를 갖는다. F-SDA에서는 주기가 19이므로 모든 사용자 노드에 대해 19 타임슬롯마다 한 번씩 전송기회를 갖는다. H-SDA를 적용할 경우, 균 내부통신 관점에서 본 사이클 주기를 T_k^L 라 하면, 표 2의 예에서 각 노드는 자신과 같은 균의 같은 노드에 대해 4 슬롯마다 한 번씩 전송할 수 있으므로 $T_k^L = 4$ 이다.

H-SDA에서의 TDM 사이클 구성을 위해 필요한 타임슬롯 수와 실제 TDM 사이클 주기와와의 관계를 몇 가지 통신망 구성에 대해 계산한 것이 표 3에 나타나 있다.

표 3. TDM 사이클 길이
Table 3. Length of TDM cycle

(l, g, b)	(T_f, T_k, T_k^L)
(2, 10, 1)	(380, 150, 150)
(4, 5, 1)	(380, 120, 120)
(10, 2, 1)	(380, 222, 222)
(2, 10, 2)	(190, 75, 75)
(4, 5, 2)	(190, 60, 60)
(10, 2, 2)	(190, 111, 111)
(4, 5, 1)	(380, 120, 120)
(4, 5, 2)	(190, 60, 60)
(4, 5, 20)	(19, 12, 4)

b 의 값이 작을 때는 B_b 와 B_f 의 비가 TDM 싸이클의 비와 일치하지만, b 의 값이 크면 SDA의 제약조건으로 인해서 서로 일치하지 않을 수도 있음을 알 수 있다

III. 성능 평가 모델

H-SDA에서 같은 군에 속한 두 노드간의 통신방식과 다른 군에 속한 두 노드간의 통신방식이 다르므로 내부패킷의 지연시간과 외부패킷의 지연시간을 따로 계산한다.

성능평가의 편의를 위해 각 군이 l 개씩 g 개의 군으로 구성된 경우를 가정한다. 성능평가의 관점에서 본 사용자노드의 내부모델이 그림 1에 나타나 있는데, 표 2에서 사용자노드 $n_{1,1}$ 에 해당한다.

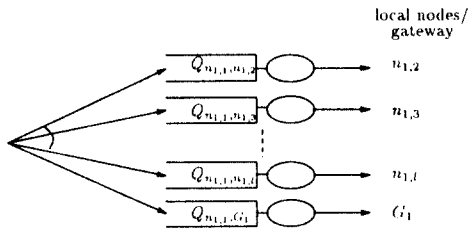


그림 1. 사용자노드 모델($n_{1,1}$)
Figure 1. Model of user node($n_{1,1}$)

사용자노드는 하나의 수신버퍼와 l 개의 송신버퍼를 유지한다고 가정한다. $l-1$ 개의 송신버퍼는 같은 군에속한 사용자노드로 루팅되는 내부패킷을 처리하고, 나머지 하나는 그 군의 게이트웨이로 루팅되는 외부패킷을 처리한다. 여기서 패킷 전송을 위한 채널/타임슬롯이 각 목적지에 대해 별도로 할당되어 있으므로 사용자노드의 각 송신버퍼는 독립적인 대기시스템(queueing system)으로 모델링될 수 있다.

그림 2에는 전형적인 게이트웨이의 내부모델을 제시하였는데, 표 2의 G_1 에 해당한다. 각 게이트웨이는 군 내부의 사용자노드로부터 다른 군의 게이트웨이 방향으로 $g-1$ 개의 송신버퍼를, 반대 방향으로 l 개의 송신버퍼를 유지한다고 가정한다. 게이트웨이에 그 군 내부의 사용자노드로부터 다른 군으로 전송되기 위해 도착한 외부패킷은 그 목적지에 따른 다른 게이트웨이로 향한 송신버퍼로 분류된다. 반대로 나

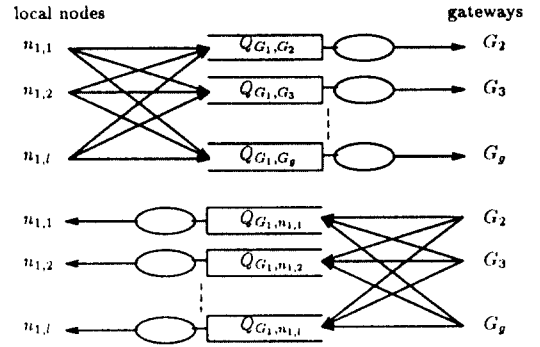


그림 2. 게이트웨이 모델(G_1)
Figure 2. Model of gateway(G_1)

는 게이트웨이로부터 도착하는 패킷들은 그 목적지에 따른 군 내부의 각 사용자노드를 향한 송신버퍼로 분류된다.

각 사용자노드에 도착하는 트래픽은 성능평가를 간단히 하기 위해 균질하다고(homogeneous) 가정한다. 즉, 외부로부터의 트래픽은 평균 도착률이 λ 인 Poisson 프로세스로 모든 사용자노드에 대해 같다고 가정한다. 또한, 각 사용자노드에 도착하는 트래픽 중에서 내부패킷의 비율을 트래픽 편중화 인수 α 라 정의하고, 모든 사용자노드에서 같은 비율을 갖는다고 가정한다. 또한, 내부패킷의 목적지는 같은 군에 속한 $l-1$ 개의 사용자노드에 대해 같은 확률 $1/(l-1)$ 로 선택한다고 가정한다. 따라서, 모든 사용자노드에서 자신과 같은 군에 속한 사용자노드로 가는 송신버퍼에는 Poisson 프로세스의 분해 원리에 따라 평균 도착률이 $\lambda\alpha/(l-1)$ 인 Poisson 프로세스로 트래픽이 도착하게 된다[12]. 또한, 외부패킷을 처리하는 송신버퍼에는 평균이 $\lambda(1-\alpha)$ 인 Poisson 프로세스로 트래픽이 도착한다[12]. 외부패킷도 다른 군에 속한 $lg-l$ 의 사용자노드에 대해 같은 확률 $1/(lg-l)$ 로 목적지가 결정된다고 가정한다.

성능 평가 인자는 아래와 같이 정의하는데, 각 지연시간은 기대값으로 정의된 것이다. 또한, F-SDA에서 모든 사용자노드로 1 홉에 패킷을 전달할 수 있으므로 내부패킷과 외부패킷의 구분이 없지만 H-SDA와의 성능 비교를 위해 H-SDA에서의 노드 분할에 대응되게 F-SDA에서도 가상적으로 노드들을 군으로 분할하여 가상내부 패킷과 가상외부패킷을 정의하고 각각의 지연시간을 계산한다.

- $E[D_h^L]$: H-SDA의 내부패킷 지연시간.
- $E[D_h^R]$: H-SDA의 외부패킷 지연시간.
- $E[D_h]$: H-SDA의 패킷 지연시간.
- $E[D_f^L]$: F-SDA의 가상내부패킷 지연시간.
- $E[D_f^R]$: F-SDA의 가상외부패킷 지연시간.
- $E[D_f]$: F-SDA의 패킷 지연시간.

위의 정의를 이용하면 $E[D_h]$ 와 $E[D_f]$ 는 다음과 같이 계산된다:

$$E[D_h] = \alpha E[D_h^L] + (1-\alpha) E[D_h^R] \quad (3)$$

$$E[D_f] = \alpha E[D_f^L] + (1-\alpha) E[D_f^R]. \quad (4)$$

3.1 내부 패킷의 지연 시간

사용자노드의 내부모델을 나타내는 그림 1에서 각 대기행렬은 독립적으로 동작하므로 입력부하가 각 사용자노드에서 동일하다는 가정하에 $l-1$ 개의 대기행렬 중에서 대표적인 하나를 분석함으로써 내부패킷에 대한 지연시간을 계산할 수 있다. $Q_{n_{si}, n_{sj}}$ 를 사용자노드 n_{si} 로부터 같은 군의 사용자노드 n_{sj} 로 가는 내부패킷을 처리하는 대기행렬이라 하자. $Q_{n_{si}, n_{sj}}$ 에 도착하는 패킷들은 평균 도착률이 $\lambda^l = \lambda\alpha/(l-1)$ 인 Poisson 프로세스이다. 그리고, $Q_{n_{si}, n_{sj}}$ 에 도착한 패킷의 서비스 시작 시간은 T_h^L 주기 내에서 미리 결정되어 있다. 비록 $Q_{n_{si}, n_{sj}}$ 가 비어있는 상태에서 도착한 패킷이라도 H-SDA 프로토콜에서 정해진 자신의 전송 타임슬롯을 기다려야만 한다. 즉, $Q_{n_{si}, n_{sj}}$ 는 $T_h^L - 1$ 타임슬롯 동안의 휴식과 1 타임슬롯 동안의 서비스를 교대로 반복하는 대기행렬이다.

이 시스템에서의 평균 큐길기와 지연시간은 [13]의 결과로부터 구해질 수 있다. 노드사이의 최대전파 지연시간을 D_{propa} 라 하자. $Q_{n_{si}, n_{sj}}$ 의 평균 큐길기를 $E[N^L]$ 라 하고 평균지연시간을 $E[D_h^L]$ 이라 하면 각각은 다음과 같이 계산될 수 있다.

$$E[N^L] = \frac{-2(\lambda^l T_h^L)^2 + 3(\lambda^l)^2 T_h^L + 3\lambda^l T_h^L + 3(\lambda^l)^2 + 6\lambda^l}{3(2 - \lambda^l(T_h^L - 2))}$$

$$+ \frac{(\lambda^l T_h^L)^2}{2(1 - \lambda^l T_h^L)}$$

$$E[D_h^L] = \frac{E[N^L]}{\lambda^l} + D_{propa}$$

$$= \frac{-2\lambda^l(T_h^L)^2 + 3\lambda^l T_h^L + 3T_h^L + 3\lambda^l + 6}{3(2 - \lambda^l(T_h^L - 2))}$$

$$+ \frac{\lambda^l(T_h^L)^2}{2(1 - \lambda^l T_h^L)} + D_{propa}. \quad (5)$$

또한, F-SDA 프로토콜에서의 가상내부패킷의 지연시간 $E[D_f^L]$ 도 식 (5)에서 T_h^L 대신에 T_f 를 대입함으로써 바로 계산된다.

3.2 외부패킷의 지연시간

내부패킷과 비교하여 외부패킷은 3 홉을 거쳐 최종 목적지까지 전달되므로 지연시간도 3 부분으로 구성된다. 즉, 패킷이 발생한 최초 사용자노드로부터 송신지 게이트웨이로 전송되는데 걸리는 지연시간과 그 게이트웨이에서 수신지 게이트웨이로 전송되는데 걸리는 지연시간 및 수신지 게이트웨이에서 최종 목적지 사용자노드로 전송되는데 걸리는 지연시간의 3 부분으로 구성된다. 외부패킷에 관한 큐잉네트워크를 정확하게 분석하기는 쉽지 않으므로 근사 해석 모델을 제시한다. 먼저, 큐잉네트워크의 독립성원리를 가정하는데, 송신지 게이트웨이에 도착한 외부패킷은 목적지 수신지 게이트웨이를 그 외부패킷의 목적지와는 무관하게 $1/(g-1)$ 의 확률로 다시 선택한다고 가정하고, 수신지 게이트웨이에 도착한 외부패킷도 $1/l$ 의 확률로 목적지 사용자노드를 다시 선택한다고 가정한다[14]. 이러한 가정하에 송신지 게이트웨이와 수신지 게이트웨이 내의 전송비율을 독립된 시스템으로 근사화시켜 외부패킷의 지연시간을 구하고 이로부터 큐잉네트워크에서 외부패킷의 평균지연시간을 추정한다.

우선, 다음과 같은 기호를 정의하자.

- $Q_{i,j}$: 노드 i 에서 노드 j 로 가는 패킷을 처리하는 대기행렬 ($i, j \in \{n_{km} \text{ or } G_s \mid k=1, 2, \dots, g, m=1, 2, \dots, l, s=1, 2, \dots, g, i \neq j\}$).
- $E[D^{i,j}]$: 대기행렬 $Q_{i,j}$ 에서의 대기지연시간 ($i, j \in \{n_{km} \text{ or } G_s \mid k=1, 2, \dots, g, m=1, 2, \dots, l, s=1, 2, \dots, g, i \neq j\}$).
- $E[D^{G_s}]$: 사용자노드로부터 송신지 게이트웨이로의 외부패킷의 대기지연시간.
- $E[D^{G_s}]$: 송신지 게이트웨이에서 수신지 게이트웨이로의 외부패킷의 대기지연시간.
- $E[D^{G_s}]$: 수신지 게이트웨이에서 목적지 사용자노드로의 외부패킷의 대기지연시간.
- p^{G_s, G_d} : Q_{G_s, G_d} 에 외부패킷이 발생할 수 있는 슬롯 경계에서 실제로 패킷이 도착할 정상상태확률 (steady state probability).

- $P_0^{n_i, n_{i,d}}$: T_h 시간동안 $Q_{n_i, n_{i,d}}$ 에 외부패킷이 도착하지 않을 정상상태 확률.
- $E[N^{d, n_i, n_{i,d}}]$: 외부패킷이 $Q_{n_i, n_{i,d}}$ 을 떠나는 시점에서 $Q_{n_i, n_{i,d}}$ 에 남아있는 외부패킷 수.

또한, 각 노드사이의 전파지연시간을 고려하면 외부패킷의 평균지연시간, $E[D_h^k]$, 은 다음과 같이 주어진다.

$$E[D_h^k] = E[D^{(1)}] + E[D^{(2)}] + E[D^{(3)}] + 3D_{prop}. \quad (6)$$

(1) $E[D^{(1)}]$

사용자노드 n_i 의 외부패킷을 처리하는 전송버퍼 $Q_{n_i, n_{i,d}}$ 에 도착하는 외부패킷은 평균도착율이 $\lambda^k \cdot \lambda(1-\alpha)$ 인 Poisson 분포를 따르므로 $Q_{n_i, n_{i,d}}$ 의 구조는 내부패킷을 처리하는 대기행렬의 구조와 동일하므로 $Q_{n_i, n_{i,d}}$ 에서의 대기지연시간, $E[D^{n_i, n_{i,d}}]$ 은 식 (5)의 T_h^k 와 λ^k 대신에 T_h 와 λ^k 을 대입함으로써 직접 구할 수 있다. 또한, 균질한 트래픽 부하를 가정하였으므로 $E[D^{(1)}]$ 은 $E[D^{n_i, n_{i,d}}]$ 와 같다.

(2) $E[D^{(2)}]$ 계산

송신지 게이트웨이에서 수신지 게이트웨이로의 대기지연시간의 전체 평균치 $E[D^{(2)}]$ 을 계산하기 전에 특정 게이트웨이 G_s 에서 G_d 까지의 지연시간 $E[D^{s, d}]$ 를 먼저 추정한다. 송신지 게이트웨이 G_s 에 도착하는 외부패킷들은 모두 s 에 속하는 l 개의 사용자노드들로부터 올 수 있으며, 이 패킷들의 도착 가능한 시간은 H-SDA의 채널/타입슬롯 할당표에 의해 한정되어 있다. 또한, G_s 에 도착한 패킷들은 $1/(g-1)$ 의 확률로 수신지 게이트웨이를 결정한다. 즉, $Q_{n_i, n_{i,d}}$ 의 관점에서 볼 때, T_h 싸이클안에 미리 정해진 l 개의 슬롯강계에서만 패킷도착이 있을 수 있으며, 또한 미리 정해진 타입슬롯에서 서비스된다. 대기행렬 $Q_{n_i, n_{i,d}}$ 에 외부패킷이 도착하고 서비스받는 시간관계가 표 2의 $Q_{n_i, n_{i,d}}$ 의 경우에 그림 3에 나타나있다.

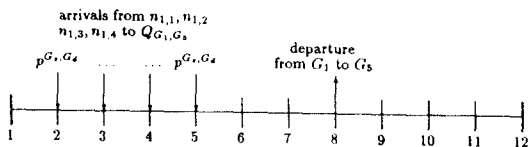


그림 3. 타이밍도($Q_{n_i, n_{i,d}}$)
Figure 3. Timing diagram($Q_{n_i, n_{i,d}}$)

$Q_{n_i, n_{i,d}}$ 에 도착하는 프로세스는 $Q_{n_i, n_{i,d}}$ 로부터 서비스받고 떠나가는 프로세스이므로 $Q_{n_i, n_{i,d}}$ 의 출발프로세스(departure process)의 특성을 관찰한다. $p_0^{n_i, n_{i,d}}$ 는 정상상태(steady state)에서 $Q_{n_i, n_{i,d}}$ 에 T_h 동안 패킷도착이 발생하지 않을 확률이며, $\pi_0^{d, n_i, n_{i,d}}$ 는 정상상태에서 $Q_{n_i, n_{i,d}}$ 을 떠나는 패킷이 $Q_{n_i, n_{i,d}}$ 가 빈 상태를 관찰할 확률이라 하면 $Q_{n_i, n_{i,d}}$ 의 출발프로세스의 분석으로부터 각각 다음과 같이 계산된다 [17]:

$$p_0^{n_i, n_{i,d}} = e^{-\lambda^k T_h}$$

$$\pi_0^{d, n_i, n_{i,d}} = \frac{2(1-\lambda^k T_h)}{2-\lambda^k(T_h+2)}$$

또한, 정상상태에서 $Q_{n_i, n_{i,d}}$ 에 패킷도착이 발생할 수 있는 l 개의 슬롯강계에서 실제로 패킷이 도착할 확률을 포함한 트래픽 입력분포와 독립원리를 가정하였으므로 모두 같은 값 $p^{s, d}$ 을 갖는데, 다음과 같이 구해진다 [17]:

$$p^{s, d} = \frac{1-p_0^{n_i, n_{i,d}}}{1-p_0^{n_i, n_{i,d}}(1-\pi_0^{d, n_i, n_{i,d}})} \cdot \frac{1}{(g-1)}$$

이로부터 $Q_{n_i, n_{i,d}}$ 에 패킷이 정상상태에서 Bernoulli 프로세스로 도착한다고 근사화하여 Embedded Markov Chain 기법을 사용한다. $Q_{n_i, n_{i,d}}$ 에서 서비스받고 떠나는 패킷이 관찰하는 평균 큐길이를 $E[N^{d, n_i, n_{i,d}}]$ 라 하면 Embedded Markov Chain의 분석으로부터 다음과 같이 유도된다 [17]:

$$E[N^{d, n_i, n_{i,d}}] = \frac{(l-1)p^{s, d}}{2(1-lp^{s, d})}$$

$E[S_{n_i, n_{i,d}}^a]$ 을 $Q_{n_i, n_{i,d}}$ 가 비어있는 상태에서 도착하는 패킷의 평균 지연시간이라 하고, $E[R_i]$ 을 $Q_{n_i, n_{i,d}}$ 에 i 개의 패킷이 있는 상태에서 새로 패킷이 도착했을 때 맨 앞에 있던 패킷의 잔여서비스시간(remaining service time)이라 정의하면, $Q_{n_i, n_{i,d}}$ 에서는 $E[S_{n_i, n_{i,d}}^a] = E[R_i]$ 가 된다. 또한, $Q_{n_i, n_{i,d}}$ 은 한 순간에 하나씩만 상태가 변할 수 있으므로, 주 큐에 있는 패킷의 수가 한 순간에 하나 증가하거나 하나 감소하는 시스템이므로 도착하는 패킷의 관점과 떠나가는 패킷의 관점이 같다. 따라서, $Q_{n_i, n_{i,d}}$ 에 도착하는 패킷이 관찰하는 평균큐길이를 $E[N^{a, n_i, n_{i,d}}]$ 라 하면, $E[N^{a, n_i, n_{i,d}}] = E[N^{d, n_i, n_{i,d}}]$ 가 된다 [15]. 이상의 관찰로부터 $Q_{n_i, n_{i,d}}$ 에서의 평균지연 시간은 다음과 같이 유도된다.

$$\begin{aligned}
 E[D^{G_s, G_d}] &= \pi_0^s E[S_{G_s, G_d}^s] + \sum_{i=1}^L \pi_i^s (iT_h + E[R_i]) \\
 &= \pi_0^s E[S_{G_s, G_d}^s] + \sum_{i=1}^L \pi_i^s (iT_h + E[S_{G_s, G_d}^s]) \\
 &= E[S_{G_s, G_d}^s] + T_h \sum_{i=1}^L \pi_i^s i \\
 &= E[S_{G_s, G_d}^s] + T_h E[N^{d, G_s, G_d}] \\
 &= E[S_{G_s, G_d}^s] + T_h E[N^{d, G_s, G_d}].
 \end{aligned}$$

$De(Q_{G_s, G_d})$ 를 H-SDA의 채널/타임슬롯 할당표에서 Q_{G_s, G_d} 의 서비스 종료시간이라고 하고, $Ar(Q_{n_m, G_s}, Q_{G_s, G_d})$ 를 Q_{n_m, G_s} 로부터 Q_{G_s, G_d} 에 패킷도착이 발생할 수 있는 시간이라고 하자. $De(Q_{G_s, G_d})$ 와 $Ar(Q_{n_m, G_s}, Q_{G_s, G_d})$ 를 이용하면, $E[S_{G_s, G_d}^s]$ 은 H-SDA의 채널/타임슬롯 할당표로부터 아래와 같이 쉽게 구할 수 있다:

$$\begin{aligned}
 E[S_{G_s, G_d}^s] &= \\
 &= \frac{\sum_{i=1}^L |De(Q_{G_s, G_d}) + T_h - Ar(Q_{n_m, G_s}, Q_{G_s, G_d})| \text{MOD } T_h}{L}.
 \end{aligned}$$

마지막으로 외부패킷의 전송에 관한 큐잉네트워크에서 송신지 게이트웨이로부터 목적지 게이트웨이까지의 평균지연시간, $E[D^{G_s}]$ 은 다음과 같이 구해진다:

$$E[D^{G_s}] = \frac{\sum_{d=1}^g \sum_{s=s}^d E[D^{G_s, G_d}]}{g \times (g-1)}. \quad (7)$$

(3) $E[D^{G_s}]$

먼저, $E[D^{G_s, G_d}]$ 을 구하는데 $E[D^{G_s, G_d}]$ 와 마찬가지로 방법으로 계산한다. Q_{G_s, G_d} 의 분석과 마찬가지로 목적지 게이트웨이에 도착하는 패킷들도 그 패킷의 원래 목적지와는 상관없이 다시 확률 $1/l$ 로 목적지 사용자노드가 다시 정해진다고 가정한다 [14]. 또한, Q_{G_s, G_d} 에 도착 프로세스를 Bernoulli 프로세스를 따른다고 근사화한다. 먼저, $p_0^{G_s, G_d}$ 을 정상상태에서 T_h 동안 Q_{G_s, G_d} 에 패킷도착이 발생하지 않을 확률이라고 하고, π_0^{d, G_s, G_d} 을 정상상태에서 Q_{G_s, G_d} 에서 서비스받고 떠나가는 패킷이 Q_{G_s, G_d} 가 빈 상태를 관찰할 확률이라면, 각각은 다음과 같이 구해진다:

$$\begin{aligned}
 p_0^{G_s, G_d} &= (1 - p^{G_s, G_d})^l \\
 \pi_0^{d, G_s, G_d} &= \frac{(1 - l p^{G_s, G_d})(1 - (1 - p^{G_s, G_d})^l)}{l p^{G_s, G_d}(1 - p^{G_s, G_d})^l}.
 \end{aligned}$$

패킷도착이 발생할 수 있는 타임슬롯 경계에서 실제

로 패킷도착이 있을 확률을 p^{G_s, G_d} 이라 하면, p^{G_s, G_d} 은 Q_{G_s, G_d} 로부터 서비스받고 떠나는 출발 프로세스의 분석으로부터 다음과 같이 계산된다:

$$p^{G_s, G_d} = \frac{1 - p_0^{G_s, G_d}}{1 - p_0^{G_s, G_d}(1 - \pi_0^{d, G_s, G_d})} \frac{1}{l}.$$

$E[N^{d, G_s, G_d}]$ 을 정상상태에서 Q_{G_s, G_d} 에서 서비스받고 떠날 때 Q_{G_s, G_d} 에 남아있는 평균 패킷 수라고 정의하면, $E[N^{d, G_s, G_d}]$ 는 Q_{G_s, G_d} 의 지연시간 분석과 같은 과정을 거쳐 다음과 같이 구해진다 [17]:

$$E[N^{d, G_s, G_d}] = \frac{(g-2)p^{G_s, G_d}}{2(1 - p^{G_s, G_d}(g-1))}.$$

Q_{G_s, G_d} 의 지연시간 분석과 마찬가지로의 과정을 밟으면,

$$E[D_{G_s, G_d}] = E[S_{G_s, G_d}^s] + T_h E[N^{d, G_s, G_d}]. \quad (8)$$

여기서 $E[S_{G_s, G_d}^s]$ 는 Q_{G_s, G_d} 가 비어있는 상태에서 도착한 패킷의 서비스시간으로 정의된 것인데, $E[S_{G_s, G_d}^s]$ 은 H-SDA의 채널/타임슬롯 할당표로부터 다음과 같이 구해진다:

$$\begin{aligned}
 E[S_{G_s, G_d}^s] &= \\
 &= \frac{\sum_{s=1, s \neq d}^g |De(Q_{G_s, G_d}) + T_h - Ar(Q_{G_s, G_d}, Q_{G_s, G_d})| \text{MOD } T_h}{g-1}.
 \end{aligned}$$

식 (8)으로부터 송신지 게이트웨이에서의 최종 목적지 사용자노드까지의 평균 지연시간 $E[D^{G_s}]$ 은 다음과 같이 구해진다:

$$E[D^{G_s}] = \frac{\sum_{d=1}^g \sum_{j=1}^L E[D_{G_s, G_d}]}{g \times l}. \quad (9)$$

$E[D^{G_s}]$ 와 식 (7), (9)를 식 (6)에 대입함으로써 외부 패킷의 평균대기지연시간 $E[D_h^s]$ 를 구할 수 있다. 또한, 식 (5)와 식 (6)의 결과를 식 (4)에 대입함으로써 $E[D_h]$ 를 구할 수 있다.

IV. 성능 평가 결과

제안된 H-SDA 프로토콜의 성능을 평가하고 F-SDA 프로토콜과의 비교를 위해 몇가지 통신망 구성에 대한 정량적 분석 결과를 본 절에서 기술한다. 통신망 구성은 $l \times g \times b \times r$ 로 나타낼 수 있는데 l 은 균의 크기, g 는 균의 수, b 는 채널 수이다. 그리고 r 은 외부

패킷을 위한 타임슬롯 수를 의미하는데 TDM 싸이클 주기안에 각 사용자노드가 r 번 외부패킷을 전송할 기회가 있음을 의미한다. 예로 들 통신망은 전체 노드의 수가 20 일때, $l \times g$ 가 4×5 로 분할된 경우이며, 채널 수는 2 개와 20인 경우에 조사하였다. 4×5 이고 채널 수가 2 일 때 외부패킷을 위한 타임슬롯을 다르게 할당할 경우에 대해서도 조사하였다. 노드사이의 전파지연시간은 0으로 가정하였다(V 질 참조). 본 절에서 제시한 성능 평가 인자는 같은 군에 속한 두 노드사이의 처리율/지연시간 특성과 다른 군에 속한 두 노드사이의 처리율/지연시간 특성 및 트래픽 편중화 인수 α 의 변화에 따른 지연시간 특성이다. 시뮬레이션 결과는 batch-mean 방법으로 95%의 신뢰구간으로 정확도가 0.1을 만족하도록 수행한 것이다 [16].

가) $(l, g, b, r) \equiv (4, 5, 2, 1)$: 두 노드사이의 내부패킷과 외부패킷의 처리율과 지연시간 관계를 해석적 방법과 시뮬레이션으로 구한 결과를 그림 4에 제시하였다.

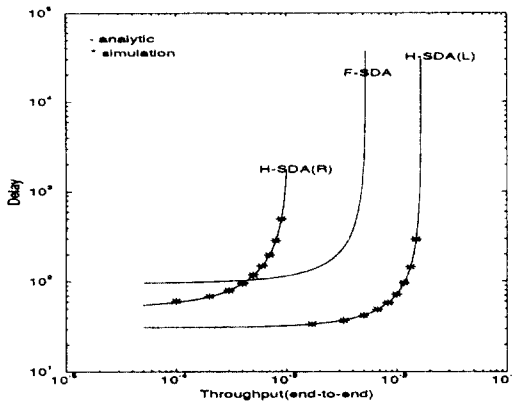


그림 4. 내부패킷과 외부패킷의 처리율 대 지연시간($l=4, g=5, b=2, r=1$)

Figure 4. Delay vs. throughput($l=4, g=5, b=2, r=1$)

그림 4에서 H-SDA(L)는 같은 군에 속한 두 노드사이의 처리율과 지연시간의 관계를 나타내며 H-SDA(R)는 서로 다른 군에 속한 두 노드사이의 성능을 보여주고 있다. F-SDA는 내부패킷과 외부패킷의 구분이 없으므로 하나만 제시하였다. 처리율(throughput)은 특정 목적지에 대해 타임슬롯당 몇개의 패킷을 전송할 수 있는가를 나타내며, 지연시간은 타임슬

롯단위로 나타내었다. 그림 4에서 시뮬레이션 결과는 H-SDA에서의 내부패킷과 외부패킷에 관한 것만 제시하였는데, F-SDA의 시뮬레이션은 H-SDA의 내부패킷에 대한 성능 평가 모델과 구조가 같으므로 생략하였다. 그림 4에서 보듯이 내부패킷과 외부패킷에 대해서 해석적 방법에 의한 결과와 시뮬레이션에 의한 결과가 거의 일치함을 볼 수 있다. 내부패킷의 지연시간은 두 가지 방법에 의한 결과가 1%의 오차 한계내에서 일치하였고, 외부패킷의 지연시간은 10%의 오차 한계 내에서 두 가지 방법의 결과가 일치함을 확인하였다. 내부패킷에 관한 성능 평가 모델은 정확한 해석 모델인 반면에 외부패킷의 경우는 근사 모델이기 때문에 오차의 범위에 차이가 있는 것으로 예측된다. 이 구성에서 $T_h^L = T_h^R = 60$ 이고 $T_f = 190$ 이다(표 3 참조). F-SDA보다 H-SDA가 트래픽 편중화 현상이 강한 두 노드 사이에 약 3 배 정도 전송 슬롯을 많이 할당받으므로 내부패킷의 처리율/지연시간 성능이 그만큼 개선된다. 반면에 각 노드는 외부패킷을 60 타임슬롯마다 하나씩 전송하게 되는데, 서로 다른 군에 속한 임의의 두 노드간의 처리율 측면에서는 목적지 분포가 uniform할 때, $(1/T_h^R)/(lg-l)$ 만큼 패킷을 전송할 수 있다. 그러나, 이 임계치는 외부패킷의 목적지가 같은 확률로 선택된다는 가정하에 나온 것이며, 실제로는 다른 군에 속한 하나의 사용자 노드로 최대 $1/T_h^R$ 즉, $1/60$ 만큼의 패킷전송이 가능하다. 즉, 외부패킷의 목적지 분포에 따라서는 F-SDA 프로토콜보다 H-SDA 프로토콜이 외부패킷을 더 많이 그리고 빨리 전송할 수도 있다. 또한, 그림 4에 나타나 있듯이 외부패킷의 경우에도 트래픽 부하가 낮을 때에는 F-SDA에서보다 H-SDA에서의 지연시간이 더 작다. 이것은 TDM 싸이클 주기가 약 3 배 정도 줄어든 효과에 기인한다. 즉, 외부패킷은 3 홉을 거쳐 전송됨에도 불구하고 트래픽 부하가 적을 때는 점유하는 중간 노드에서의 대기지연시간(queueing delay)이 거의 없으므로 줄어든 싸이클 주기로 인해 F-SDA에서보다 더 빨리 전송되는 특성이 있음을 알 수 있다.

H-SDA의 전체적인 성능은 트래픽 편중화 정도에 따라 달라지므로 α 의 변화에 따른 지연시간 성능을 그림 5에 제시한다.

그림에서 S_h 는 H-SDA에서의 입력부하를 의미하며, S_f 는 F-SDA에서의 입력부하를 나타내는데, 타임슬롯당 도착하는 패킷의 수를 의미한다. 또한 지연시간은 타임슬롯 단위로 나타내었다. 그림 5에서 알

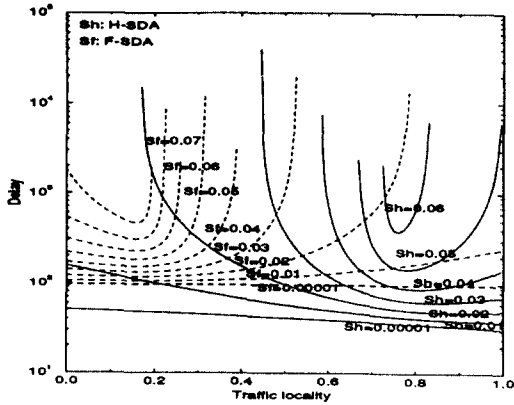


그림 5. 트래픽편중화 대 지연시간($l=4, g=5, b=2, r=1$)
Figure 5. Delay vs. traffic locality($l=4, g=5, b=2, r=1$)

수 있듯이, 같은 트래픽 부하 조건에서는 트래픽의 편중화가 심할수록 H-SDA의 지연시간이 F-SDA의 지연시간보다 더 작다. 대략적으로 $\alpha > 0.5$ 일 때 같은 트래픽 부하에 대해서 H-SDA가 F-SDA보다 더 나은 지연시간 성능을 보임을 알 수 있다.

F-SDA의 지연시간 곡선은 $\alpha = [(1/T_f) \times (l-1)] / [(1/T_f) \times (l-1) + (1/T_f) \times (lg-l)] = 0.16$ 근처에서 최소값을 가지며 아래로 볼록하다. α 가 0에 가까울수록 가상내부패킷 전송에 할당된 타임슬롯 수에 비해 도착하는 가상내부패킷은 적은 반면, 상대적으로 가상외부패킷 전송에 할당된 타임슬롯 수에 비해 도착하는 가상외부패킷은 많아진다. 따라서 가상내부패킷의 경로에 비해 가상외부패킷의 경로가 상대적으로 병목 현상을 나타낸다. 반면에 α 가 0.16 보다 커질 때는 가상내부패킷의 경로가 상대적인 병목 현상을 나타낸다. α 가 0.16 부근에서는 가상내부패킷과 가상외부패킷이 그들에게 할당된 타임슬롯 수에 같은 비율로 도착하는 시점이므로 전체 지연시간에 기여하는 가상내부패킷과 가상외부패킷의 기여도가 같아서 지연시간이 최소값을 나타낸다. 한편, H-SDA의 지연시간 곡선은 $\alpha = [(1/T_h^L) \times (l-1)] / [(1/T_h^L) \times (l-1) + 1/T_h] = 0.75$ 부근에서 최소값을 가지며 F-SDA와 비슷한 특성을 보여주고 있다. 즉, 동일한 도착율에 대해 $\alpha = 0.75$ 보다 작아질수록 전체 지연시간에 대한 내부패킷의 기여도가 외부패킷의 기여도보다 더 커지고, 0.75보다 커질수록 그 반대의 현상을 나타낸다. 한편, 0.75 부근에서는 내부패킷과 외부패킷에 할당된 타임슬롯 수에 비례하여 내부패킷과

외부패킷이 도착하는 시점이므로 지연시간은 최소값을 나타낸다.

나) $(l, g, b, r) \equiv (4, 5, 20, 1)$: 이 구성은 채널/타임슬롯 할당표 2에 해당하는데, 가)의 구성과 비슷하나, 채널 수가 20으로 많은 경우이다. 내부패킷과 외부패킷의 지연시간과 α 의 변화에 따른 전체 지연시간이 각각 그림 6, 그림 7에 제시되어 있다.

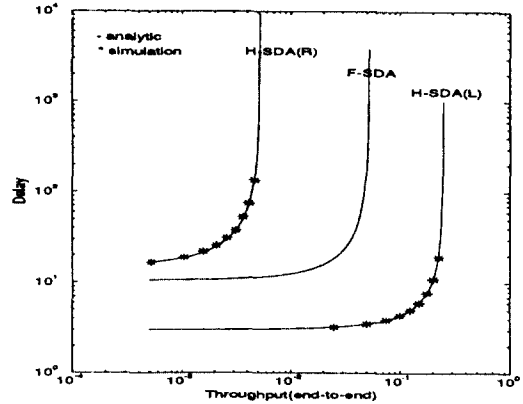


그림 6. 내부패킷과 외부패킷의 처리율 대 지연시간($l=4, g=5, b=20, r=1$)

Figure 6. Delay vs. throughput($l=4, g=5, b=20, r=1$)

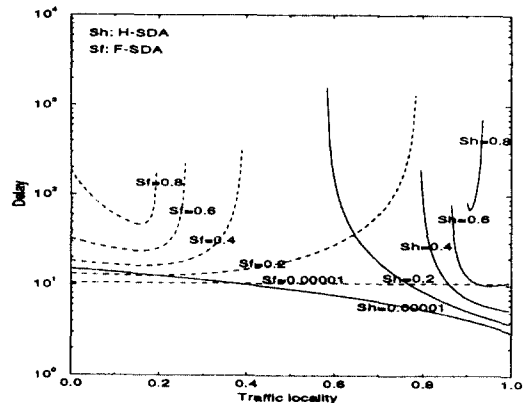


그림 7. 트래픽편중화 대 지연시간($l=4, g=5, b=20, r=1$)
Figure 7. Delay vs. traffic locality($l=4, g=5, b=20, r=1$)

이 구성은 이용가능한 채널의 수가 20 이므로 전체 TDM 싸이클의 주기는 가) 구성과 비교하여 더 줄어

든다. 즉, $T_f=19$ 이고, $T_k^L=4$, $T_h=12$ 가 된다. 내부 패킷의 관점에서 볼 때 TDM 싸이클 주기는 $T_k^L=4$ 로서 약 5 배 정도 T_f 보다 작아짐으로 인해 F-SDA보다 그만큼 많은 양의 내부패킷을 같은 군 내의 노드로 빨리 전송할 수 있음을 보이고 있다. 반면에 외부 패킷은 $T_h(=12)$ 마다 한번씩의 전송기회를 갖는다. 균질한 트래픽 도착을 가정했을 때는 $(1/12)/16$ 만큼의 외부패킷을 다른 군에 속한 특정 사용자노드에 전송할 수 있다. 그러나 가)의 구성과 마찬가지로 복직지 분포에 따라서는 다른 군의 특정 사용자노드로 12 타임슬롯마다 한번씩 패킷전송이 가능하여 19 타임슬롯마다 한번씩 패킷 전송이 가능한 F-SDA보다 더 나은 성능을 보일 수도 있다. 그림 7은 가)의 결과와 비슷한 결과를 보여주고 있는데 같은 패킷 도착을 조건하에서 $\alpha > 0.6$ 이면 H-SDA가 F-SDA보다 더 나은 지연시간 특성을 보임을 알 수 있다. 그리고 $\alpha = 0.9$ 부근에서 지연시간이 최소값을 보이는 것은 가)의 구성에서와 같은 이유에 기인한다.

다) $(l, g, b, r) \equiv (4, 5, 2, 2)$: 이 구성은 가)와 비슷하지만, 외부패킷 전송을 위한 타임슬롯을 많이 할당하여 한 TDM 싸이클에 두 번 외부패킷을 전송할 수 있도록 한 것이다. H-SDA에서 하나의 TDM 싸이클에 두 번 외부패킷을 전송할 수 있으면, 사용자노드에서 송신지 게이트웨이로 $(l \times g) \times 2$ 만큼, 게이트웨이간의 통신을 위해서 $(g(g-1)) \times 2$ 만큼, 수신지 게이트웨이에서 최종 복직지 사용자노드로 $(g \times l) \times 2$ 만큼 모두 120 개의 타임슬롯이 필요하여 2 개의 채널에 할당하면 채널/타임슬롯 할당표에서 60을 차지하게 된다. 따라서 가)와 비교하여 TDM 싸이클 주기는 30이 늘어난 90이 된다. 90 타임슬롯내에 같은 군에 속한 사용자노드 사이에는 내부 패킷을 하나씩 보낼 수 있으므로 $T_k^L=90$ 이 된다. 반면에, 90 타임슬롯 내에 각 사용자노드는 외부패킷을 두 번 전송할 기회가 있는데, 이 두 번의 전송기회가 TDM 싸이클에서 같은 간격으로 할당된 예를 생각하면 $T_h=45$ 가 된다. 이 구성에서의 패킷도착율과 지연시간의 관계가 그림 8, 9에 나타나 있다.

T_k^L 이 가)에 비해 약간 증가한 관계로 인해 내부패킷의 지연시간 특성이 약간 나빠진 반면, T_h 는 감소하였으므로 외부패킷의 처리율/지연시간 특성은 개선된다. α 의 변화에 따른 지연시간 특성이 그림 9에 나타나 있는데, 대략 $\alpha > 0.4$ 일 때 H-SDA가 F-SDA보다 더 나은 지연시간 특성을 보이고 있다. 그림 5와 그림 9을 비교해보면, 가)의 구성에서는 $\alpha > 0.5$ 인 영

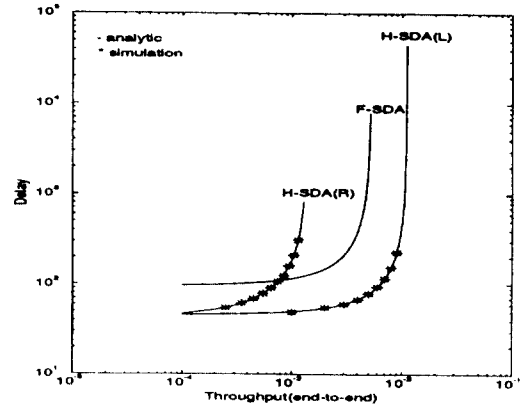


그림 8. 내부패킷과 외부패킷의 처리율 대 지연시간($l=4, g=5, b=2, r=2$)

Figure 8. Delay vs. throughput($l=4, g=5, b=2, r=2$)

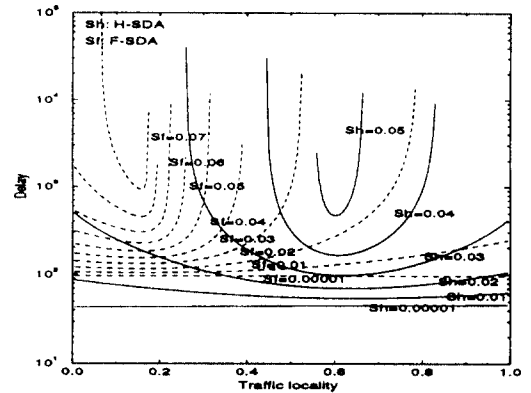


그림 9. 트래픽편중화 대 지연시간($l=4, g=5, b=2, r=2$)

Figure 9. Delay vs. traffic locality($l=4, g=5, b=2, r=2$)

역에서 다)의 구성에서는 $\alpha > 0.4$ 인 영역에서 H-SDA가 F-SDA보다 더 나은 특성을 보임을 알 수 있다. 즉, 외부패킷의 전송을 위한 타임슬롯을 더 할당함으로써 α 가 낮은 범위에서도 H-SDA의 성능이 F-SDA보다 더 나은 지연시간 특성을 보이도록 할 수 있다. 또한, $\alpha = 0.6$ 에서 지연시간이 최소값이 됨을 알 수 있는 것은 가)와 마찬가지로 이유이다.

지금까지 몇 가지 통신망 구성에 대한 성능 분석 결과를 제시하였는 바, H-SDA는 F-SDA에 비해 같은 군에 속한 사용자노드사이의 통신에 더 많은 전송 기회를 주고, 서로 다른 군에 속한 사용자노드 사이

의 통신에 상대적으로 적은 전송 기회를 주는 알고리즘이다. 제안된 H-SDA에서 트래픽 편중화 현상이 강한 관계에 있는 두 노드간에 보낼 수 있는 패킷의 양은 TDM 싸이클 주기의 축소에 비례하여 증가한다. 게다가, 줄어든 TDM 싸이클 주기로 인해 외부패킷의 지연시간도 부하가 적을 때 F-SDA보다 더 작아진다. 또한, 부하가 많을 때도 목적지 분포에 따라서는 F-SDA보다 더 많은 외부패킷을 더 빨리, 많이 전송할 수 있는 장점이 있다. 또한, H-SDA는 F-SDA보다 트래픽의 편중 현상이 클수록 전체적으로 우수한 처리율/지연시간 성능을 보인다. 따라서, 통신망이 트래픽의 편중 현상을 잘 반영하도록 분할되어 개층적으로 구성된다면, 개별 사용자노드의 입장에서도 전체 통신망의 입장에서도 H-SDA는 효과적인 알고리즘임을 알 수 있다.

V. 결 론

TDM 방식의 다중채널 통신망에서는 TDM 싸이클 주기가 성능을 결정짓는 중요한 인수이다. 본 논문에서는 노드간의 트래픽 흐름의 편중화 현상을 반영하여 TDM 프레임 구조를 구성함으로써 TDM 싸이클 주기를 줄이는 방안을 제안하였다(H-SDA).

H-SDA에서 각 사용자노드들은 그들간의 트래픽 편중화 정도에 따라 몇 개의 군으로 분할된다. 같은 군에 속한 노드사이에는 한 홉만에 통신하도록 하였으며 다른 군에 속한 노드사이의 통신은 각 군마다 하나의 게이트웨이를 두어 3 홉에 통신할 수 있도록 하였다. 결과적으로 H-SDA에서 트래픽 편중화 현상이 심한 사용자노드들 간의 통신을 위해서는 더 많은 타임슬롯이 할당될 수 있도록 하며, 그렇지 않은 사용자노드들 간의 통신을 위해 상대적으로 적은양의 타임슬롯을 할당되도록 하였다. 따라서 서로 통신량이 많은 노드 간에 전송기회를 더 많이 부여함으로써 처리율과 지연시간을 향상시킬 수 있으며, 또한, 기존의 SDA에서 이용율이 낮았던 타임슬롯의 이용율도 개선시킬 수 있다.

또한, H-SDA의 TDM 프레임 설계시 저부하에서의 최소지연시간과 고부하에서의 최대처리율 관점에서 최적인 통신망 분할에 관해 분석하였다. 일반적으로 군의 수가 $g < \sqrt[3]{N^2/2}$ 의 범위에서 증가할 수록 저부하에서의 지연시간은 점점 짧아지는 반면 망의 처리율은 점점 감소한다. 제안된 H-SDA의 성능 평가를 위해 Embedded Markov chain에 기반한 해석적

모델을 개발하였으며, 그 타당성은 시뮬레이션에 의해 검증하였다.

H-SDA에서 내부패킷은 1홉만에 전송될 수 있는 반면, 외부패킷은 3홉을 기치므로 H-SDA를 LAN/MAN 환경에 적용할 경우, 전파지연으로 인한 지연시간이 커진다. 이 문제를 완화할 수 있는 하나의 방안으로서는 그림 10의 중앙집중형 WDM star 구조를 생각해볼 수 있다 [7].

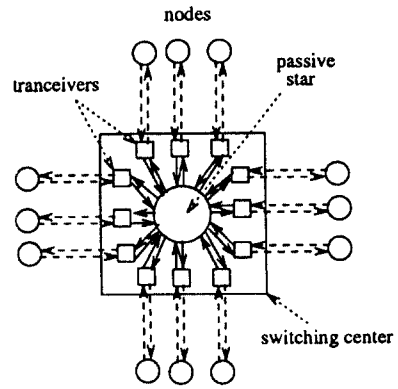


그림 10. 중앙집중형 WDM star
Figure 10. Centralized WDM star

송/수신 인터페이스(transceiver)를 각 노드로부터 분리하여 중앙의 star 부근에 위치시키고 별도의 선으로 송/수신 인터페이스와 노드를 연결한다. 송/수신 인터페이스는 노드와 star사이의 접속 및 경로 배정 기능을 한다. 이 구조에서 외부패킷은 노드를 공유하지 않고 중앙의 송/수신 인터페이스만 공유하므로 전파지연으로 인한 지연시간을 줄일 수 있다. 또한, 이 구조는 star와 송/수신 인터페이스 간의 거리가 짧으므로 TDM 프레임 동기(synchronization)와 슬롯 동기를 유지하기 쉬운 장점이 있으므로 앞으로 더 연구가 되어야 할 구조이다.

또한, TDM 방식의 통신망에서 타임슬롯을 동적으로 활용하는 것도 앞으로 연구되어야 할 분야인데 H-SDA에서는 두가지 방식으로 가능하다. 첫째는 [9]에서처럼 군 내부통신에서 다른 사용자노드가 사용하지 않는 타임슬롯을 잠시 빌려 여러 홉을 거쳐 최종목적지에 도달하게 함으로써 망의 성능을 향상시키는 것이다. 둘째는 자신의 게이트웨이를 이용하는 방법이다. 예를 들어 표 2에서 $n_{1,1}$ 이 $n_{1,2}$ 에게 내

부패킷을 전송할 때, 먼저 패킷을 G_1 으로 전송하고 G_1 은 그 패킷을 $n_{1,2}$ 에게 중계함으로써 한 내부통신의 성능을 간접적으로 향상시킬 수 있다.

현재 노드간의 흐름이 균질하지 않은 특성을 갖는 트래픽 행렬이 주어졌을 때 망을 최적으로 분할하는 heuristic 알고리즘과 TDM 방식의 다중채널 통신망에서 송수신 인터페이스의 수가 여러 개인 때의 최적 프레임구조 및 동작 채널 이용방안에 대해 연구중에 있다.

참 고 문 헌

1. Biswanth Mukherjee, "WDM-based local light-wave networks, part I: single-hop systems," *IEEE Network Mag.*, vol.6, no.3, pp.12-27, May 1992.
2. Biswanth Mukherjee, "WDM-based local light-wave networks, part II: multihop systems," *IEEE Network Mag.*, vol.16, no.4, pp.20-32, July 1992.
3. G. Hill, "Wavelength domain optical network techniques," *Proc. IEEE*, vol.78, no.1, pp.121-132, Jan. 1990.
4. Charles A. Bracket, "Dense wavelength division multiplexing networks: principles and applications," *IEEE J. Select. Areas Commun.*, vol. SAC-8, no.6, pp.948-964, Aug, 1990.
5. I. Chlamtac and A. Ganz, "Channel allocation protocols in frequency-time controlled high speed networks," *IEEE Trans. Commun.*, vol.COM 36, no.4, pp.430-440, Apr. 1988.
6. I. Chlamtac and A. Ganz, "A multibus train communication (AMTRAC) architecture for high speed fiber optic networks," *IEEE Trans. Commun.*, vol.COM-36, no.6, pp.903-912, July 1988.
7. S. Bhattacharya, D. H. C. Du, A. Pavan, S. R. Tong, R. Vetter and K. Williams, "A network architecture for distributed performance heterogeneous computing," *Technical Report, U. of Minnesota*, 1992, Submitted to IEEE Computer.
8. K. M. Sivalingam, K. Bogneni and P. W. Dowd, "Pre-allocation media access control protocols for multiple access WDM photonic networks," *ACM SIGCOMM'92*, pp.235-246, 1992.
9. I. Chlamtac and A. Ganz, "Toward alternative high speed network concepts: the SWIFT architecture," *IEEE Trans. Commun.*, vol.COM-38, no.4, pp.431-439, Apr. 1990.
10. Mark J. Lorence and M. Satyanarayanan, "IPwatch: A tool for monitoring network locality," *Operating Systems Review*, vol.24, no.1, pp.58-80, Jan. 1990.
11. Will E. Leland and Daniel V. Wilson, "High time resolution measurement and analysis of LAN traffic: implications for LAN interconnection," *Proc. IEEE INFOCOM'91*, pp.1360-1366, April 1991.
12. Sheldon M. Ross, *Stochastic Process*. John Wiley & Sons, 1983.
13. I. Rubin and Z. Zhang, "Message delay analysis for TDMA schemes using contiguous-slot assignments," *IEEE Trans. Commun.*, vol. 40, pp.730-737, April 1992.
14. T. E. Stern, "Packet scheduling protocols in multi beam communication satellite," *Proc. Int. Symp. Inform. Theory*, June 1979.
15. L. Kleinrock, *Queuing Systems*. John Wiley & Sons, 1975.
16. M. H. MacDougall, *Simulating Computer Systems*. The MIT Press, 1987.
17. Seonuck Paek, Yanghee Choi, Chongsang Kim, "Channel allocation based on traffic locality in multichannel high-speed network," *Seoul National Univ. Technical Report 93-1, Dept. of Computer Eng.*, 1993.



白 善 旭(Seonuck Paek) 정회원
 1986년 : 서울대학교 컴퓨터공학과 학사
 1989년 : 서울대학교 컴퓨터공학과 석사
 1989년 ~ 현재 : 서울대학교 컴퓨터 공학과 박사과정



崔 陽 熙(Yanghee Choi) 종신회원
 1955년 7월 27일생
 1975년 2월 : 서울대학교 공과대학 전자공학과(학사)
 1977년 2월 : 한국과학원 전기 및 전자공학과(석사)
 1984년 6월 : 프랑스 국립 전기통신 대학 전산과(공학박사)

1977년~1979년 : 한국전기통신연구소 연구원
 1981년~1984년 : 프랑스 국립 전기통신연구소 연구원
 1988년~1989년 : IBM Thomas J. Watson Research Center 방문연구원
 1984년~1991년 : 한국전자통신연구소 책임연구원
 1991년~ 현재 : 서울대학교 컴퓨터공학과 조교수
 서울대학교 중앙교육연구전산원 부원장



金 宗 相(Chong Sang Kim) 정회원
 1960년 : 서울대학교 공과대학 전자공학과 학사
 1965년 : 서울대학교 공과대학 전자공학과 석사
 1975년 : 서울대학교 공과대학 전자공학과 박사
 1979년 ~ 현재 : 서울대학교 공과대학 컴퓨터공학과 교수

1986년~1988년 : 한국정보과학회 회장
 1992년~ 현재 : 서울대학교 컴퓨터신기술연구소 소장