

## 분리된 예약채널에 의한 요구할당 다중접근방법에 관한 연구

正會員 韓 正 允\* 正會員 宋 周 錫\*

### A Study on Demand Assignment Multiple Access Algorithm based on Separated Reservation Channel

Jeong Hang Han\*, Joo Seok Song\* *Regular Members*

#### 要 約

본 연구에서는 전송프레임에 대한 예약과 데이터전송이 순차적으로 반복되는 기존의 다중접근방식에서의 채널효율의 저하와 지연특성의 불안정성을 개선하기 위해 위상채널 구조에서 예약 채널과 데이터전송 채널을 논리적으로 분리하여 서로 독립적으로 운영되도록 구성하고, 그 위에서 요구할당 다중접근 알고리즘을 설계했다. 이것은 예약과 데이터 전송을 각각 서로에 대해 독립적으로 수행하게 하고, 데이터전송 채널에서의 프레임 구조를 제거하여 예약을 슬롯단위로 수행함으로써 일정한 수준의 즉 예약 채널 용량-고정적인 대역폭 낭비를 감소하면서 데이터 전송 채널의 효율을 극대화시키고, 각 지구국에서의 트래픽 변동을 보다 효과적으로 수용하며, 안정된 전송지연 특성과 공평성을 획득함으로써 시스템의 전체적인 성능을 향상시키는데 목적이 있다.

#### ABSTRACT

In this study, we have several objectives. First, in spite of a fixed bandwidth waste of reservation channel, we maximize the efficiency of transmission channel by the multiple access algorithm that performs channel reservation and data transmission independently, eliminating the frame structure of transmission channel and reserving it by the slot unit. Second, in order to improve the entire system performance, we accommodate the variation of traffic at each earth station more effectively, and accomplish the stable delay characteristics and the equity of service for users. For this purpose, we design the satellite channel that consists of reservation channel and transmission channel which are logically separated and operate independently. We also design a demand assignment multiple access algorithm based on the satellite channel structure.

#### I. 서 론

\*延世大學校 電算科學科  
Dept. of Computer Science, Yonsei University.  
論文番號 : 92-70(接受1992. 1. 17)

위성통신기술의 중요 응용분야 가운데 하나로 패킷위성통신망을 들 수 있다. 즉 지역적으로 넓게 분산되어 있고, 각각의 정보통신요구가 다양한 다수의

사용자들을 연결하여 서로 데이터를 교환할 수 있는 데이터교환망으로서의 위성망을 말한다<sup>(1)(2)</sup>. 기존의 지상망기술을 이용한 광역 데이터교환망의 구현은 구축과 유지, 보수, 난점과 운영의 복잡성, 그리고 급격하게 변화하는 사용자의 정보통신 요구사항의 수용의 어려움등을 내포한다. 이에 따라 위성통신기술을 이용한 데이터교환망의 구축은 보다 빠른시간 내에 광범위한 지역을 수용할 수 있는 효과적인 방안으로서 우리나라에서도 많은 관심을 받고 있다<sup>(3)</sup>. 이러한 위성통신시스템의 특징으로는 첫째로, 방송(broadcasting) 시스템이다. 위성을 통해 지상으로 보내지는 전파는 지역적으로 대단히 넓은 수신범위를 갖는다. 둘째, 큰 전파지연시간을 갖는다. 전파의 전송지연 시간은 위성체의 고도에 의해 결정되는데 약 250ms의 일정한 전송지연시간을 갖으며, 이것은 근거리 지상망(LAN)이나 광섬유를 이용한 고속통신망 등에 비해 매우 큰 지연시간이다<sup>(1)(4)</sup>. 셋째, 위성망에서의 통신매체는 다수의 지구국에 의해 공유되는 일정한 주파수 대역(frequency band)이다. 주파수 대역은 지상망에서 사용되는 동축 케이블이나 광섬유와 같이 무한정으로 생산, 설치할 수 있는 것이 아니기 때문에 채널의 이용이 특히 효율적이어야만 한다. 이렇게 제한된 주파수 대역으로 형성되는 위성 채널을 망 내의 여러 지구국이 효율적으로 공유할 수 있도록 하는 것이 다중접근 프로토콜이다. 위성망에서 사용되는 다중접근 프로토콜은 위성통신의 특성 때문에 기존의 근거리 통신망에서 사용되고 있는 CSMA/CD, Token passing 등을 그대로 사용할 수는 없다<sup>(5)(6)</sup>. 일반적으로 위성 통신에 사용되는 다중접근 프로토콜은 주로 주파수분할 방식(Frequency Division Multiple Access, FDMA), 시분할 방식(Time Division Multiple Access, TDMA), 코드분할 방식(Code Division Multiple Access CDMA), 등이 사용되며, 데이터 전송용 위성망에서는 주로 TDMA 방식이 이용된다<sup>(4)(6)(7)</sup>. TDMA 방식은 크게 고정할당(Fixed Assignment), 임의할당(Random Access), 요구할당(Demand Access) 방식으로 분류할 수 있으며, 망의 전송 트래픽 특성에 따라 적합한 방식을 지원한다<sup>(3)(6)</sup>.

본 연구에서 설계하고자 하는 다중접근방법은 다수의 소규모 지구국들에 의해 구성되는 위성망보다는 소수의(고정된) 지구국에 의해 지역적으로 분산된 LAN, MAN 등의 지상망을 연결하여 대규모의 WAN을 형성하는 Backbone 망으로서의 위성망을

위한 것이다. 이러한 Backbone 망은 일반적으로 망의 트래픽 요구가 안정되어 쉽게 변하지 않고, 부하가 비교적 높은 수준에서 유지되고, 지역망(local access network)보다는 큰 전송지연을 허용하고 있다는 특징을 지닌다<sup>(5)</sup>. 본 연구에서는 특히 중간 이상의 트래픽 부하에서의 효율적인 채널이용과 안정된 전송지연, 그리고 각 지구국의 트래픽에 대한 서비스의 공평성을 제공할 수 있는 프로토콜의 설계에 목적을 둔다. 이를 위해 본 연구에서는 예약 채널과 데이터 전송 채널을 완전히 분리하여 서로 독립적으로 운영되도록 구성한다. 이것은 예약과 데이터 전송을 각각 서로에 대해 비동기적으로(asynchronous) 수행하게 함으로써 일정한 수준의-즉 예약 채널 용량-고정적인 대역폭의 낭비를 감수하고라도 데이터 전송 채널의 효율을 극대화시키고, 각 지구국에서의 트래픽 변동을 보다 효과적으로 수용하며, 안정된 전송지연 특성을 획득함으로써 시스템의 전체적인 성능을 향상시키고자 함이다.

## II. 예약채널과 전송채널의 분리

### 2.1 기존의 채널구조와 다중접근 방식의 문제점

현재까지 개발되어 주로 이용되고 있는 위성채널에 대한 요구할당 다중접근 프로토콜은 물리적으로는 단일한 채널(동일한 주파수 대역)을 예약과 데이터전송을 위해 시간적으로 분리하여, 예약을 위한 대역폭과 데이터전송을 위한 대역폭으로 나누어 상호간의 동기에 의해 사용한다<sup>(5)(8)</sup>. 이것을 그림으로 나타내면 다음과 같다.

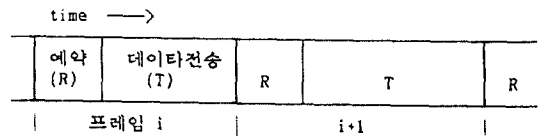


그림 1. 기존 위성채널의 구조

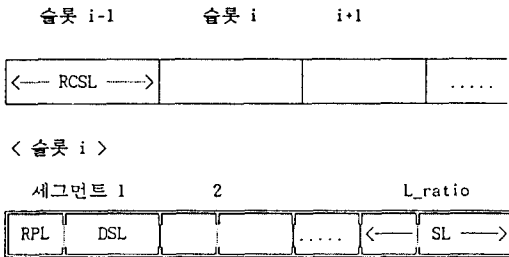
Fig 1. Structure of traditional satellite channels

이와 같은 방식에서는 예약과 데이터전송이 순차적으로 반복되어지며, 일반적으로 하나의 예약주기에 있어서 하나의 데이터전송주기(프레임)에 대한 예약이 수행된다. 예약은 각 지구국이 예약주기의 예약슬롯을 고정적으로 할당받거나 경쟁(contention)에 의해 할당받는 형태로 이루어진다. 일반적으로 하나의 최

소 데이터전송주기는 단일 위성전송지연(round trip delay)보다 길게 설정되고, 크기가 고정되거나 혹은 가변적일 수 있다<sup>(5)(8)</sup>. 따라서 프레임 i에서의 예약주기에서의 예약요청은 프레임 i+1의 데이터 전송을 예약하게 된다. 이러한 예약요청들은 프레임 i-1의 데이터전송 주기와 프레임 i의 예약주기 동안에 각 지구국에 도착한 패킷들과 프레임 i-1의 예약주기에서 예약요청한 것들 중에서 예약되지 못하여 전송이 지연된 패킷, 그리고 전송 오류에 의해 재전송되는 패킷들에 대한 것이다. 이러한 채널구조와 예약방식에서는 프레임단위로 예약을 수행하기 때문에 프레임내에 사용되지 못하고 낭비되는 슬롯이 발생하거나, 각 프레임의 전송주기 동안에 도착하는 패킷들이 다음 프레임의 예약주기가 돌아올 때 까지 기다리게 되며(특히 가변길이 프레임일 경우 프레임내의 낭비되는 슬롯은 줄지만 지연시간의 변화가 심하다), 예약슬롯의 할당 방식에 따라서는 부하에 따라 이러한 지연이 매우 커질 수도 있다. 이것은 예약과 데이터 전송이 서로 배타적으로 반복되는 채널구조상의 조건과 예약이 프레임 단위로 수행되는 조건에 의한 것이라 할 수 있다.

2.2 예약채널의 분리

각 지구국이 본 다중접근 알고리즘에 의해 공유하는 실제 위성채널은 다음과 같은 구조를 가진다<sup>(1)(9)</sup>.



- 총 가용 채널대역폭 : GB (bps)
- TCB (Transmission Channel Bandwidth) (bps)
- RCB (Reservation Channel Bandwidth) (bps)
- RCSL(Real Channel Slot Length) (bits)
- RPL (Reservation Packet Length) (bits)
- DPL (Data Packet Length) (bits)
- DSL (Data Segment Length) (bits)
- SL (Segment Length) (bits)

그림 2. 본 알고리즘에서 이용하는 채널의 물리적 구조  
Fig 2. Physical structure of channel used in this algorithm

한편, L\_ratio는 다음과 같이 구해진다.

$$L\_ratio = (\text{예약채널 대역폭} / \text{전송채널 대역폭}) * (\text{데이터패킷 크기} / \text{예약패킷 크기})$$

$$= (DPL / RSL) * (RCB / TCB)$$

$$RCSL = (L\_ratio * RSL) + DPL$$

$$GB = TCB + RCB$$

$$TCB / RCB = DPL / (L\_ratio * RSL) \quad \dots\dots\dots \odot$$

실제 위성채널에서 하나의 슬롯은 L\_ratio개의 동일한 크기의 세그먼트로 분할되며, 각 세그먼트마다 하나씩의 예약슬롯이 존재한다. 따라서 하나의 위성채널슬롯은 L\_ratio 개의 예약패킷과 하나의 데이터패킷을 전송할 수 있는 크기로 고정된다. 따라서 하나의 데이터패킷을 송신 또는 수신할 때 걸리는 시간과 L\_ratio 개의 예약패킷이 송수신되는데 걸리는 시간이 위성채널의 한슬롯시간, RCSL/GB(sec)로 동일하다. 위성 채널에서 총 가용대역폭(GB)는 예약을 위한 예약 대역폭(RCB)과 데이터전송을 위한 데이터전송 대역폭(TCB)으로 이루어진다. 위성채널 상의 데이터패킷 전송과 예약패킷의 전송은 서로 독자적인 메카니즘에 의해 수행된다. 따라서 위의 채널구조는 서로 다른 주파수대역에서 각각 TCB와 RCB의 대역폭을 가지고 형성되어 서로 독립적으로 운영되는 데이터전송채널과 예약채널의 구조에 논리적으로 대응될 수 있다. 즉,

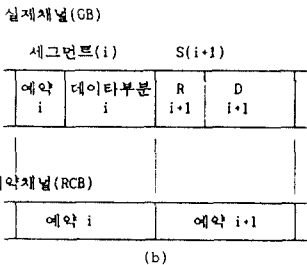
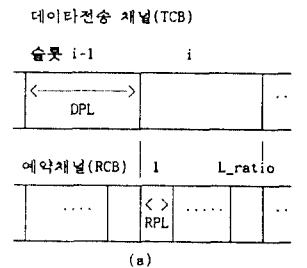


그림 3. 채널의 논리적 구조와 예약슬롯의 대응  
Fig 3. Logical structure of channel and mapping of reservation slots

위의 데이터전송채널 예약채널은 완전히 독립적으로 데이터패킷과 예약패킷을 동시에 전송, 수신한다. 위의 논리적으로 분리된 두 채널이 실제 위성채널과 일치하기 위해서는

① 데이터전송채널에서의 데이터패킷 전송시간(DPL/TCB)과 실제 위성채널에서의 데이터패킷 전송시간(RCSL/GB)이 같아야 한다.

$$\Rightarrow \text{㉠에서 } GB = TCB + (TCB * (L\_ratio * RPL) / DPL) \\ = (TCB * RSCL) / DPL \text{ 이고}$$

$$RCSL / GB = DPL / TCB \text{ 이다.}$$

② L\_ratio 개의 예약패킷이 전송되는데 걸리는 시간이 예약채널에서(L\_ratio\*RPL/RCB)와 실제 위성채널에서(RSCL/GB) 같아야 한다.

$$\Rightarrow \text{㉡에서 } GB = RCB + ((DPL * RCB) / (L\_ratio * RPL)) \\ = (RCB * RCSL) / (L\_ratio * RPL) \text{ 이고}$$

$$RCSL / GB = (L\_ratio * RPL) / RCB \text{ 이다.}$$

조건 ②는 예약채널의 예약패킷 송수신 시간이 실제 위성채널의 한 세그먼트(SL)의 전송시간과 동일하다는 것을 나타낸다. 그림 3(b)에서 볼 수 있듯이 실제채널에서 예약 i가 지난후 세그먼트 i의 나머지 시간동안 시스템에 도착한 데이터 패킷들에 대한 예약시도는 최소한 세그먼트 i+1의 예약슬롯이 시작될 때 까지는 기다려야 된다. 이렇게 실제채널에서 기다리는 시간은 작은 대역폭(RCB)을 사용하는 논리적 예약채널에서의 예약슬롯 i의 전송시간이 길다는 것에 의해 흡수되어 나타난다. 따라서 위의 구조를 가지는 예약채널에서의 예약슬롯의 할당과 운영이 실제 위성채널에서의 그것과 동일하다.

### III. 분리된 예약채널을 이용한 다중접근 방법

#### 3.1 가정하는 위성망 시스템의 개요

본 연구에서 가정하는 위성망은 비교적 소수의 지구국이 대용량의 단일 uplink와 단일 downlink를 위성체를 통해 공유하여 지상 지역망을 연결하는 SWAN의 근간망(backbone)으로서의 위성망이다. 망 내의 각 지구국에서 위성체를 통한 통신 지연시간은 동일(약 250ms)하고, 위성체의 미세한 궤도변화와 위성채널에 대한 동기(synchronization)는 시스-

템 자체에서 제공된다고 가정한다. 위성망에는 중앙 제어 지구국이 존재하지 않고, 위성체는 특별한 자료 처리기능을 갖지 않고, 다만 uplink를 통해 수신되는 신호를 downlink의 주파수로 변환하여 필요한 증폭 과정을 거쳐 모든 지구국으로 방송한다. 각각의 지구국은 고유의 주소를 부여받으며, 또한 각각의 사용자들로부터 도착하는 다양한 크기의 메시지들을 고정된 크기의 패킷으로 분할해준다.

#### 3.2 위성채널의 구조 및 패킷의 구조

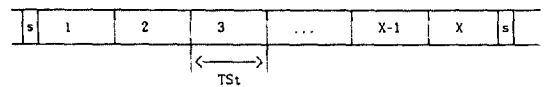
##### 3.2.1 예약채널(Reservation Channel)

망 내의 모든 지구국이 고정할당(Fixed Assignment) TDMA 방식에 의해 다중접근하는 채널로 위성망 시스템의 총 가용 대역폭(bandwidth) 중에서 고정적으로 낭비되는 부분이다. 한 예약주기(reservation cycle)는 망 내에 존재하는 지구국의 갯수(N)과 망 제어에 필요한 슬롯 하나를 합쳐(N+1)개의, 일정한 크기의 예약슬롯(reservation slot)으로 구성된다. 각각의 지구국은 한 예약주기 내에 한번씩의 예약패킷 전송기회를 갖는다. 각각의 예약슬롯은 하나의 예약 혹은 제어 패킷을 전송할 수 있는 크기로 고정된다.

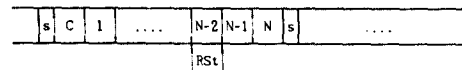
##### 3.2.2 데이터전송채널(Data Transmission Channel)

망 내의 지구국들이 데이터 패킷을 전송하는 데이터전송 전용의 채널이며, 연속된 전송슬롯(transmission slot)으로 구성된다. 각각의 전송슬롯은 하나의 데이터 패킷을 전송하는데 필요한 크기로 고정

‘데이터전송채널.’



‘예약 채널.’



- s : 동기 시간(SEC)
- C : 제어슬롯
- TSt : 전송슬롯(SEC)
- TMs : 데이터 패킷 크기(bit)
- RMs : 예약 패킷 크기(bit)
- Wf = RCB / GB : 고정 낭비 비율
- L\_ratio = (RCB/TCB) \* (TMs/RMs) : 전송슬롯에 대응되는 예약슬롯의 갯수
- GB : 총 가용 대역폭(bps)
- TCB : 데이터전송채널 대역폭(bps)
- RCB : 예약채널 대역폭(bps)
- RSt : 예약슬롯(SEC)

그림 4. 예약채널과 데이터전송채널의 구조  
Fig 4. Structure of Reservation and Data Transmission Channels.

된다. 각 지구국은 예약채널을 이용한 예약체계에 의해 데이터전송채널에서 특정 전송슬롯을 할당받고, 그 전송슬롯에서 자신의 데이터패킷을 전송한다.

### 3.2.3 패킷 구조

위성망 내에서 교환되는 패킷은 데이터패킷, 예약패킷, 제어패킷의 세가지로 구분되며, 각각 일정한 크기로 고정되어 있다. 하나의 데이터패킷은 데이터 전송채널에서 하나의 전송슬롯에 해당한다. 예약 및 제어 패킷은 데이터패킷의 전송과는 독립적으로 예약채널을 통해 전송되며, 데이터패킷에 비해 매우 작은 크기를 갖는다. 망내의 각 지구국의 매 예약슬롯마다 예약채널로부터 패킷을 송,수신하여 데이터 전송채널의 전송슬롯을 예약하거나 지구국의 추가, 삭제 등의 망 제어기능을 수행한다. 본 연구에서는 각 패킷의 세부적인 구조를 설계하기보다는, 다중접근 프로토콜 자체의 운영과 성능에 영향을 주는 항목들만을 개념적으로 구분하여 파악한다.

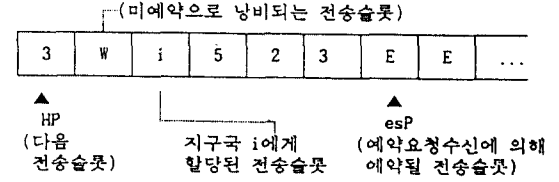
### 3.3 예약벡터와 전송버퍼의 구조와 운영

망 내의 모든 지구국은 동일한 구조의 예약벡터와 전송버퍼를 유지하며 자신의 예약을 통해 전송슬롯을 할당받고 데이터전송에 참여한다. 전송버퍼는 망 사용자로부터 각 지구국으로 전달되는 데이터 패킷을 전송슬롯을 할당받아 전송을 수행할 때 까지 저장한다. 예약벡터는 데이터전송채널의 각 전송슬롯에 대한 예약상황을 나타내며, 모든 지구국에서 일관된 방식으로 동일하게 유지되어야 한다. 예약벡터의 각 항목은 하나의 전송슬롯에 대응하고, 각 지구국에서는 예약벡터에서 자기 자신에게 할당된 전송슬롯에 패킷을 전송한다.

본 연구에서는 예약벡터와 전송버퍼의 크기가 무한하다고 가정한다. 예약벡터와 전송버퍼는 모두 FIFO로 구현한다. 예약벡터는 해당 전송슬롯을 할당받은 지구국의 주소나 비예약으로 전송되는 슬롯상태(W), 아직 예약이 가능한 슬롯(E) 등의 세가지 상태를 표시한다. 각 지구국에서는 하나의 전송슬롯 시간이 지날 때 마다 예약벡터의 포인터 HP를 규칙적으로 갱신한다. 예약벡터의 esP는 예약요청에 의해 각 지구국에 할당될 수 있는 가장 가까운 전송슬롯을 나타내며, 예약채널을 통해 수신되는 지구국으로부터의 예약요청에 의해 갱신된다.

예약채널에 대한 지구국의 접근은 고정할당방식을

#### < 예약벡터 >



#### < 전송버퍼 >

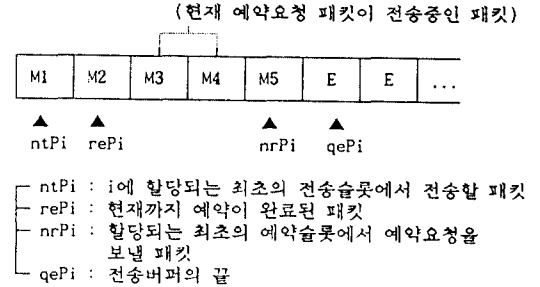


그림 5. 지구국 i의 예약벡터와 전송버퍼

Fig. 5. Reservation Vector and Transmission Buffer at Earth Station i

사용하기 때문에 최악의 부하(각 예약슬롯마다 하나씩의 예약요청 발생)가 시스템에 발생한다고 해도 각 지구국은 최소한 N 개의 전송슬롯마다 하나씩의 패킷을 전송할 수 있다<sup>10)</sup>. 다른 극단적인 예로서 망내에서 지구국 i만이 전송할 패킷을 가지고 있을 때는 최대한 (N+1)/L\_ratio 개의 전송슬롯마다 하나씩의 전송슬롯을 확보할 수 있다. 예약벡터에서는 수신을 기준으로 FIFO로 예약하기 때문에 전송슬롯 HP에서 일정한 갯수(X개의 전송슬롯) 이후의 전송슬롯에 대한 예약은 일종의 압축예약 효과를 나타낸다. 각 지구국은 최소한 (N+1)/L\_ratio 개(즉, 한 예약주기)에서 최대한 N개의 전송슬롯마다 하나씩의 전송슬롯을 할당받는 것이 정상이고, HP가 하나의 전송슬롯마다 갱신되는 동안 L\_ratio 개의 예약슬롯이 수신된다. 그러나 이 때 예약벡터에 이미 어느정도(X개) 이상의 예약이 되었다고 생각해 보자. 전체 시스템의 평균 부하가 1(packet/slot time) 이하이면 L\_ratio 개의 예약슬롯이 지나갈 동안에 하나 이하의 예약요구가 수신되기 때문에 수신된 예약요청패킷을 전송했던 지구국은 예약주기의 크기에 관계없이 esP가 지칭하는 전송슬롯을 할당받을 수 있다. 이것은 예약이 고정된 프레임단위로 수행되지 않고, 전송슬롯 단위로 수행되기 때문이다. 이에 따라 낭비되는

전송슬롯을 줄임으로써 예약벡터의 크기가 작아지고 각 데이터패킷의 버퍼지연이 감소하며, 시스템의 성능이 향상된다. 이러한 압축효과는 HP와 esP간의 슬롯거리가 클수록 커진다.

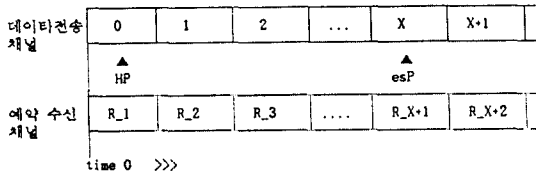


그림 6. 슬롯단위 예약과 채널이용  
Fig 6. Reservation for slot unit and channel usage

HP와 esP사이의 거리 X가 1 전송슬롯이면 예약 구간 R<sub>1</sub>의 예약슬롯 중 하나의 예약 요청만 있으면 esP가 지정하는 전송슬롯이 낭비되지 않는다. X가 2 이면 R<sub>1</sub>이나 R<sub>2</sub>동안의 예약슬롯 중 하나의 예약요청만 있으면 된다. 일반적으로 거리가 X 전송슬롯이면 HP를 포함하여 X개의 전송슬롯에 대응되는 예약 슬롯 가운데 하나의 예약만이라도 수행되면 esP의 전송슬롯은 낭비되지 않는다. 즉, 거리 X가 (N+1) / L\_ratio 이상이면 한 예약주기 내에 하나라도 예약이 있으면 전송슬롯을 낭비하지 않게 된다. 따라서 최대한 L\_ratio를 크게 하는 것이 바람직하지만, 여기에는 고정적인 낭비인 예약채널의 대역폭과의 tradeoff가 있다<sup>(8)(9)</sup>.

### 3.4 설계된 다중접근 알고리즘

본 연구에서 제안하는 다중접근 알고리즘은 망 내의 모든 지구국에서 동일하게 수행되는 분산수행 알고리즘<sup>(2)(9)</sup>이다. 본 알고리즘의 주요 고려사항은 데이터패킷의 도착여부, 예약슬롯의 할당 및 예약요청패킷의 전송, 예약요청의 수신여부, 예약벡터와 버퍼의 유지, 전송슬롯의 할당과 데이터패킷의 전송 등이며 트래픽제어와 지구국의 추가와 삭제에 관련된 알고리즘을 가정한다.

분산 다중접근 알고리즘의 수행과정을 흐름도로 나타내면 다음과 같다.

각 지구국에서는 데이터전송 및 수신을 위한 프로세스와 패킷의 생성, 예약요청 및 예약수신을 위한 프로세스가 각각 독립적으로 수행된다. 프로세스들은 망의 전체적인 동기화에 의해 데이터전송슬롯 타이밍과 예약슬롯 타이밍을 기준으로 수행된다. 본 다중

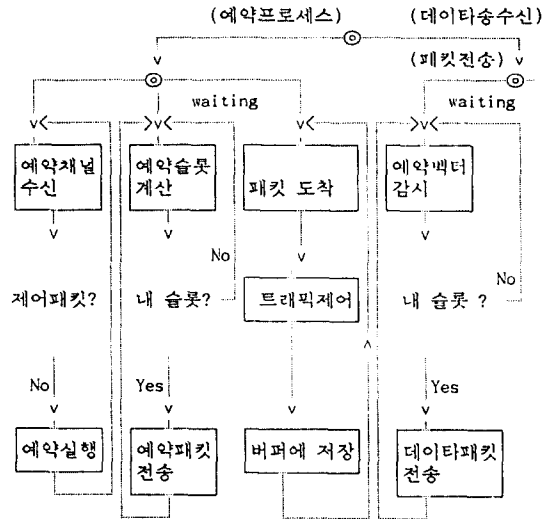


그림 7. 다중접근 알고리즘의 흐름도  
Fig 7. Flow chart for multiple access algorithm

접근 알고리즘에서는 답신체계에 의한 위성지연과 시스템 부하의 증가를 최대한 감소시키기 위해 송신자가 재전송 여부를 결정하는 방법을 가정하며, 위성채널이 매우 높은 신뢰도를 가지기 때문에 오류에 의한 재전송으로 인한 시스템의 성능변화는 고려하지 않는다.

#### <분산 알고리즘의 주요 자료형>

```
예약벡터(:R_vector)=structure
    node id: 지구국의 주소 형;
    next :R_vector의 포인터;
end structure;
```

```
전송버퍼(:T_buffer)=structure
    user_packet: 전송될 패킷 형;
    next :T_buffer의 포인터;
end structure;
```

#### 3.4.1 패킷생성 프로세스

패킷생성 프로세스는 각 사용자들로부터 지구국으로 도착하는 패킷들을 트래픽 제어 알고리즘에 의해 전송버퍼로 저장한다. 본 알고리즘의 데이터패킷지연은 전송버퍼에서의 지연과 예약벡터에서의 지연에 의해 결정된다. 따라서 각 지구국에서는 주어진 지연 시간 조건과 채널효율을 고려하여, 전송버퍼의 크기

(qePi-nrPi)와 예약벡터의 크기 (esP-HP)를 적절하게 유지하도록 패킷 집수를 조절해야한다.

본 알고리즘에서 이러한 트래픽 제어는 각 지구국 의 패킷 생성율에 이미 반영되어 있다고 가정한다.

<트래픽 제어 알고리즘(Traffic Control Algorithm)>

**TCA;**

```
D_packet          : 전송채널 패킷형;
사용자패킷수신(), 패킷거부() : 시스템제공기능;
BEGIN
A : 사용자패킷수신(D_packet) :
    패킷집수판단(R_vector, T_buffer, parameters);
    if(패킷집수=YES)
        than begin
            qePi.user_packet=D_packet;
            qepi=qePi.next;
            endthan;
        else 패킷거부(D_packet);
    goto(A);
END TCA;
```

**3.4.2 예약요청 프로세스**

예약요청 프로세스는 예약채널의 예약전송슬롯 타임을 감시하고 있다가 자신에게 할당된(고정할당) 예약슬롯일 때 예약요청패킷을 전송한다. 이때 전송버퍼의 nrPi를 참조하여 비예약 데이터패킷이 없으면 예약요청을 하지않는다. 예약요청 후에는 전송버퍼의 nrPi를 갱신한다. 망의 제어, 지구국의 추가나 제거에 관련된 제어정보도 예약채널을 이용하여 제어패킷을 통해 전달한다.

<예약 요청 알고리즘(Reservation Request Algorithm)>

**RRA;**

```
(*지구국 i에서의 예약요청*)
ntPi, qePi          : 전송버퍼 포인터;
새로운_예약슬롯(), 제어수행(), 예약채널송신() :
시스템 제공기능;
Rcti      : 지구국 i를 기준으로 예약주기 추적;
R_packet   : 예약채널의 패킷 자료형;
Control_packet   : 제어패킷 형;
BEGIN
A : while (Rcti <> 0) do (*Rcti가 0이면 지구국
    i에게 할당된 예약슬롯*)
    begin
        B : while(새로운_예약슬롯())=FALSE) do
```

```
goto(B);
Rcti=Rcti-1;          (*고정할당*)
endwhile;
(*지구국 i의 예약 요청*)
if(제어수행(i_node_id)=TRUE)
than begin(*지구국 i에서 제어패킷을 발송*)
    R_packet=Control_packet;
    예약채널송신(R_packet);
endthan;
else if (nrPi <> qePi)
    than begin(*전송버퍼에 비예약 패킷이 있음*)
        R_packet=nrPi.user_packet;
        nrPi=nrPi.next;
        예약채널송신(R_packet);
    endthan;
    else goto(C); (*전송버퍼에 비예약 패킷 없음*)
(*지구국 i에서 새로운 예약주기 수행*)
C : Rcti=(N+1);
    goto(A);
END RRA;
```

**3.4.3 예약실행 프로세스**

예약실행 프로세스는 예약채널을 매 예약슬롯단위로 수신하여, 제어패킷일 경우에는 적절한 제어프로세스로 패킷을 보내주고, 예약요청패킷일 경우에는 예약을 수행한다. 예약의 수행은 예약요청패킷 내의 송신자 주소를 예약벡터의 esP에 기록하고 esP를 갱신한다. 이 과정은 망 내의 모든 지구국에서 동일하게 발생하며, 예약을 요청한 지구국에서는 자신의 전송버퍼의 rePi를 갱신한다.

<예약 수행 알고리즘(Reservation Perform Algorithm)>

**RPA;**

```
예약채널수신()      : 시스템제공기능;
esP                  : 예약벡터 포인트;
rePi                 : 전송버퍼 포인터;
BEGIN
A : 예약채널수신(R_packet);
    if(R_packet=Control_packet)
        than Control_machanism(R_packet);(*망 제어기능 호출*)
    else if(R_packet <> EMPTY)
        than begin (*전송채널에 대한 예약을 예약백
```

```

    터에 수행*)
    esP.node_id=R_packet.node_id ;
    esP=esP.next ;
    (*예약요청을 보낸 지구국에서는 자신의 전송버퍼
    에서 예약 종료 포인터를 갱신한다.*)
    if(R_packet.node_id=i_node_id)
        than rePi=rePi.next ;
    endthan ;
    goto(A) ;
END RPA ;

```

### 3.4.4 데이터패킷 전송 프로세스

데이터송수신 프로세스에서는 데이터전송채널에 대한 동기를 유지하며 데이터패킷을 송수신한다. 위 성으로부터 방송되는 데이터패킷을 수신한 지구국은 패킷 내의 목적지 주소를 검사하여 자신에게로 전송된 패킷 만을 접수하고 다른 목적지주소를 갖는 패킷은 버린다. 데이터패킷전송 프로세스에서는 항상 예약벡터를 참조하여 자신에게 할당된 전송슬롯인지를 확인하고 자신의 전송슬롯에서 전송버퍼의 ntPi가 지정하고 있는 데이터 패킷을 전송하고 자신의 전송버퍼의 ntPi를 갱신한다. 데이터패킷전송 프로세스는 예약벡터에서 HP가 자신에게 할당된 전송슬롯이 될 때 까지 매 전송슬롯시간(TSt)마다 HP를 갱신하며 기다린다.

<데이터패킷 전송 알고리즘(Data Packet Transmission Algorithm)>

```

DPTA;      (*지구국 i에서의 패킷전송 알고리즘*)
ctP          : 예약벡터 포인터 ;
ntPi        : 전송버퍼 포인터 ;
새로운_전송슬롯(), 패킷송신() : 시스템 제공기능 ;
BEGIN
A : while (새로운_전송슬롯( )=FALSE) do
    goto(A) ; (*매 전송슬롯 마다 주기적으로 예약
    벡터의 갱신*)
    ctP=ctP.next ;
    if(ctP.npde_id=i_node_id)
        than begin (*지구국 i에 할당된 전송슬롯*)
            패킷송신(ntPi.user_packet) ;
            ntPi=ntPi.next ; (*전송버퍼의 갱신*)
            goto(A) ; (*패킷송신에 한 전송슬롯이 소요
            되므로 예약벡터가 갱신*)
        endthan ;

```

```

    else goto(A) ; (*다른 지구국에 할당된 전송슬롯임*)
END DPTA ;

```

## IV. 성능분석 및 시뮬레이션

### 4.1 전송버퍼 및 예약벡터의 크기

본 알고리즘에서는 각 지구국에서 데이터패킷을 접수한 후에 예약이 이루어질 때까지 전송버퍼에 저장하며, 그 후 전송시까지는 예약벡터에 저장한다고 볼 수 있다. 따라서 각 지구국에서 데이터패킷을 저장하기 위한 시스템 버퍼를 요구하고, 이것에 의한 시스템 비용도 중요한 성능평가요소라고 할 수 있다.

지구국 i에서의 패킷 생성율을  $\lambda TBi$  (packets/sec)이고 한 예약주기를 RC(sec)라 할 때, 전송버퍼의 크기( $qePi-nrPi$ ) 변화는  $(\lambda TBi - (1/RC))$ (sec) 개의 패킷 만큼이다. 한편, 각 전송슬롯에 대한 예약은 매 TSt마다 L\_ratio 개의 예약슬롯이 존재하므로 L\_ratio개의 지구국에서 예약요청을 할 수 있다. 예약벡터의 크기(esP-HP)는 예약 채널에서 수신되는 예약요청패킷의 수가 1(예약요청/TSt) 이상으로 증가하면 예약벡터의 크기가 증가한다. 이것은 예약벡터의 HP가 매 TSt마다 주기적으로 갱신되기 때문이다.

### 4.2 데이터 패킷의 전송지연 특성

각 지구국의 사용자들(지상망)로 부터 지구국으로 도착된 데이터패킷은 트래픽제어 알고리즘(TCA)에 의해 접수 여부가 결정되고, 접수된 데이터패킷은 전송버퍼에 저장된다. 각 지구국은 고정할당(Fixed Assignment) 방식에 의해 예약채널을 접근하고, 자신의 예약슬롯에서 예약패킷을 전송한다. 망 내의 모든 지구국에서 동시에 예약채널에서 예약패킷을 수신하고, 예약벡터를 갱신함으로써 전송슬롯을 예약

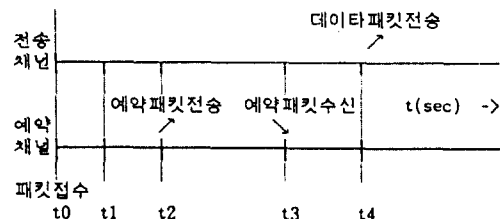


그림 8. 데이터패킷의 전송과정

Fig 8. Transmission process of data packets



하며, 각각에 예약된 전송슬롯에서 데이터패킷을 전송한다.

데이터패킷이 지구국에 접수되어 발송될 때 까지를 그 패킷에 대한 전송지연시간이라고 할 때, 시스템 내에서의 데이터패킷 전송지연시간(W)은  $\langle W = \text{전송버퍼지연}(WTBi) + \text{예약패킷발송시간} + \text{예약패킷 위성지연} + \text{예약벡터지연}(WRV) \rangle$ 이다. 여기서 예약패킷 발송시간( $=1/L\_ratio$ )과 위성지연(약 250 ms)은 시스템에 대해 고정된 상수값으로 일정하다.

한편, 본 다중접근 알고리즘에서의 데이터패킷의 흐름을 큐잉모델로 나타내면 다음과 같다<sup>(8)(11)(12)(13)</sup>.

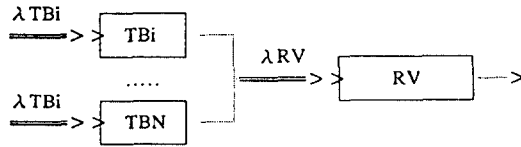


그림 9. 시스템의 큐잉 모델  
Fig 9. Queuing model of system

본 다중접근 알고리즘에 의해 패킷생성율이 λTBi (packets / TSt)인 N개의 전송버퍼(TB)와 패킷생성율이 λRV인 하나의 예약벡터(RV)에 의한 큐잉시스템이 형성된다. 각 전송버퍼는 λTBi의 Poisson 프로세스를 입력으로 하는 M / G / 1 큐로 모델링된다<sup>(5)(8)(11)</sup>.

- R : 패킷도착순간의 평균 잔여서비스시간 =  $XTBi / 2$
- XTBi : 전송버퍼의 평균 서비스시간 =  $(N+1) / L\_ratio$
- NTBi : 전송버퍼 내의 대기 패킷수
- ρTBi : 전송버퍼의 트래픽 밀도 =  $\lambda TBi * XTBi$

따라서 전송버퍼지연은  $\langle WTBi = R + NTBi * XTBi \rangle$ 이고, Little's Theorem에 의해  $\langle NTBi = \lambda TBi * WTBi \rangle$ 이다. 여기서

$$\begin{aligned} WTBi &= R + NTBi * XTBi \\ &= R + (WTBi * \lambda TBi) * XTBi \\ &= (XTBi / 2) + \rho TBi * WTBi \\ &= XTBi / (2 * (1 - \rho TBi)) \end{aligned} \quad \text{이다.}$$

한편, 예약벡터는 시스템내의 N개의 전송버퍼로부터

터의 패킷들이 통합되어 λRV의 Poisson 프로세스에 의해 입력되는 M / G / 1 큐이다<sup>(5)(8)(11)</sup>.

- λRV :  $\sum \lambda TBi$  (i는 1부터 N까지)
- RRV : 패킷도착순간의 평균 잔여서비스시간 =  $XRV / 2$
- XRV : 예약벡터의 평균 서비스시간 =  $1(TSt)$
- NRV : 예약벡터 내의 대기 패킷수
- ρRV : 예약벡터의 트래픽 밀도 =  $\lambda RV * XRV$

따라서 예약벡터지연은  $\langle WRV = RRV + NRV * XRV = 1 / 2 + NRV \rangle$ 이고, Little's Theorem에 의해  $\langle NRV = \lambda RV * WRV \rangle$ 이다. 여기서  $WRV = 1 / 2 + NRV = 1 / 2 + (WRV * \lambda RV) = 1 / (2 * (1 - \lambda RV))$  이다.

위의 결과에 의해 데이터패킷 전송지연시간(W)는  $W = WTBi + (1 / L\_ratio) + DRT + WRV = XTBi / (2 * (1 - \rho TBi)) + 1 / L\_ratio + DRT + 1 / (2 * (1 - \lambda RV))$  에 의해 구해진다.

### 4.3 시뮬레이션 및 결과

본 연구에서는 설계된 다중접근 알고리즘을 이산 사건 시뮬레이션(Discrete Event Simulation)했다. 시뮬레이션에 사용한 프로그래밍 언어는 c이고, SMPLE 등에서 제공하는 라이브러리 함수들을 이용했다<sup>(13)</sup>. 시뮬레이션은 UNIX 환경의 솔본(Solbourne) 워크스테이션에서 수행했으며, 시뮬레이션 시간은 실제 위성망 시스템에서 50,000 전송슬롯(TSt)

- 에 해당한다. 본 시뮬레이션에서 이용한 가정은
  - (1) 지구국의 전송버퍼(TBi)로의 데이터패킷 도착은 λTBi (packets / TSt)의 Poisson 프로세스의 입력분포를 따른다.
  - (2) 각 지구국에서 패킷 생성율은 동일하며, 트래픽의 분포가 시간적으로 균등하다. 즉, 시스템의 총 부하를 G라 하면 λTBi = G / N 이다.
  - (3) 각 지구국에서의 전송버퍼와 예약벡터는 FIFO의 무한 비퍼를 가장한다.
  - (4) 각 지구국에서의 패킷 생성율에는 이미 트래픽 제어 알고리즘에 의한 트래픽 제어가 반영되어 있다.
  - (5) 시스템 내에서의 예약패킷 전송은 위성망의 높은 신뢰도에 의해 오류가 없다.
  - (6) 알고리즘의 수행에 필요한 시스템 동기는 시스템에 의해 제공된다.

⑦ 성능측정시 패킷의 처리 및 내부적인 계산처리 시간은 제외한다.

그림 10은 지구국의 수 N이 64일 때 본 알고리즘에 의한 부하(G) 대 지연(W) 특성을 나타낸 것이다. 여

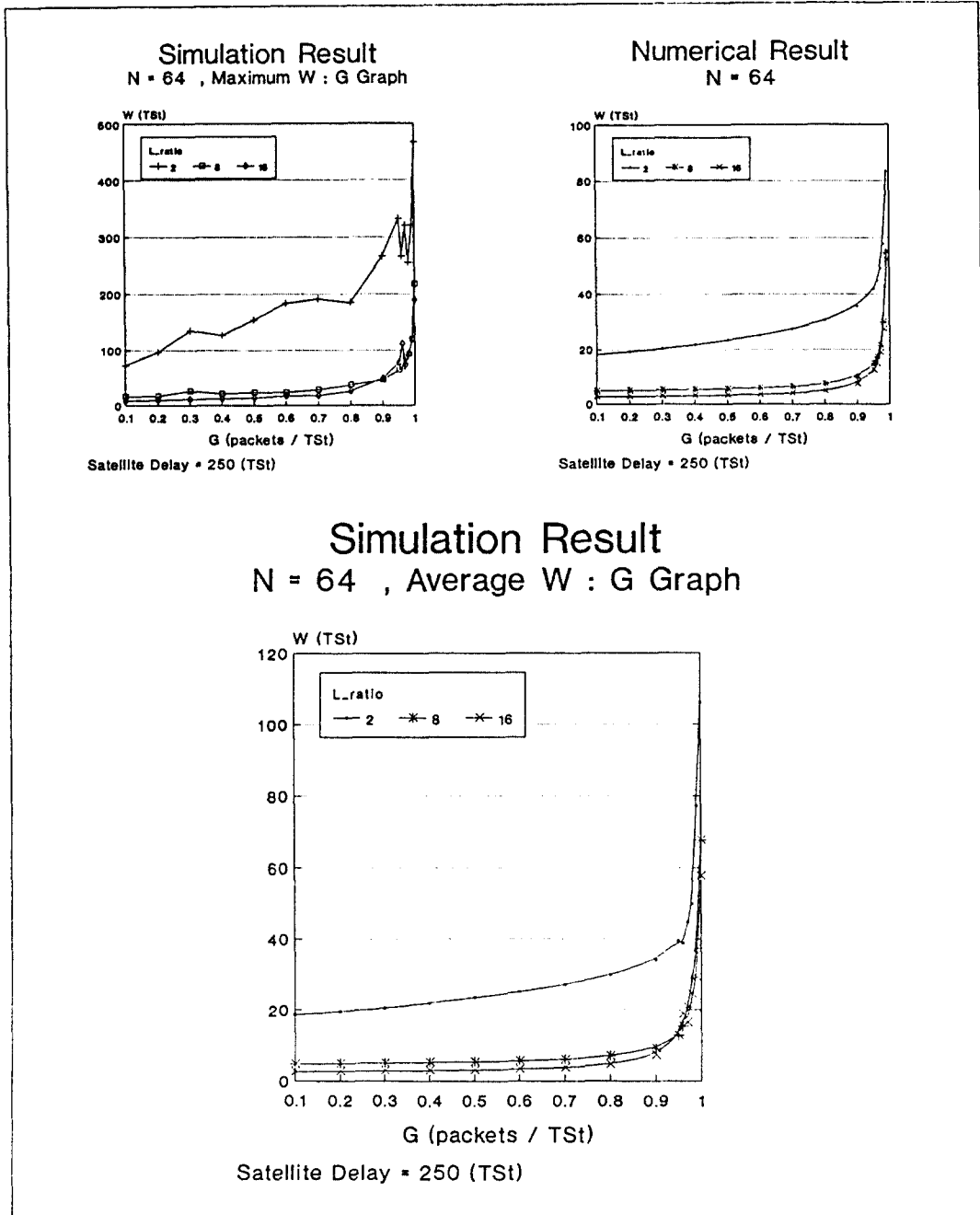


그림 10. 지구국의 수가 64일 때의 부하 대 지연 관계  
Fig 10. Relationship between Load and Delay when N=64

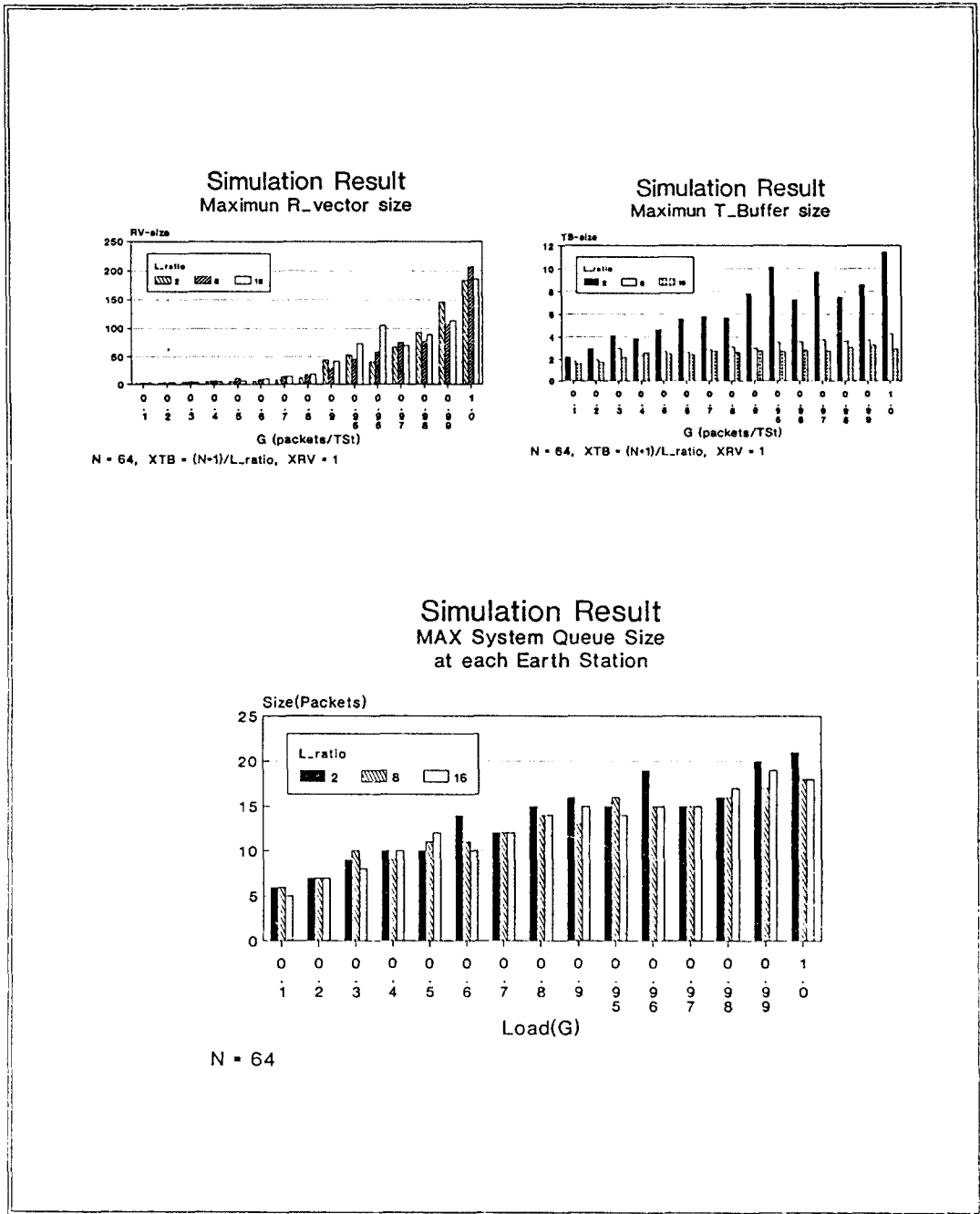


그림 11. 각 지구국 시스템 큐의 최대 크기  
Fig 11. Maximum size of system queues at each station

기서 지연시간의 단위는 전송채널의 전송시간슬롯(TSt)으로 나타내며, 이것은 데이터 패킷의 크기와 전송채널의 대역폭에 의해 결정되는 상대적인 값이다. 그리고 지연시간은 데이터 패킷의 도착에서부터 전송슬롯을 할당받아 전송이 시작될 때까지의 시간이다. 본 알고리즘의 시뮬레이션 결과는 위성지연시간(약 250 ms)이 250개의 전송슬롯에 해당한다고 가정된 것이다.

위의 결과에서, L\_ratio가 클수록 전송지연특성이 좋게 나타난다. 그러나 L\_ratio가 4이상 일 경우 그 결과가 크게 차이가 없고, L\_ratio를 크게 한다는 것은 채널자원에서 순수하게 낭비되는 예약채널의 대역폭을 크게한다는 것이기 때문에 적용 시스템의 지연 요구사항에 적합하도록 선택해야 한다. 전송채널의 효율(throughput)은 고정할당 예약에 의한 요구할당 알고리즘이기 때문에 시스템의 부하(G)와 거의 근사하게 나온다. 따라서 시스템 전체의 채널효율은 다음과 같다.

$$\text{총 효율} = (\text{전송채널 대역폭} / \text{총 시스템 대역폭}) * \text{시스템의 부하}$$

한편, 예약채널에 대한 접근이 고정할당방식에 의하므로 지구국의 수가 늘어남에 따라 전송지연이 커짐을 알 수 있다. 그러나 시스템의 부하가 약 0.96-0.97(packets/slot)에 이를 때 까지 안정된 지연특성을 나타낸다.

그림 11은 본 알고리즘의 수행에 필요한 전송버퍼와 예약벡터의 최대크기 및 각 지구국에서 요구되는 시스템 큐의 최대 크기를 나타낸다. 여기에서 시스템의 부하가 증가할수록 전송버퍼 및 예약벡터에서 대기중인 데이터 패킷이 늘어남을 알 수 있다. 특히 높은 부하에서는 예약벡터에서의 대기 패킷수가 매우 커지는데, 이것은 시스템내의 모든 지구국으로부터 전송채널에 대한 예약이 집중되기 때문이다. 한편 각 지구국에서는 패킷이 접수된 후부터 전송될 때까지 각자의 시스템큐에 실제로 저장하고 있어야 한다. 그림에서와 같이, 본 알고리즘에서는 각각의 지구국이 최소한 갖추고 있어야 하는 저장 능력이 데이터 패킷 30개 이하로, 이에 의한 각 지구국의 부담이 작다. 이것은 시스템 내의 모든 지구국이 고정할당방식에 의해 공평한 서비스를 보장받기 때문이다. 이러한 시스템 큐의 최대 크기는 L\_ratio보다는 시스템 자체에 부여되는 통신량, 즉 부하(G)에 의해 증가한다.

## V. 결론 및 이후의 연구방향

본 연구에서는 특히 중간 이상의 트래픽 부하에서의 효율적인 채널이용과 안정된 전송지연, 그리고 각 지구국의 트래픽에 대한 서비스의 공평성을 제공할 수 있는 프로토콜의 설계에 목적을 두었다. 이를 위해 실저위성채널의 구조를 논리적으로 완전히 독립적으로 운영되는 전용 예약채널 및 데이터 전송채널로 분리하여, 독립적인 예약채널과 데이터전송 채널 상에서의 다중접근 알고리즘을 설계했다. 본 알고리즘에서는 예약이 데이터채널의 프레임단위에서 이루어지는 것이 아니라 각각의 전송시간슬롯 단위로 이루어지기 때문에 프레임구조가 필요없으며, 이에 따라 프레임 내의 채널 낭비나 프레임구조에 의한 전송지연을 줄일 수 있었다. 그리고 예약채널에 대한 접근은 고정할당 방식을 이용함으로써 각 지구국에 공평한 서비스를 제공하며, 시스템의 부하에 따라 최대 전송지연시간을 보장한다. 또한, 예약채널에 의한 예약기법을 이용하기 때문에 데이터패킷 전송에서 충돌을 완전히 제거하고, 전송채널에 대한 예약이 각각의 슬롯에 대해 이루어지게 하여 채널의 효율을 증가시킬 수 있게 했다.

시뮬레이션 결과 시스템 내의 지구국의 갯수 N이 증가함에 따라 데이터 패킷의 전송지연시간이 약간씩 증가하지만, 저부하로부터 부하가 약 0.96-0.97까지의 폭넓은 구간에서 매우 안정된 전송지연 특성을 보였다. 또한 고정할당 방식에 의해 각 지구국의 데이터 패킷들에 대한 서비스가 공정하게 이루어지기 때문에 각 전송버퍼에서의 평균 지연시간 분포 및 평균 대기 패킷의 수도 서로 비슷하게 측정되었다. 채널의 효율은 시스템의 부하가 증가함에 따라 선형적으로 증가하여 만족할 만한 결과를 보여준다. L\_ratio의 증가에 의한 전송지연시간 곡선의 변화는 거의 없지만, 전송지연자체는 줄어든다.

이후의 연구방향으로는 고정할당 방식에서의 문제점인 지구국의 추가와 제어에 관련된 망제어 알고리즘 및 보다 능동적인 트래픽제어 알고리즘의 보완 등이 있다.

## 참 고 문 헌

1. Irwin Mark Jacobs, Richard Binder, Estil V. Hoversten, "General Purpose Packet Satellite Networks," Proceeding of IEEE, Vol.66, No.11, p.1448-1467, NOV 1978.
2. Gilbert Falk, "Integration of Voice and Data in the Wideband Packet Satellite Network," IEEE Journal on Selected Area in Comm, Vol. SAC-1, No.6, p.1071-1083, DEC 1983.
3. 과학기술처, "데이터 전송용 위성 통신 시스템 개발에 관한 연구," 1990.
4. Timothy Pratt, Charles W. Bostian, "Satellite Communications," John Wiley & Sons, Inc, 1986.
5. Dimitri Bertsekas, Robert Gallager, "DATA NETWORK," Prentice Hall, Inc, 1987.
6. D. Raychaudhuri, K. Joseph, "Channel Access Protocols for Ku-band VSAT Networks: A Comparative Evaluation," IEEE Communication Magazine, vol.26, no.5, p.34-44, MAY 1988.
7. Simon S. Lam, "Satellite Packet Communication Multiple Access Protocols and Performance," IEEE Trans.on.comm, Vol.COM-27, No. 10, p.1456-1466, OCT 1979.
8. Maralidhar M. Balagangadhar, Raymond L. Pikholtz, "Analysis of a Reservation Multiple Access Technique for Data Transmission via Satellite," Trans.on.comm, Vol.com.27, No.10, OCT 1979.
9. Edward P. Greene, Anthony Ephremides, "Distributed Reservation Control Protocols for Random Access Broadcasting Channels," IEEE Trans.on.comm, Vol.COM-29, No.5, p.726-735, MAY 1981.
10. Ostap S. Kosovych, "Fixed Assignment Access Technique," IEEE Trans.on.comm, Vol. COM-26, No.9, p.1370-1376, SEP 1978.
11. Fouad A. Tobagi, Mario Gerla, Richard W. Peebles, Eric G. Manning, "Modeling and Measurement Techniques in Packet Communication Network," Proceeding of the IEEE, Vol.66, No.11, NOV 1978.
12. Averill M. Law, W. David Kelton, "Simulation Modeling and Analysis," MacGrow-Hill Book Company, 1982.
13. M.H. MacDougall, "Simulating Computer Systems Techniques and Tools," The MIT Press, 1987.



**韓正元(Jeong Hang Han) 正會員**  
 1968年 8月 2日生  
 1990年 2月: 延世大學校 電算科學科 卒業  
 1992年 2月: 延世大學校 大學院 電算科學科 卒業  
 1992年 2月~現在: 三星電子 컴퓨터 部門 勤務



**宋周錫(Joo Seok Song) 正會員**  
 1953年 3月 2日生  
 1976年 2月: 서울 大學校 電氣工學科 卒業  
 1979年 2月: 韓國科學院 電氣및 電子工學科 卒業  
 1988年: Univ. of California, Berkeley 電算科學 博士

1979年~1982年: 韓國 電氣通信 研究所 專任 研究員  
 1982年: 中央 電氣 株式會社 開發 諮問  
 1988年~1989年: Naval Postgraduate School 助教授  
 1989年~現在: 延世大學校 電算科學科 副教授  
 ※關心分野: Computer Network, Protocol Engineering, Distributed Operating System