

위성통신을 위한 서비스 적응적인 전송 데이터 전송 프로토콜 설계

김수정*, 이수호* 정회원

Design of Service-adaptive Tactical Data Transmission Protocol for Satellite Communications

Sujeong Kim* and Sooho Lee* *Regular Members*

요약

본 논문에서는 낮은 위성채널 대역폭을 이용한 서비스 적응적인 전송 데이터 전송 프로토콜(STTS, Service-adaptive Tactical-data Transport Stream)을 제안한다. STTS는 기존 디지털 방송 표준인 DVB-S 및 DVB-S2의 전송 스트림 표준을 기반으로 서비스 확장성을 위한 추가 필드 및 신뢰성 보장을 위한 스케줄러를 제공하도록 설계된다. 또한, 다양한 전송 데이터를 낮은 대역폭으로 전송하기 위해서 트래픽 모델과 중요 데이터의 재전송 기능이 데이터 전송 패킷 손실에 미치는 영향을 시뮬레이션을 통하여 검증하고, 이를 기반으로 STTS를 적용한 시스템 설계 시 고려할 사항을 확인한다.

Key Words : satellite communications, service-adaptive, transmission protocol, tactical communications

ABSTRACT

In this paper, we propose a Service-adaptive Tactical Data Transmission Protocol (STTS) based on Satellite Communications with narrow bandwidth. STTS is designed to provide additional field for scalability and scheduler for reliability of transport stream protocol based on digital broadcasting standard, DVB-S and DVB-S2. It is also verified the effects of lost data packets with narrow bandwidth through the simulator by traffic model and re-transmission of critical data, and checked the design considerations based on STTS system.

I. 서론

최근 전쟁 방식은 지상, 공중, 해상, 수중, 우주 등 다차원 공간에서의 정보 수집 및 다양한 통신 기술을 활용하는 네트워크 중심전(NCW, Network-Centric Warfare)으로 진화하고 있다. 이러한 네트워크 중심전에서 위성통신은 전송 데이터를 실시간으로 전송할 수 있는 핵심 통신체계로써 음성, 데이터, 이미지 등의 다양한 서비스로 제공되는 전송 정보 전달을 위한 신속한 전송이동통신 체계로 운용된다.

위성통신의 장점은 지역적 영향 없이 정보를 동시에 전송하는 동보 전송 및 지상망 시설 없이 정보를 공유하는 것이다. 이러한 위성통신의 장점을 이용하면 추가적인 지상망 시설의 설치 없이 전송 데이터 전송을 전장에서 신속하게 전송할 수 있다. 그러나 점점 높아지는 위성통신 주파수 대역 및 넓은 통신 거리에 따른 다양한 감쇠 요인과 넓은 통신 커버리

지는 적군에 의한 제밍에 노출될 수 있다. 따라서 위성통신의 장점을 활용하고 단점을 극복하기 위해서는 전송 데이터를 낮은 점유 대역폭을 이용하여 효율적으로 전송하는 것이다. 또한, 낮은 점유 대역폭을 기반으로 음성, 텍스트 등 서비스 종류에 따라 확장성과 신뢰성을 제공하며, 실제 활용을 위한 시스템 설계 및 구현이 용이하도록 설계되어야 한다.

본 논문에서는 위성통신 디지털 비디오 전송 표준인 DVB(Digital Video Broadcasting) 기술을 기반으로 전송 데이터를 32kHz 이하의 낮은 통신 대역폭을 이용하여 실시간으로 전송할 수 있는 위성통신 기반의 서비스 적응적인 전송 데이터 전송 프로토콜을 제안한다.

본 논문의 구성은 다음과 같다. II장에서는 디지털 방송 프로토콜 표준인 DVB-S, DVB-S2의 특성을 살펴본다. III장에서는 제안한 전송 데이터 전송 프로토콜에 대해서 자세하게 설명하며, IV장에서는 시뮬레이션을 통하여 서비스별 트래

*LIG넥스원 우주영상연구센터 (sujeong.kim@lignex1.com, soohoo.lee@lignex1.com)

접수일자 : 2016년 9월 12일, 수정완료일자 : 2016년 9월 29일, 최종 게재확정일자 : 2016년 9월 30일

픽 패킷에 대한 손실 및 수용 가능한 트래픽 량을 검증할 것이다. 마지막으로 V장에서는 결론 및 향후 연구 방향을 제시한다.

II. 관련 연구

디지털 위성 방송의 표준으로써 비디오와 오디오에 대한 규격을 정의하고, 다중화를 위해 필요한 MPEG-2 TS 패킷 구조와 다중화된 비디오, 오디오 및 서비스 정보를 위성을 통해 전송하기 위한 DVB-S와 DVB-S2의 규격의 전송 시스템에 대하여 설명한다.

1. DVB-S/DVB-S2 표준

위성 신호를 수신하기 시작하면 DVB-S 물리 계층 데이터는 1 바이트의 Sync 정보에 의해 187 바이트의 MPEG-TS 데이터에 대한 패킷 동기를 맞추는데 사용한다. 1 바이트의 Sync 정보를 포함한 188 바이트 단위의 전송 스트림 (TS, Transport Stream) 8개 단위로 에너지 분산을 위한 랜덤화 과정을 거치며 188 바이트의 전송 스트림 블록은 부호화되고 변조된다. DVB-S2 물리 계층 데이터는 프레임 단위로 구성되며, 프레임의 시작점과 프레임의 구성 정보를 포함하는 90 심볼의 물리 계층 헤더와 90 개의 변조 심볼로 이루어진 일련의 슬롯들로 구성된다. 물리 계층 프레임은 헤더를 제외하고 스크램블링 된 후 정해진 변조 방식에 따라 변조되어 전송된다[1].

2. 프로토콜 분석

DVB-S 및 DVB-S2 프로토콜 분석은 크게 입력 스트림, 채널 코딩, 송신 모드의 3가지로 수행하였다. DVB-S의 입력 스트림은 MPEG-2 TS이며, 고정된 입력 속도에 맞추어 하나의 스트림을 처리한다. DVB-S2의 경우는 다중 스트림을 처리할 수 있으며, 특히 오디오 및 비디오 정보 외에 IP 정보를 처리할 수 있으므로 다양한 입력 속도를 지원한다. 채널 코딩의 경우에는 DVB-S가 위성통신에서 많이 사용되는 Reed-Solomon 및 컨볼루션 부호화를 사용하여 burst error에 강한 반면 높은 오버헤드를 가진다. DVB-S2는 새로운 에러정정코드로서 BCH(Bose Chaudhuri Hocquenghem) 및 LDPC(Low Density Parity Check)를 사용하여 AWGN(Additive White Gaussian Noise)채널에서 좀 더 나은 성능을 보인다. 반면에 구현이 복잡한 단점을 가진다. DVB-S는 모든 프레임이 동일한 부호화 및 변조방식을 사용하므로 고정 전송 속도를 지원하는 반면에 DVB-S2의 경우에는 고정 전송 속도는 물론 적응적 부호화 및 변조방식을 지원하므로 환경에 따른 감쇠에 강한 면모를 보이지만 각 프레임의 길이가 달라지므로 동기를 맞추기에 어려운 단점을 가진다[2][3].

본 논문에서는 DVB-S 및 DVB-S2에서 모두 사용되는 MPEG-2 TS[4]를 기반으로 프로토콜을 설계하였고, DVB-S에서의 고정 크기의 MPEG-2 TS 패킷을 기반으로 DVB-S2의 단점이었던 하나의 고정된 데이터 수신 부분을 DVB-S2의 장점인 다양한 데이터 수신 처리를 통합하여 프로토콜을 설계하였다. 서비스 확장성을 위하여 Optional 필드를 제공하여 추가적인 서비스를 지원할 수 있도록 하였고, 스케줄러 기능을 추가하여 우선순위가 높은 서비스는 먼저 처리하거나 반복적으로 전송할 수 있도록 하여 서비스의 신뢰성을 제공한다.

III. 본론

본 논문에서 제안한 서비스 적응적인 전송 데이터 전송 프로토콜(STTS, Service-adaptive Tactical-data Transport Stream)은 위성통신에서의 항재밍을 위한 낮은 점유 대역폭을 이용하여 실시간 전송 데이터의 서비스 종류에 따라 확장성과 신뢰성을 제공하기 위함이다. STTS는 또한 시스템 설계 및 구현이 용이하도록 프로토콜 구조를 제공한다. 그림 1은 STTS의 전반적인 개념을 나타낸다. 다양한 입력 서비스는 서비스 데이터 유닛(SDU, Service Data Unit)으로 정의되어 STTS 시스템을 통해 STTS PDU (Protocol Data Unit)로 변환하여 전송하는 구조로 이루어진다. STTS 시스템은 게이트웨이, MUX, Classifier 및 변조기로 구성된다.

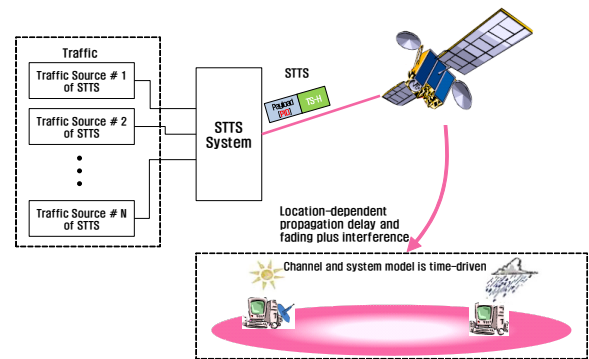


그림 1. STTS 개념도

STTS 시스템은 음성 코덱(Adaptive Multi-rate CODEC)을 사용하여 아날로그 음성 정보의 부호와 및 패킷화를 수행하고 패킷화된 음성 정보를 Transcoding한다. 따라서 아날로그 무전기와 연결한 음성 전송 정보를 직접 연결하여 처리한다. MUX 및 Classifier는 다양한 입력의 SDU를 STTS PDU로 변환하고, Program ID에 따라서 트래픽들을 분류화(Classification)을 한다. 변조기는 STTS PDU를 위성 채널에 보내기 위한 변조 과정을 수행하고 PHY 프레임 생성 및 부호화 처리를 한다. 그림 2는 STTS 시스템의 동작 구성을 보여준다.

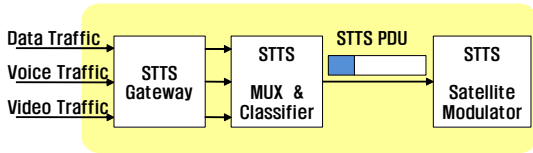


그림 2. STTS 시스템 동작 구성

STTS 게이트웨이에서 처리되는 SDU는 우선순위 또는 서비스 특성에 따라서 스케줄러를 제공한다. 그림 3은 STTS 시스템의 스케줄러를 나타낸다. 입력 SDU는 각각의 트래픽 특성에 따라서 스케줄링 된다. 예를 들어, 음성 트래픽의 경우 일정한 간격을 가지고 주기적으로 전송되어야 하므로 전용 버퍼에 할당한다. 또한, 트래픽 우선순위를 고려하여 Protection Level 또는 Service Level을 결정하고, 결정된 사항에 따라서 STTS PDU 패키지를 구성한다. 최종적으로 변조한 후 위성 채널을 통해 전송된다.

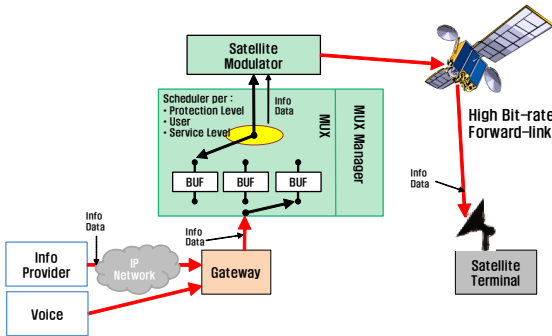


그림 3. STTS 시스템 스케줄러

1. STTS 패킷 헤더 구조 및 정의

본 프로토콜 설계에서는 32kbps의 낮은 위성통신 대역폭에서 운용할 수 있도록 패킷의 총 길이를 155 바이트, 헤더는 3바이트로 정의하여 설계하였다. STTS 패킷 헤더는 Program ID로 트래픽 종류별로 구분이 가능하며, 다양한 입력의 SDU에 대해서도 효율성 제공을 위해 SDU Packing 및 Fragmentation 기능을 지원 및 제공한다. 또한, Optional 필드가 있어 총 8가지 Option별 확장성을 제공하는 특징을 가진다. 그림 4는 STTS 패킷의 헤더 구조를 나타낸다.

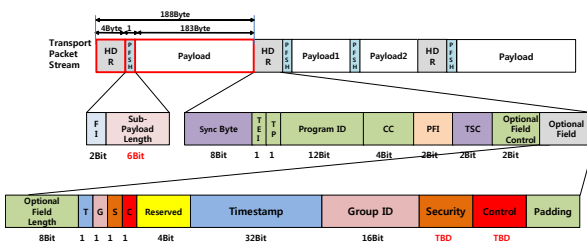


그림 4. STTS 패킷 헤더 구조

STTS 패킷 헤더의 길이는 3바이트이며, 총 8가지 필드로 구성된다. 또한 8가지 Optional 필드를 사용하여 확장 지원

이 가능하다. Sync 바이트는 1바이트로 구성되며, 패킷의 시작 바이트를 나타낸다. Program ID 필드는 4비트로 구성되며, 데이터 타입을 나타낸다. 예를 들면, '0001'은 음성 메시지, '0010'은 제어 메시지, '0011'경보 메시지 등으로 구분된다. CC는 Continuity Counter의 약자로서 패킷의 순차적인 전송에서 각 패킷마다 1씩 증가하면서 4비트 회전 증가 값으로 4비트로 구성된다. 각 Program ID 별로 별도의 CC가 증가된다. PFI 필드는 2비트로 구성되며, Packing 및 Fragmentation 여부를 나타낸다, TEI은 Transport Error Indication로서 STTS 패킷에 비트에러가 발생하였을 경우, 변조기에 의해 설정된다. TP는 Transport Priority로서 STTS 전송 경로에 같은 성격의 전송 매체가 있다면 그 중에서 우선순위를 결정한다. '1'이 '0'보다 높은 우선순위를 갖는다. TSC(Transport Scrambling Control) 필드는 2비트로 구성되며 해당 TS가 스크램블 되었는지 여부를 나타내는 필드이다. Optional 필드는 2비트로 구성되며 optional 필드 사용 여부를 나타낸다. 표 1과 2는 STTS 패킷 헤더 필드 및 Optional 필드 정의를 보여준다.

표 1. STTS 패킷 헤더의 필드 정의

필드명	길이 (bits)	정의
Sync Byte	8	"0x47" 값 사용
Program ID	1	Payload의 데이터 타입
CC	4	ID별 패킷의 순차적인 전송
PFI	2	Packing 및 Fragmentation 상태
TEI	1	패킷 에러 설정
TP	1	데이터 우선순위 결정
TSC	2	Payload의 Scrambling 여부
Optional Field Control	2	Optional 필드의 사용 여부 "00" Reserved "01" Optional 필드가 없음 "10" Optional 필드만 있음 "11" Payload에 이어서 Optional 필드가 있음

표 2. STTS 패킷 헤더의 Optional 필드 정의

필드명	길이 (bits)	정의
Optional Field Length	8	총 Optional 필드의 총 바이트 수
T	1	'1', Timestamp 필드 포함
G	1	'1', Group ID 필드 포함
S	1	'1', Security 필드 포함
C	1	'1', Control 필드 포함
Timestamp	32	Program Clock Reference
Group ID	16	그룹 및 가입자 관리
Security	TBD	암호화 지원
Control	TBD	제어 기능

2. STTS 패킷의 PFSH 필드 정의

PFSH는 Packing and Fragmentation Sub Header의 약자로서 1바이트로 정의된다. 본 필드는 입력 SDU가 Fragmentation

되었을 경우에 Payload의 Fragmentation 순서를 알려준다. 그리고 Packing이 되었을 경우는 Packing된 SDU 길이를 알려준다. 표 3은 STTS 패킷의 PFSH 필드를 정의하였다.

표 3. STTS 패킷의 PFSH 필드 정의

필드명	길이 (bits)	정의
FI	2	입력 SDU의 Fragmentation 여부를 나타냄 (Fragmentation Indicator) [정의] "00": Reserved "01": Start Fragmentation "10": Last Fragmentation "11": Continuous Fragmentation
Sub-Payload Length	6	Packing된 입력 SDU 길이를 알려줌

3. STTS 패킷의 운용

입력 SDU가 Fragmentation 또는 Packing 될 경우 설계된 STTS 패킷 헤더의 동작을 확인한다. PFI 필드 값에 따라서 Fragmentation만 되었을 경우와 Packing만 되었을 경우, 또한 Fragmentation과 Packing 둘 다 되었을 경우를 확인하였다.

3.1 Fragmentation and No Packing

입력 SDU가 302바이트라고 가정할 경우 입력 SDU는 Fragmentation되고, Packing은 발생하지 않는다. Fragmentation을 수행하였다는 것을 STTS 패킷 헤더 내의 PFI 필드 값을 "01"로 설정함으로써 수신 측에 알려준다. 입력 SDU는 STTS 패킷 크기에 따라서 둘로 나뉘게 되는데, 이때 각각 Fragmentation된 입력 SDU 순서를 PFSH 내의 FI 필드에 설정한다. 맨 처음 Fragmentation 된 SDU는 "01", 마지막 Fragmentation 된 SDU는 "10"으로 설정한다. 이것은 SDU 순서가 바뀌더라도 FI 필드에 표기된 것을 확인하여 올바르게 디코딩 할 수 있다. Packing은 발생하지 않았으므로, PFSH 필드 내에 Sub-Payload Length 값은 "000000"으로 설정한다.

3.2 No Fragmentation and Packing

입력 SDU가 각각 100바이트, 50바이트라고 가정할 경우, 이 두 SDU의 크기를 합한 값이 하나의 STTS 패킷에 수용될 수 있는 크기이므로 Packing이 가능하다. Packing이 수행되었다는 것은 STTS 패킷 내의 PFI 필드 값이 "10"으로 설정함으로써 수신 측에 알려준다. 한 STTS 패킷 내에 두 개의 입력 SDU가 Packing 되므로, Packing 된 SDU 크기를 알려주기 위한 서브 헤더로 2바이트가 사용된다. 첫 번째 Packing되는 100바이트의 SDU 사이즈를 수신 측에 알려주기 위한 목적으로 PFSH 안의 Sub-Payload Length를

"011001"로 설정한다. 그리고 두 번째 Packing되는 50바이트 SDU 크기를 "010101"로 설정한다. Fragmentation은 수행하지 않았으므로 FI 필드는 "00"으로 설정한다.

3.3 Fragmentation and Packing

입력 SDU 크기가 201바이트, 100바이트, 151바이트로 연속해서 입력될 경우, 첫 번째 입력 SDU는 STTS 패킷 크기보다 크기 때문에 Fragmentation이 수행되고, 첫 번째 입력 SDU가 Fragmentation 된 후 나머지 SDU와 두 번째 입력 SDU는 Packing이 수행된다. 세 번째 입력 SDU는 STTS 패킷이 수용할 수 있는 크기이므로 Fragmentation과 Packing이 수행되지 않는다. 그림 5는 Fragmentation과 Packing이 되었을 경우 PFI 필드 값의 변화를 보여준다.

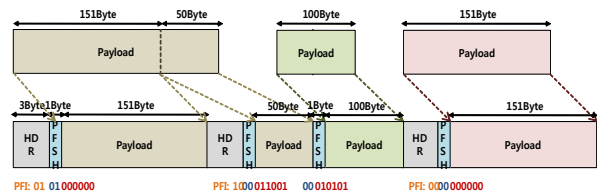


그림 5. Fragmentation and Packing

IV. 시뮬레이션

시뮬레이션을 위하여 입력 SDU에 해당하는 트래픽을 크게 음성, 데이터, 제어 트래픽으로 분류하였다. 데이터는 통제 명령, 정보, 경보, 일반 메시지로 구분하였고, 일반 메시지를 제외한 데이터는 비화 또는 음어화하여 송신하는 것을 가정하였다. 음성 트래픽은 12.2kbps, 총 데이터 트래픽은 15.32kbps로 가정하였다. 이는 위성통신 대역폭 32kbps 고려시 트래픽 Margin을 고려하여 총 데이터 속도를 약 28kbps로 산정하였다. 트래픽 발생 예측에서는 통제 명령은 초당 1개, 정보 및 제어 메시지는 초당 10-20개 송신, 일반 메시지는 초당 1개 정도 전송하는 것을 기준으로 하였다.

시뮬레이션에 대한 전반적인 개요와 시뮬레이션 환경은 그림 6과 표 4에 각각 나타낸다.

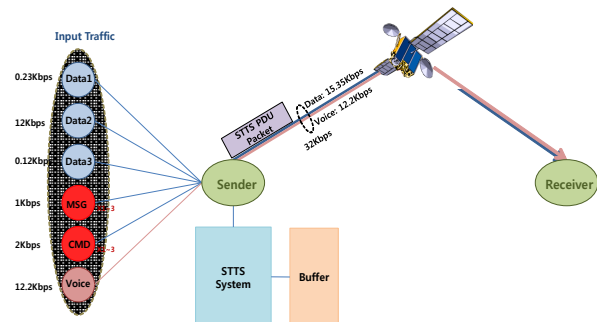


그림 6. 시뮬레이션 개요

입력 트래픽은 Data1(통제 명령), Data2(정보), Data3(경보), MSG(일반 메시지), CMD(제어 메시지), Voice(음성)의 총 6가지로 구성되며, 정의된 트래픽 속도에 따라 생성된다. 본 시뮬레이션은 입력 SDU의 트래픽 패턴 특성에 따라서 STTS 시스템에서 수용할 수 있는 트래픽 량을 시험하였으며 수용하지 못하는 경우 주요 손실 트래픽 종류를 분석하였다. 이를 통하여 STTS 시스템의 수신 버퍼 사이즈를 결정할 수 있었다. 그리고 중요 트래픽인 일반 및 제어 메시지의 Duplication 동작 및 예외 처리 동작에 대하여 시뮬레이션 하였다. 시뮬레이션 환경은 표 4와 같다. 시뮬레이터는 NS-2[5]를 사용하였고, 100초 동안 시뮬레이션 하였다. 큐는 DropTail을 사용하였고, 초기 큐 사이즈는 10K바이트로 설정하였다. 위성 채널 대역폭은 32Kbps, 링크 에러율은 없다고 가정하였다. 마지막으로 Traffic 생성기의 종류는 CBR(Constant Bit Rate), Exponential On/Off, Pareto On/Off의 3가지를 사용하였다.

표 4. 시뮬레이션 환경

분류	내용
시뮬레이터	NS-2 (Network Simulator)
시뮬레이션 시간	100S
큐(Queue) 종류	Droptail Queue
큐(Queue) 크기	110Kbytes(가변)
대역폭	32Kbps -Voice:12.2Kbps, Data: 15.35Kbps
링크 에러율	0%
입력 Traffic	-Data1, Data2, Data3 -Message, Command -Voice
Traffic 생성기 종류	-CBR -Exponential On/Off -Pareto On/Off

1. CBR 트래픽 적용

CBR 트래픽 생성기는 일정한 시간 간격으로 고정된 크기를 가진 패킷을 만들어내는 생성기이다. CBR 객체의 멤버 변수는 패킷 생성률, 패킷 크기, 최대 패킷 개수, 패킷 간 전송 시간 간격, 패킷 생성 주기에 랜덤 노이즈 등으로 구성된다. CBR 트래픽 생성기로 만들어진 입력 SDU는 STTS 시스템으로 입력되는데 패킷 손실 없이 모든 패킷을 수용하기 위해서는 적절한 큐 크기를 가져야 한다. 만일 큐 크기가 크면 패킷 손실은 없지만 지연이 많이 발생하는 문제점이 있기 때문에 큐 크기와 패킷 손실률을 확인하였다. 패킷 손실률은 송신 패킷 수 대비 수신 패킷 수를 카운트하여 계산하였으며, 큐 크기에 따른 패킷 손실 개수는 표 5에서 보여준다.

그림 7은 큐 크기 변화에 따른 패킷 손실률을 나타낸다. STTS 시스템은 110Kbytes 이상으로 큐 크기를 설정하였을 경우 패킷 손실을 발생하지 않는 것을 확인할 수 있다. 그림 8은 서비스 별 패킷 손실률을 보여준다. 트래픽 비율이 높은 음성 패킷 손실이 가장 큰 것을 알 수 있다.

표 5. 큐 크기에 따른 패킷 손실 개수

Buffer Size(bytes)	Sending Packet	Received Packet
10	14606	13756
20	14606	13841
40	14606	13973
60	14606	14208
80	14606	14479
100	14606	14560
110	14606	14606
120	14606	14606
140	14606	14606
160	14606	14606

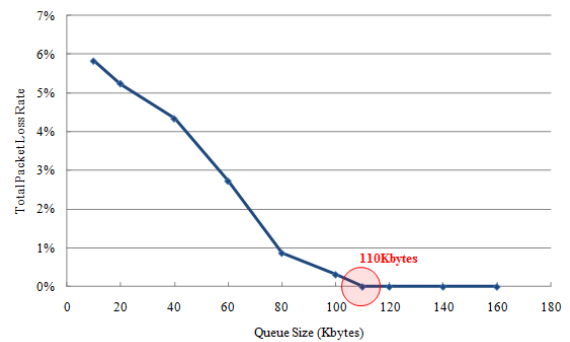


그림 7. 큐 크기 변화에 따른 패킷 손실률

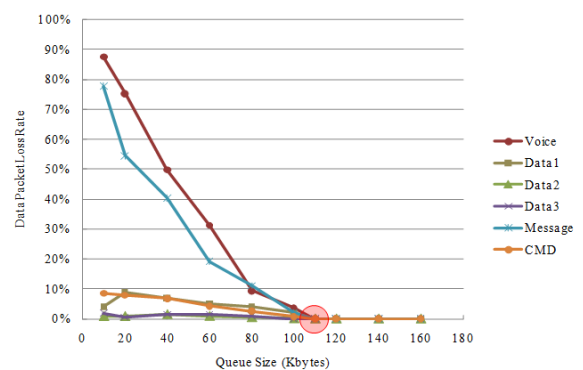


그림 8. 큐 크기 변화에 따른 입력 SDU별 패킷 손실률

2. Exponential On/Off 트래픽 적용

Exponential On/Off 트래픽 생성기는 On 기간과 Off 기간의 길이를 Exponential 분포에 따라 정하는 것을 의미한다. On 기간에 패킷을 일정한 Rate로 생성하고, Off 기간에는 패킷을 생성하지 않는다. 멤버 변수는 On 기간동안 패킷 생성률, 패킷 크기, On 기간의 평균 길이인 burst time과 Off 기간의 평균 길이인 idle time으로 구성한다. 큐 크기는 CBR 트래픽 특성에 맞게 설정된 큐 크기인 110바이트로 설정하고, burst time과 idle time을 0.1~0.9s, 0.9~0.1s로 변화하면서 측정된 패킷 손실을 시뮬레이션 하였다. 그림 9는 burst와 idle time 설정에 따른 패킷 손실 결과를 나타낸다. burst time이 적고, idle time이 클수록 패킷 손실률이 증가됨을 확인할 수 있다. 이는 burst time이 적고, idle time이 크면 단

시간에 생성된 많은 패킷이 STTS 시스템으로 한꺼번에 유입되기 때문이다. 그러므로 입력 SDU가 Exponential 트래픽 특성을 가질 경우는 큐 크기를 다시 설정할 필요가 있다. 그림 10의 경우는 패킷 손실률이 약 10%로 가장 큰 burst time인 경우에 대해서 큐 크기를 재설정된 결과를 나타낸다. 초기 큐 사이즈는 CBR 트래픽 생성기를 사용했을 때 결정된 110바이트로 설정하였다. 큐 크기가 증가함에 따라서 STTS 시스템에서 수용할 수 있는 트래픽 량이 커짐으로써, 패킷 손실률이 줄어드는 것을 볼 수 있다. 결국 큐 크기가 220바이트 이상으로 설정하였을 경우 패킷 손실이 발생하지 않는 것을 확인 할 수 있다.

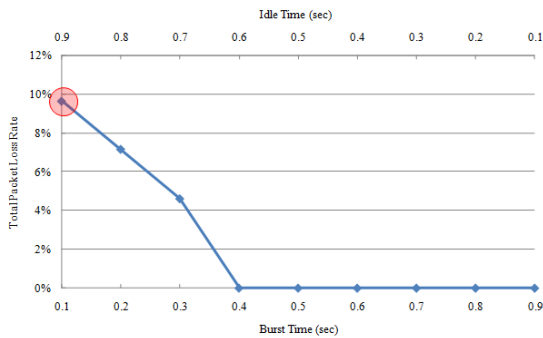


그림 9. Burst Time, Idle Time 설정에 따른 패킷 손실률 (Burst Time=0.1~0.9s, Idle Time=0.9~0.1s)

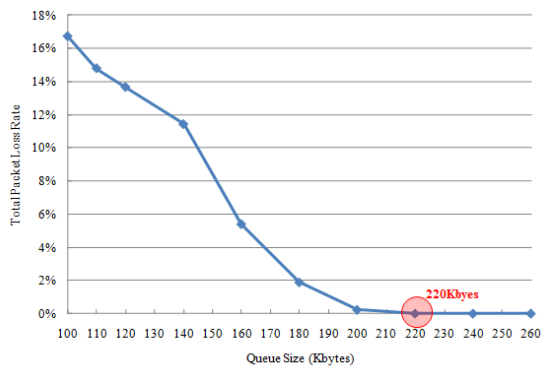


그림 10. 큐 사이즈 변화에 따른 패킷 손실률 (Burst Time=0.1s, Idle Time=0.9s)

3. Pareto On/Off 트래픽 적용

Pareto On/Off 트래픽 생성기는 On 기간과 Off 기간의 길이를 Pareto 분포에 따라 정하는 것을 의미한다. On 기간에 패킷을 일정한 Rate로 생성하고, Off 기간에는 패킷을 생성하지 않는다. 멤버변수는 On 기간 동안 패킷 생성률, 패킷 크기, On 기간의 평균 길이인 burst time과 Off 기간의 평균 길이인 idle time 및 Pareto 분포에서 사용하는 shape 파라미터로 구성된다. Exponential On/Off 시뮬레이션과 동일하게 시뮬레이션을 수행하였다. 그림 11과 같이 Exponential과 비슷한 결과를 나타내지만 Exponential 트래픽에 비해 적은 패

킷 손실률을 보였다. 이는 Exponential 트래픽이 Pareto 트래픽에 비해 누적 분포가 크기 때문이다. 그러므로 Pareto 트래픽 특성을 가질 경우도 큐 크기를 다시 설정할 필요가 있다. 그림 12는 패킷 손실률이 약 8%로 가장 큰 burst time과 idle time인 경우에 대하여 큐 크기를 재설정된 결과를 나타낸다. 최종적으로 큐 크기가 135Kbytes 이상으로 설정하였을 경우 패킷 손실이 발생하지 않는 것을 확인할 수 있다. 이는 Exponential 트래픽 보다 낮은 값을 가지는 반면에 CBR 트래픽에 비해서는 높은 값을 가진다.

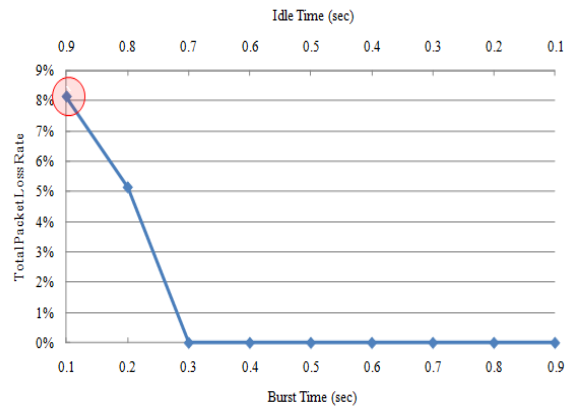


그림 11. Burst Time, Idle Time 설정에 따른 패킷 손실률 (Burst Time=0.1~0.9s, Idle Time=0.9~0.1s)

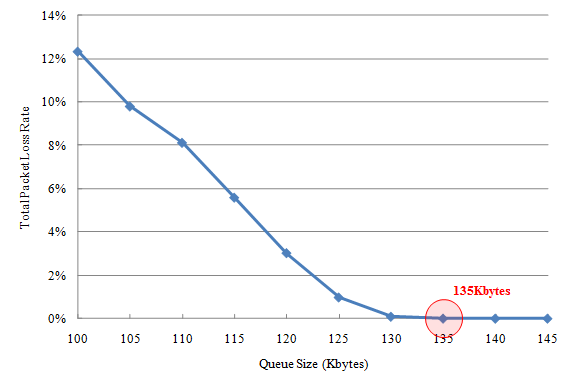


그림 12. 큐 사이즈 변화에 따른 패킷 손실률 (Burst Time=0.1s, Idle Time=0.9s)

4. 기타 시뮬레이션

중요 트래픽인 메시지를 수신 측에서 에러없이 디코딩하기 위하여 1~3번 duplication 동작을 수행하여 전송할 때 다른 패킷들에 미치는 영향을 시뮬레이션 하였다. 그림 13은 CBR 트래픽 생성기를 통해 패킷을 생성할 때, duplication 동작을 수행 시 패킷 손실률을 나타낸다. 일반 메시지와 제어 메시지의 반복 횟수가 증가함에 따라 다른 패킷들의 패킷 손실률이 증가한다. 특히 트래픽 비율을 많이 차지하는 음성 트래픽의 손실이 큰 것을 알 수 있다.

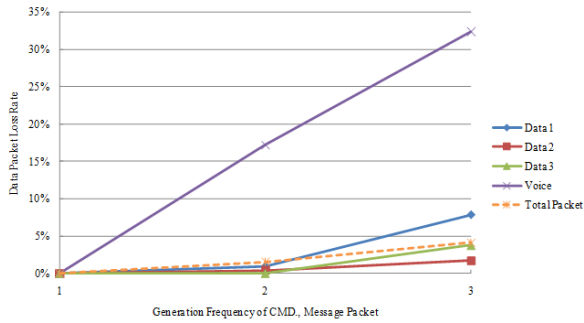


그림 13. Duplication 횟수에 따른 패킷 손실률(CBR 트래픽)

그림 14와 15는 동일한 조건에서 Exponential On/Off 트래픽일 경우와 Pareto On/Off 트래픽일 경우의 Duplication 횟수에 따른 패킷 손실률을 나타낸다. Exponential 트래픽은 CBR 트래픽에 비해서 누적 분포가 크기 때문에 CBR 트래픽에 비해서 전체적으로 다른 패킷들의 손실률이 높은 것을 확인할 수 있다. Pareto 트래픽의 경우는 Exponential 트래픽과 비교했을 때, 음성 패킷을 제외한 모든 패킷 손실률이 작은 것을 알 수 있다.

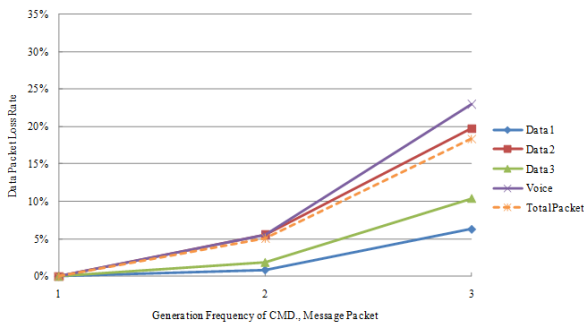


그림 14. Duplication 횟수에 따른 패킷 손실률 (Exponential On/Off 트래픽)

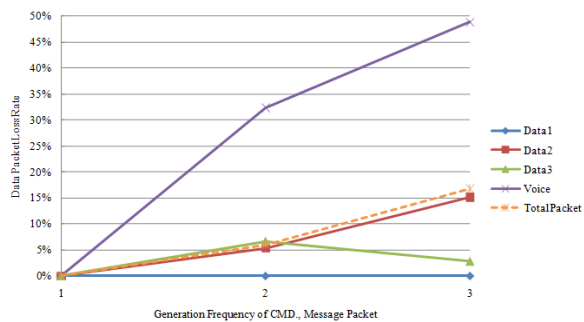


그림 15. Duplication 횟수에 따른 패킷 손실률 (Pareto On/Off 트래픽)

그림 16은 시뮬레이션 시간을 증가 시켰을 경우의 패킷 손실률을 나타낸다. 큐 크기는 앞선 시뮬레이션에서 각각 트래픽 패턴에 따라서 결정된 값을 설정하였다. 즉, CBR 트래픽(110Kbytes), Exponential 트래픽(220Kbytes), Pareto 트래픽(135Kbytes)으로 설정하였다.

기존 100초로 시뮬레이션 했을 때 발생되지 않았던 패킷 손실이 시뮬레이션 시간이 증가함에 따라서 발생한다는 것을 확인할 수 있었다. 이는 시뮬레이션 시간이 증가할수록 큐에 쌓이는 패킷의 양보다 전송되는 패킷의 양이 적기 때문이다. 그러므로 STTS 시스템의 사용 시간을 고려하여 큐 크기를 설계하는 것이 요구된다.

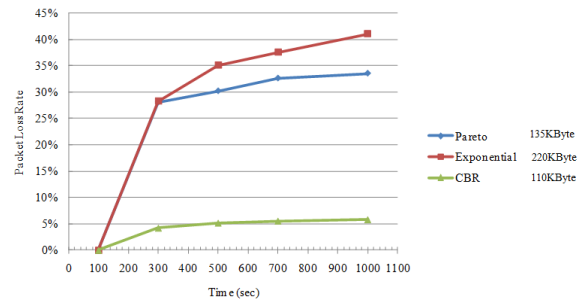


그림 16. 시뮬레이션 시간 증가에 따른 패킷 손실률

V. 결론

낮은 대역폭의 위성 채널을 활용하여 전송 데이터를 전송하는 프로토콜을 설계한 후 다양한 입력 SDU의 트래픽 패턴 특성에 따라서 본 논문에서 제안한 STTS 시스템에서 사용할 수 있는 트래픽 량을 시험하였으며 수용하지 못하는 경우 주요 손실 트래픽의 종류를 분석하였다. 이를 통하여 수신 버퍼 크기를 결정하였다. 그리고 중요 트래픽인 일반 메시지와 제어 메시지의 Duplication 동작 및 예외 처리 동작에 대하여 시뮬레이션 하였다. 시뮬레이션 결과 제안된 STTS를 이용한 시스템 설계 시 두 가지 사항이 요구된다. 첫 번째, 시스템의 입력 트래픽 모델 검토가 필요하다. 즉, 실제 전장에서 각 호스트에 전달하는 트래픽의 burst한 정도를 파악하여 시스템 동작 및 운영에 따른 트래픽 변화를 고려해야 한다. 두 번째, 트래픽 모델 및 특성에 따라 시스템 버퍼 크기나 스케줄러가 고려된 시스템 설계를 해야 한다. 특히, 높은 패킷 손실을 보이는 음성 트래픽에 대한 스케줄러를 설계하여 향후 완전한 위성통신을 위한 전송 시스템 설계 및 구현을 수행할 예정이다.

참고 문헌

[1] ETSI EN 300 421, 'Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for 11/12GHz satellite services', pp. 8-14, 1997.08.
 [2] ETSI TS 101 154, 'Digital Video Broadcasting (DVB); Implementation guidelines for the use of Video and Audio Coding in Broadcasting Applications based on the MPEG-2

Transport Stream', pp. 89-100, 2012.11.

[3] ETSI EN 302 307(V1.3.1), 'Digital Video Broadcasting (DVB): Second Generation Framing Structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2)', pp. 12-14, pp. 50-71, 2013.03

[4] ISO/IEC 13818-1, 'Generic coding of moving pictures and associated audio information: Systems', pp. 125-130, 2012.11.

[5] The network simulator ns-2, <http://www.nsnam.org>

저자

김 수 정(Sujeong Kim)

정회원

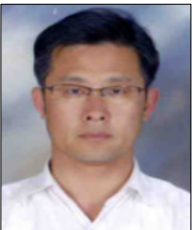


- 1996년 2월 : 한국항공대학교 항공통신정보공학 학사졸업
- 2014년 2월 : 광운대학교 전자통신공학 박사졸업
- 2014년 7월 ~ 현재 : LIG넥스원 우주영상연구센터 수석연구원

<관심분야> : 위성통신, 프로토콜, 군 전송시스템

이 수 호(SooHo Lee)

정회원



- 1989년 2월 : 인하대학교 전자공학 학사졸업
- 1999년 2월 : 인하대학교 전자공학 석사졸업
- 2007년 1월 ~ 현재 : LIG넥스원 우주영상 연구센터 수석연구원

<관심분야> : 합성개구레이더, 시뮬레이션, 인공위성