Systems Design Of A Satellite Link Protocol

John Murphy *   Edward Chow †   Richard Markley ‡

Jet Propulsion Laboratory

California Institute of Technology

Advanced Information Systems Section

Pasadena, California 91109

Abstract

In this paper we show how it might be possible to adapt asynchronous transfer mode technology to satellite links. Asynchronous transfer mode is a high-speed protocol designed with optic fiber as the intended transmission media. Several problems arise when satellite channels are used. We propose to move the error recovery and detection from one layer of the asynchronous transfer mode protocol to a higher layer.

*Academic Part Time, email: murphy@systems.caltech.edu

†Member of Technical Staff, email: edward_chow@isd.jpl.nasa.gov

‡Technical Manager, email: markley@isd.jpl.nasa.gov

We base the retransmission straegy on the service carried, which utilises the ability of asynchronous transfer mode protocol to differentiate services. Our simulation results show that we not only increase the raw data throughput for satellite channels to almost the theoretical limit, but we have improved the data transfer efficiency of the asynchronous transfer mode protocol by 7.5 percent. Our results also show that it is possible to guarantee data services with no loss of data under certain conditions.

Introduction

ATM is an emerging standard for broadband communications and it involves the switching of small fixed-length packets[1,2]. ATM allows the sharing of resources between many different services by statistically multiplexing the sources. While ATM has been proposed for high speed services, at rates of 155 Mb/s and 622 Mb/s, there is increasing interest in ATM at lower rates[3], such as 1.544 Mb/s (T1). The interest is stimulated by two factors: (1) expensive or lack of sufficient bandwidth and (2) lack of services requiring the larger bandwidths.

When dealing with mobile communication systems it is unlikely that the larger bandwidths will be available in the near future. However, there are a number of areas, civilian and military, where mobile or wireless operation is critical and there is a desire to investigate the performance and advantages of ATM in this area. There is also a current need to share the bandwidth resource in a more efficient manner. Multiplexing the services that are currently using the bandwidth is seen as an advantage of ATM.

The Jet Propulsion Laboratory operates the Deep Space Network (DSN) for NASA. The ground facilities of the DSN consists of three main antenna sites, around the world, and JPL, all connected by commercial satellite links. The bit rates being used are generally T1 or less. The services that use these links vary from command files for spacecraft and ground distribution of telemetry, to voice communications among operators and test data. There is also the future possibility of using these links for digital video. The accuracy of the data from and to spacecraft is critical while other data and voice may not be as important. At present there is no automatic method of differentiating between the different services that use the network.

Asynchronous transfer mode has been designed with optic fiber as the expected transmission media. This means that the expected errors will be produced by Gaussian noise and hence will introduce random geometrically distributed bit errors. With bit errors expected to be of the order of 1E-10, it means that, at most, only one bit error will be expected in each 424-bit cell. Therefore error detection and one bit error correction will almost fully protect the data. In satellite links bursty errors are common. These bursts may be a result of Gaussian noise, but rather than producing randomly distributed bit errors they produce a large group of bit errors due possibly to the coding of the channel. It is also possible that degraded performance may occur on the link causing increased bit errors which may destroy the link for a fraction of a second or more. The overall performance of the satellite links is usually about 1E-7 and single-bit correction is not likely to protect the data. The loss for the data and the header of the ATM cell is shown in Fig 1 versus the bit error rate. It is seen that for optic fiber media there is about one cell lost every century for a 155 Mb/s link whereas for satellite links the loss will be about one cell every minute. Therefore different error-detection methods must be implemented when using ATM over satellites. If the errors were likely to be all bursty then the cell header could be used to discard incorrect cells. However with both bursty and random error the information in the cells also needs to be protected from errors. There are many schemes proposed to adapt ATM to non optic fiber environments[4] however these do not take into account the type of error found on the satellite links used by JPL.

The rest of this article is organized as follows: the model for the satellite channel is described; modifications to the ATM protocol are investigated; a model for the system and the simulation are shown and the results are presented.

Channel Model

The satellite link is characterized by the delay across the link and the bit rate of the link. In this article the delay across the link is taken to be 270 ms, and the bit rate of the link is T1, or 1.544 Mb/s. There is also the error due to incorrect decisions on the bits at the receiver. There are many models proposed to model this bit error and most are based on reasoning why the errors occur. Rather than take one of the predefined models or be involved in the actual coding and transmission schemes being used on the link, we use an empirical model based on actual test data taken from the

DSN and we also assume Gaussian noise. Therefore we end up with a combination of burst errors and single-bit errors due to Gaussian noise.

The experimental data used to model the burst-error distribution are from fractional T1 links, which vary in bit rate from 56 kb/s to 224 kb/s and the tests were taken over a period on the order of two days. A total of 1E+11 bits were used from the test. On investigation of these results we first made a distinction between non-burst events and burst events. During the non-burst events only single-bit errors occur in a random manner and the satellite link is in this mode of operation about 99.75 percent of the time. Secondly, while in a burst event, groups of continuous bit errors occur. These burst errors do not occur in a random fashion and so we model the inter arrival time of the burst errors by another empirical distribution. Thus the resulting model is thus made up of random single-bit events and burst events. In the burst event the probability distribution for the inter arrival of burst errors is shown in Fig 2. The mean time between bursts is approximately 36 s, and the variance is 24 s. The duration of the burst itself is given by the burst length and the probability distribution for this is shown in Fig 3. The mean burst duration is 0.15 s with a variance of 0.26 s. Thus a burst may have tens of thousands of bits in error continuously. We model both the burst length and the burst inter arrival in terms of time rather than in terms of bits so that this can be applied to different bit rate links.

Proposed ATM Adaption Layer

The ATM protocol has three layers corresponding approximately to the bottom two layers of the OSI model[5]. The bottom layer, which is called the physical layer, is concerned with the physical transmission of the bits. The second layer, the ATM layer, takes a payload of 48-byte cells and puts a 5-byte header on to it to form a 53-byte cell. This header has the routing and addressing information in it, as well as an 8-bit cyclic redundancy code (CRC) that detects header errors[2] and corrects single-bit errors. The 48-byte payload comes from the third layer, the ATM Adaption Layer (AAL), which adapts the cells to the different services. The lower part of the AAL is called the Segmentation And Reassembling (SAR) sublayer which breaks a message up into cells. There are a number of proposed AAL's specified for different applications[1,2]. The proposed AAL for data transfer is AAL 3/4, which uses a 44-byte information load and a 2-byte header and trailer. In the trailer of AAL 3/4 is a 10-bit CRC that protects the information load from errors. However due to the small CRC, the potential undetected error rate is high (1E-3), and there is no correction for double-bit errors. Furthermore the four bits allowed for a sequence number will limit the throughput on the long-delay satellite links, e.g., 10.5 kb/s when the delay is 270 ms.

We suggest that it is more efficient to consider moving the CRC and the sequence numbers to the higher Convergence Sublayer[6] (CS). By moving the sequence number to this layer we gain larger blocks to put error detection and correction data on. For the AAL 3/4 the overhead is about 8 percent but to modify that for a T1 satellite link we would need at least an 11-bit sequence number. Also by moving the CRC to the CS sublayer the undetected error is reduced to less than 1E-9. The ATM standards specify many AAL's that can be used, but it is permissible to specify a different AAL for a particular application. We suggest that high-speed, long-delay satellite links need a unique AAL. This can be achieved by inserting the sequence number and the CRC at the CS sublayer, requiring a type of framing. The AAL that we propose is capable of supporting 155 Mb/s throughput with a 270 ms delay on a single channel. The proposed framing structure is shown in Fig 4.

In this article it is also suggested that there be service selection to improve efficiency for reliable delivery. This is needed because real-time data and voice usually cannot tolerate delays of one and a half times the round trip propagation delay. For example, voice will tolerate the 270 ms propagation delay, but if it is in error, it will take another 540 ms for retransmission, even assuming no congestion delay. The total delay is 810 ms, which is unacceptable. Therefore there is no use in trying to detect errors or retransmit voice cells. In fact, any service that is delay sensitive in the sense of less than a few seconds is considered here to be real time and so is not framed for error detection and retransmission. For real-time services no framing is required. Even if the link is fully occupied we propose to make room for the retransmission of reliable data by discarding some of the real-time cells, e.g., the voice cells.

It can be seen that the frame is at the CS sublayer and takes 568 bytes of user data and adds 8 bytes of overhead to form a 12 by 48-byte frame to be split up by the

SAR into 12 cells. The 8 bytes of overhead is made up of a 32-bit CRC, which could be the standard IEEE 32-CRC[6]. There is a sequence number of 16 bits which allows a full 155 Mb/s of data on a link with the delay as given above. The sequence number needed for 155 Mb/s depends on the size of the frame, and the error characteristics need to be known for an optimally sized frame. It is shown in the results section that a frame size of about 12 cells is a fair estimate for the conditions of our channels. At the receiver there is a problem when a cell is lost in that it is not know whether it is a voice cell or is one of the framed cells. To overcome this and to maintain the framing synchronizing, a one-bit flag is proposed to show the start of the frame. This one-bit flag is the last bit of the cell and tells whether the cell is the first in the frame or not.

The efficiency that we gain is twofold: first we have only 8 bytes of overhead on a frame of 568 user information bytes and secondly we have the sequence numbers to transmit over long delay links. The gain in efficiency compared to AAL 3/4 is that instead of carrying 528 bytes of user information bytes in 12 cells we now carry 568 bytes, which is a 7.5 percent increase in efficiency.

Simulation Details

The AAL model and the selective retransmission model were simulated using SES/workbench. SES/workbench is a discrete-event package that allows hardware and software simulation. The model of the retransmission scheme and the error probabilities in this paper are created by use of the graphical interface. The SES/workbench compiles the code to C and runs on a four-processor SparcServer 629. Each simulated data point is the equivalent of 17 days on a T1 satellite link, but rather than modeling the 2E+12 bits, only the burst event is simulated. This reduces the simulation to 5E+9 bits and further optimization of the model compresses this to 5E+7 events. In real time this can take between 3 to 36 hours of run time. In total 1300 hours of CPU time was expended in the simulation tests.

The model consists of the generation of real-time and framed cells as shown in Fig 5. There is a buffer at the transmitter, which varies in size, that can be used to influence the throughput. The frames are numbered just as they leave the transmitter buffer with modulo 65 563. The channel has two sources of noise, burst errors and Gaussian noise. At the receiver the real-time cells and all the errored cells are released while the others are framed. If any bits in the frame are in error the whole frame is discarded. If the frame CRC is okay then an acknowledgment is sent back to the transmitter, through a channel in which it also may encounter errors. The transmitter, after transmitting the cells onto the channel, keeps the framed cells in a time-out buffer. This buffer is 111.3 KBytes, which accommodates the two-way propagation delay plus a small processing delay. When an acknowledgment arrives the frame is discarded from the time-out buffer. Otherwise after the time-out delay, the frame skips to the head of the transmitter buffer, after checking that there is sufficient room in the buffer. If there is not sufficient space then real-time cells are discarded from the buffer, and if this does not release enough space then the frame is discarded. The real-time source cells check the delay in the transmit buffer before joining the queue. If the delay is more than a specified value, in this case 50 ms, then the channel is assumed to be in burst error and the cells are discarded.

Results

It is possible to guarantee delivery of the framed cells, with the given error probabilities by means of retransmission. However if the mix of framed cells to real-time cells is high then some framed data will be lost due to the burst errors. Tests were carried out for three type of errors, first only bursty errors, then only Gaussian errors, and finally for both Gaussian and bursty errors. For each of these types of errors the effect of the buffer size is investigated. Also, the effect of changing the framed data mix on the cell loss is investigated.

For burst errors, the cell loss for framed data varies depending on the mix of framed data and to a lesser extent on the transmitter buffer as is shown in Fig 6. The cell-loss rate for framed data can be decreased to whatever value is required by either backing off the framed data mix or increasing the transmitter buffer size. However as the percentage of framed data gets toward 100 percent the effect of the buffer is small and the graph becomes almost vertical meaning that either all the data gets through or gets lost. It becomes important to find the value of mix of framed data mix that allows all the framed data to get through without error. For both bursty error and Gaussian noise the buffer required for no framed data loss is shown in Fig 7. As expected, as the percentage of framed data increases towards 100 percent the buffer size required for no loss increases faster than exponential. It is therefore apparent that the best scheme is to have a reasonable mix of framed data and real-time traffic.

Gaussian noise introduces only single-frame errors for bit error rates lower than 1E-5. Hence the Gaussian error model on its own will not be sensitive to transmitter buffer size. This can be seen by comparing how the bursty error and the Gaussian error vary the framed data loss with transmitter buffer size, as shown in Fig 8. The variation on loss by varying the transmitter buffer size is almost negligible in the Gaussian-error case compared to the large variation in the burst-error case. To combat Gaussian errors a buffer the same size as the time-out buffer will suffice.

The gain of getting no loss for the framed data cells has a penalty side by having to retransmit the ones in error. If there is no excess capacity, as is assumed in this article, then there is going to be loss in the real-time cells. To see the effect of this loss, a plot of real-time cells, or voice cells, lost due to burst events for various transmitter buffer sizes and against the framed data mix is shown in Fig 9. In the burst event it is possible to lose all the voice cells if the mix of framed data is high enough. This means that we have run out of bandwidth to retransmit. However this loss when averaged out over the day is of the order of 1E-3. Also the loss is concentrated into intervals of seconds. There may be schemes to lessen the effect of the loss of this voice and spread it out more evenly. However if the critical objective is not to lose framed data this is the price that may have to be paid. Also the size of the transmitter buffer has almost no effect. This is because when the delay expected in the transmitter buffer is more than 50 ms, the real-time cells are discarded. The loss of the voice cells is dependent almost entirely on the error statistics of the model and not on the retransmission model.

The frame size is an important decision and the optimum value depends on the error rate of the link as well as the distribution of errors. Small frames have high protocol overhead but have the advantage that when an error occurs only a small number of bits are lost. Large frames have lower protocol overhead but when one is lost a large number of bits are lost. It is really the Gaussian error that determines the size of the frame that should be used. The burst errors affect the large and small frames almost the same, because a number of continuous frames will be in error. Throughput when the Gaussian error rate is varied between 1E-5 and 1E-6 and the percentage of framed data varied between 50 percent and 80 percent is shown in Fig 10. What is noticed is that for the lower mix of framed cells, for example 50 percent, the lower frame size is preferred, but for higher percentage of framed data, for example 80 percent, larger frames would achieve higher throughput. There is a need to compromise the throughput by picking a fixed frame size. If there is only one link then there could be a optimum-sized frame, or even an adaptive frame size as is already implemented on other applications.

The theoretical throughput limit of any retransmission scheme plotted against the proposed scheme is presented in Fig 11. What is noticed is that we approach the limit almost over the whole range of bit error rates. Therefore we conclude that for these satellite links there is no need to use more complicated retransmission schemes[7] as there is only a small amount of efficiency left to be captured with an increasing cost of transmitters and receiver protocols.

Conclusions

In this paper we have shown that it is possible to use ATM over satellite links. We have proposed to place the error control in the convergence sublayer of the ATM rather than the segmentation and reassemble sublayer. This improves both the efficiency of the protocol and the efficiency of the error detection. The moving of the error control functions is compliant with the ATM standards. We also propose to differentiate the recovery mechanism, or retransmissions, on the service. This allows a guarantee to be given to data services of no loss while not affecting the real time services. While the simulation was for a T1 link, this approach can be incorporated into a data stream that is part of a larger link, such as a 45 Mb/s link.

Acknowledgments

The authors wish to express their appreciation to Dr. Michael Chelian, JPL, who investigated the error distributions on the DSN. They also express sincere thanks to Dr. Julia George, JPL, and Dr. Ed Upchurch, JPL, who helped with many useful and helpful comments on the SES/workbench. The first author was on leave from

Electronic Engineering, Dublin City University, Ireland, while this work was being done and would like to acknowledge Mr. Peter Colaluca, of SES/Europe, for support of the SES/workbench and wishes to further express his thanks to Prof. Charles Mc Corkell, Dublin City University, for his continuing support and encouragement.

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

[1]de Prycker, M., *Asynchronous Transfer Mode: Solution for Broadband ISDN,* 2nd ed., Ellis Horwood, 1993, pp.102-145.

[2]Lane, J., *Asynchronous Transfer Mode: Bandwidth for the Future,* 1st ed., Telco, 1992.

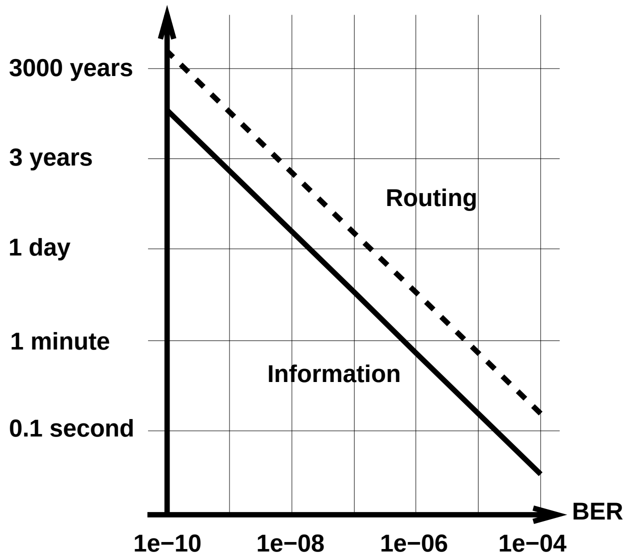
[3]Chow, E. T., and Markley, R. W., "Asynchronous Transfer Mode Link Performance Over Ground Networks", *TDA Progress Report 42-114, JPL,* Aug. 1993, pp. 185-189.

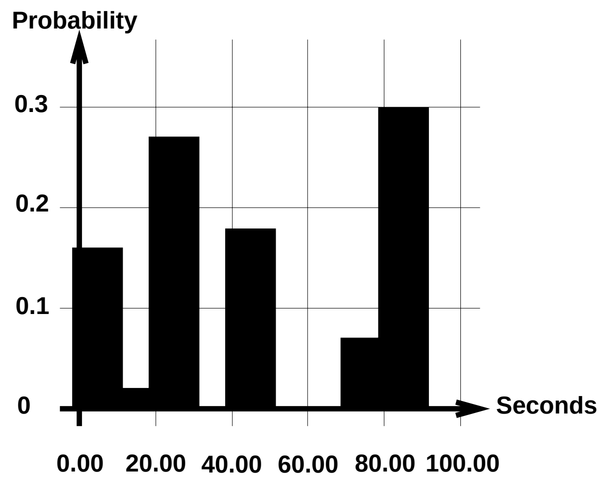
[4]Biersack, E. W., "Performance Evaluation of Forward Error Correction in an ATM Environment", *IEEE J. Select. Areas Commun.,* vol. 11, No. 4, 1993, pp. 631-640.

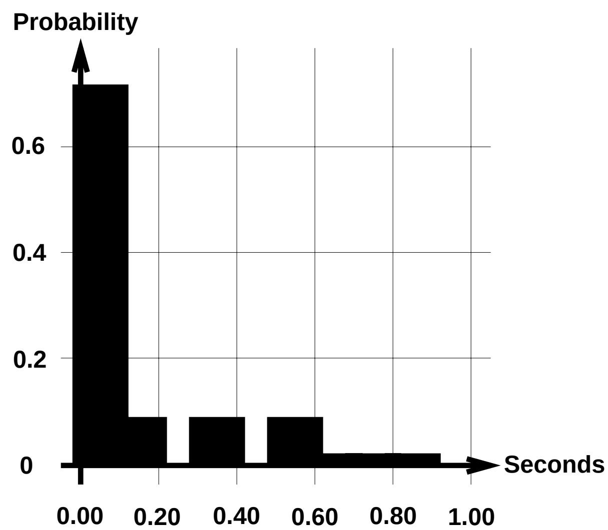
[5]de Prycker, M., Peschi, R., and van Landegem, T., "B-ISDN and the OSI protocol reference model", *IEEE Network,* vol. 7, No. 2, 1993, pp. 10-18.

[6]Bertsekas, D., and Gallagher, R., *Data Networks,* 2nd ed., Prentice Hall, 1992.

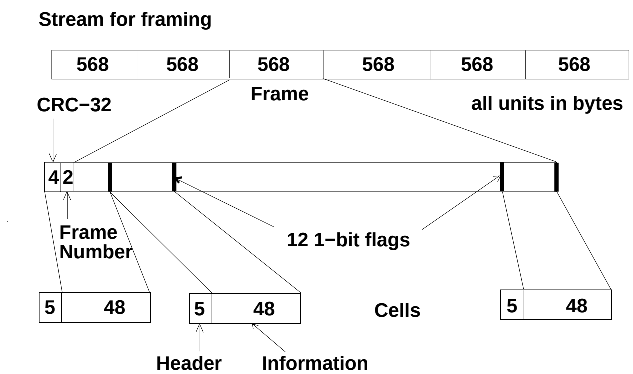
[7]Schwartz, M., *Telecommunication Networks: Protocols, Modelling and Analysis,* 1st ed., Addison-Wesley, 1987, pp.119-160.

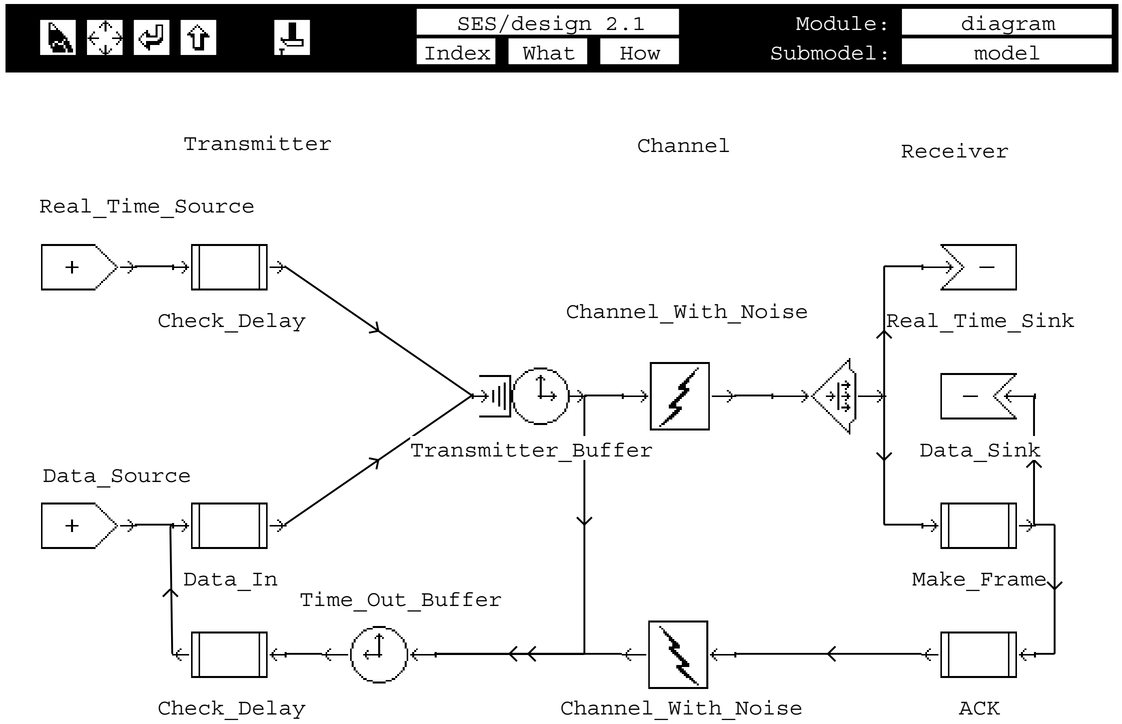

3000 years
3 years
1 day
1 minute
0.1 second
Routing
Information
BER
1e−10
1e−08
1e−06
1e−04


Probability
0.3
0.2
0.1
0
Seconds
0.00
20.00
40.00
60.00
80.00
100.00

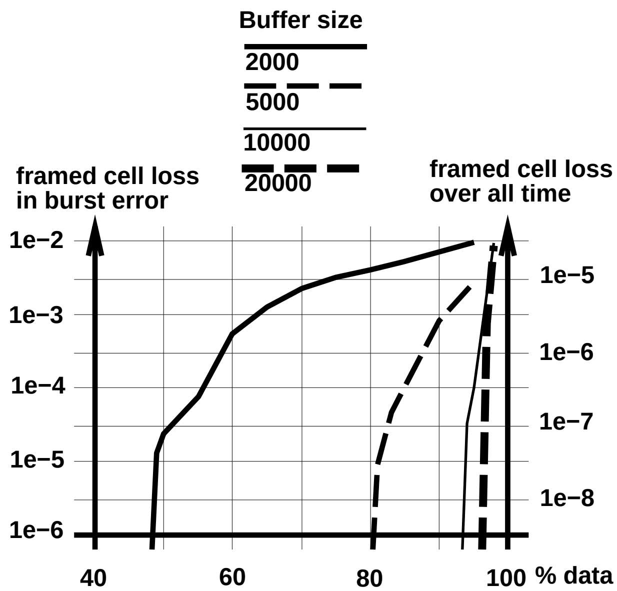


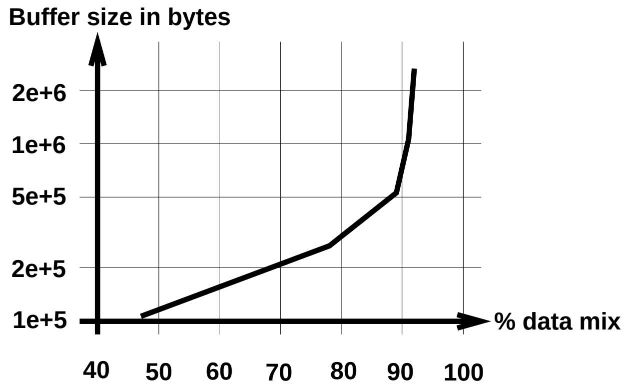
Probability
0.6
0.4
0.2
0
Seconds
0.00
0.20
0.40
0.60
0.80
1.00

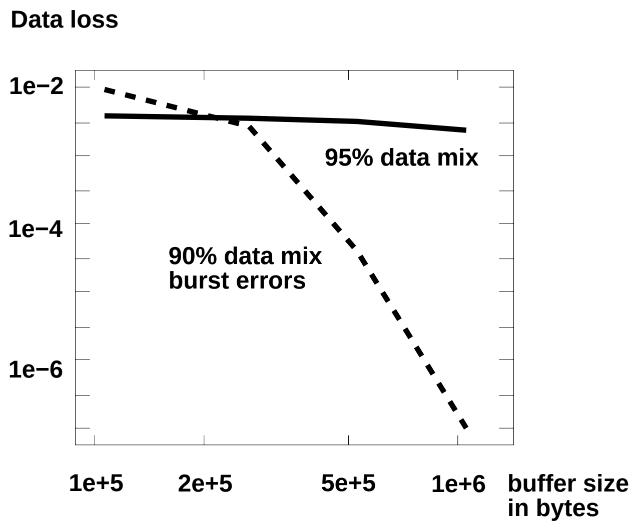

Stream for framing
568
568
568
568
568
568
Frame
all units in bytes
CRC−32
4
2
Frame
Number
12 1−bit flags
5
48
5
48
Cells
5
48
Header
Information

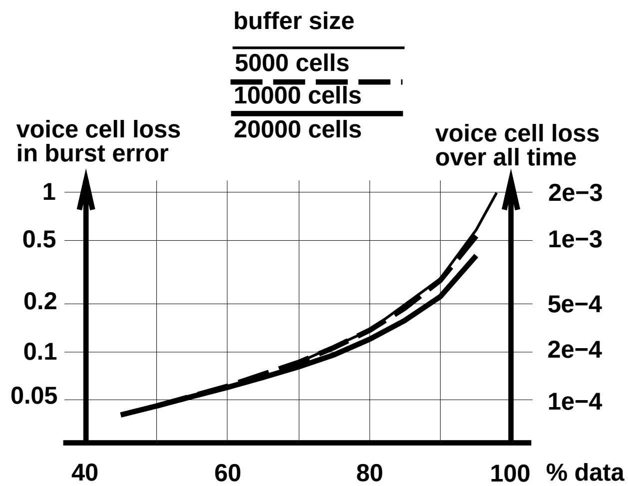


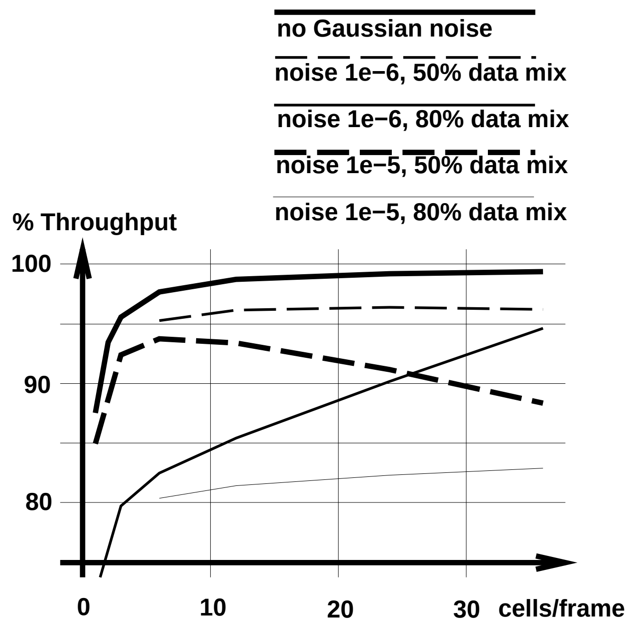
SES/design 2.1
Index
What
How
Module:
diagram
Submodel:
model
Transmitter
Channel
Receiver
Real_Time_Source
Check_Delay
Channel_With_Noise
Real_Time_Sink
Transmitter_Buffer
Data_Sink
Data_Source
Data_In
Make_Frame
Time_Out_Buffer
Check_Delay
Channel_With_Noise
ACK


Buffer size
2000
5000
10000
20000
framed cell loss
in burst error
framed cell loss
over all time
1e−2
1e−3
1e−4
1e−5
1e−6
1e−5
1e−6
1e−7
1e−8
40
60
80
100
% data

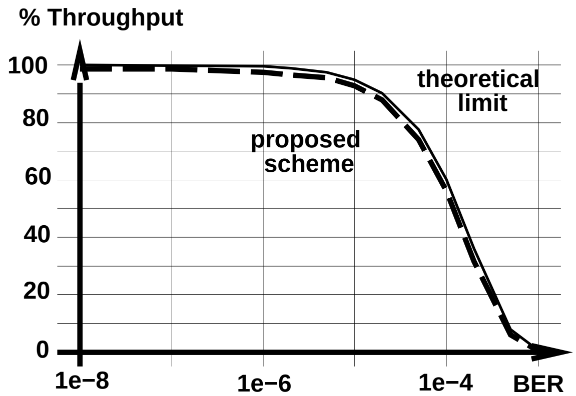

Buffer size in bytes
2e+6
1e+6
5e+5
2e+5
1e+5
% data mix
40
50
60
70
80
90
100


Data loss
1e−2
1e−4
1e−6
95% data mix
90% data mix
burst errors
1e+5
2e+5
5e+5
1e+6
buffer size
in bytes


buffer size
5000 cells
10000 cells
20000 cells
voice cell loss
in burst error
voice cell loss
over all time
1
0.5
0.2
0.1
0.05
2e−3
1e−3
5e−4
2e−4
1e−4
40
60
80
100
% data



no Gaussian noise
noise 1e−6, 50% data mix
noise 1e−6, 80% data mix
noise 1e−5, 50% data mix
noise 1e−5, 80% data mix
% Throughput
100
90
80
0
10
20
30
cells/frame


% Throughput
100
80
60
40
20
0
1e−8
1e−6
1e−4
BER
theoretical limit
proposed scheme