PERFORMANCE EVALUATION OF A MULTI-LEVEL ALLOCATION ALGORITHM FOR VBR TRAFFIC OVER A GEOSTATIONARY SATELLITE*

Nedo Celandroni, Erina Ferro, Francesco Potortì
Area della ricerca CNR – CNUCE - via Vittorio Alfieri 1 – I-56010 Ghezzano, Pisa
{N.Celandroni,E.Ferro,F.Potorti}@cnuce.cnr.it

KEYWORDS: satellite communications, channel access protocol, multilevel allocation, VBR traffic, performance evaluation

ABSTRACT

In this paper we present VnL-TDMA, a new MAC (medium access control) protocol for accessing in TDMA a satellite channel shared among various earth stations which transmit mixed traffic (real-time and non-real time). We concentrate on the analysis of the real-time allocation for variable bit rate (VBR) traffic, and analyse its performance using a synthetically generated VBR traffic modelled on the basis of the statistical parameters of an MPEG-2 coded movie in VBR mode. The peak traffic rate of the movie is almost four times higher than its mean rate, which means that trivial peak rate allocation would waste a lot of the satellite channel bandwidth, while VnL makes unused bandwidth instantaneously available to all earth stations to enable them to transmit best-effort traffic.

1 INTRODUCTION

The medium access control (MAC) protocol used to access a satellite link must be able to guarantee both high link utilisation and low delay transmission for the variable bit rate (VBR), real-time data. In particular, real-time digital video needs a high transmission bandwidth even after compression, it must be sent with a minimal delay, and it cannot tolerate a high error rate. Dedicated transmission lines, that are a good choice for constant bit rate (CBR) traffic, are underused for VBR traffic, especially MPEG-2 VBR traffic for which the peak/mean ratio computed over a group of pictures (GOP) may be as high as 6. On the other hand, current general purpose networks are designed for best-effort traffic.

This paper proposes a MAC protocol for shared satellite channels which is able to provide good channel efficiency for mixed traffic, while still guaranteeing delay and bandwidth for MPEG-2 VBR traffic. In our scenario, various earth stations share in TDMA a geostationary satellite link, each earth station acting as a concentrator of both real-time and best-effort traffic. A centralised MAC protocol, run by the *channel dispatcher,* allows the satellite network to be shared according to the allocation requests issued by the stations. When an earth station realises that the current allocation is not sufficient for sending the traffic coming from the terrestrial network, it asks the channel dispatcher for a larger share of the channel bandwidth, and it queues the incoming traffic while waiting for its request to be granted. Requests to the channel dispatcher are made separately for real-time and non real-time traffic, and the channel dispatcher uses different algorithms to allocate bandwidth for the two types of traffic. The bandwidth allocation algorithms used for non real-time traffic are not described in this paper; any of those described in (Celandroni et al. 1992; Zein et al. 1995, Dornier 1992, Jahangir and Le-Ngoc 1994, Le-Ngoc and Krishnamurthy 1995) can be employed.

This paper focuses on the allocation algorithm for the VBR real time traffic, it describes the simulation environment used to obtain the performance evaluation, and it presents some of the simulation results obtained.

2 THE ALLOCATION ALGORITHM USED FOR REAL-TIME VBR TRAFFIC

In (Celandroni et al. 1997) we presented an allocation algorithm for real-time traffic, with centralised control, based on two levels of bandwidth allocation for VBR traffic (V2L-DA/TDMA). Now we present an extension of the V2L algorithm to *n* levels of allocations, which we call VnL, and we demonstrate that the efficiency of the channel utilisation increases with the number of levels in which the bandwidth is divided.

* Work supported by the Consiglio Nazionale delle Ricerche under the 5% Multimedia Programme

Three parameters (*Amin*, *Amax*, *nlev*) define the minimum throughput, maximum throughput (booking) required by the application, and the number of allocation levels, equally spaced in the range [*Amin*, *Amax*], respectively, as illustrated in Figure 1. The channel dispatcher accepts a new allocation request for real-time traffic only if the sum of the new and the outstanding bookings does not exceed the percentage of the channel bandwidth which is dedicated to real-time traffic. This threshold can be tuned to avoid starvation of non real-time traffic. However, with the use of many allocation levels, this problem is not likely to be an issue, because most of the time the space allocated for real-time traffic is much less than the space booked. Moreover, any channel space allocated to a station for real-time data and not used, can be used by that same station to send its non real-time traffic. The relationship between *Amin* and *Amax* depends on the type of real-time application that generates the request. If *Amin* = *Amax*, the request comes from a CBR application. In the following, we study the case where *Amin* is less than *Amax,* i.e. the request relates to a VBR application. For each VBR flow entering a station, the station books a bandwidth *Amax*. Once the allocation has been granted, the station keeps measuring the throughput of the flow and requests an allocation equal to one of the *nlev* allocation levels. The bandwidth booked but not allocated is managed by the channel dispatcher to satisfy requests for non real-time traffic allocation coming from any station. The earth stations measure the input traffic at each TDMA frame, assuming that an allocation is received in each frame. In order to compute the correct allocation level to request, each station keeps two counters for each VBR flow, which we call positive and negative virtual queue, *pvq* and *nvq*, respectively. The *pvq* keeps track of the volume of data that would be queued at the station if the requested allocation were granted immediately. The *nvq* is the cumulative unused allocation space that would be wasted if the allocation were one level below the requested one.

Let I_i be the volume of input traffic to a station for a VBR flow during frame *i*, let A_{i-1} be the last requested allocation level, and let pvq_i and nvq_i be the virtual queues for the current frame. At frame *i*, the virtual queues are updated as follows:

Astep = (Amax-Amin) / (nlev-1);

$pvq_i = \max(0, pvq_{i-1} + I_{i-1} - A_{i-1})$;

$nvq_i = \min(0, nvq_{i-1} + I_{i-1} - A_{i-1}\text{-step})$ if A_{i-1} > Amin, else 0.

When the data throughput is between the current allocation level and the one below it, both virtual queues are 0, otherwise one and only one of the two virtual queues is different from 0. If pvq is positive, then a request is made for a higher allocation level, while, if nvq is negative, a request is made for a lower allocation level. The value of the allocation request for the current frame, A_i, is:

$$A_i = \min(\text{Amax}, A_{i-1} + [\text{ceil}(pvq_i / (T\ \text{Astep})) + \text{floor}(nvq_i / (T\ \text{Astep}))]\ \text{Astep})$$

where *T* is the time interval between successive requests, *floor* gives the greatest integer not greater than its argument, and *ceil* gives the smallest integer that is not smaller than its argument. All the requests are made on the basis of the last allocation level requested, not on what is currently granted by the channel dispatcher. This means that queues build up at the station when the input traffic increases, while unused allocation will be granted to the station when the input traffic decreases.

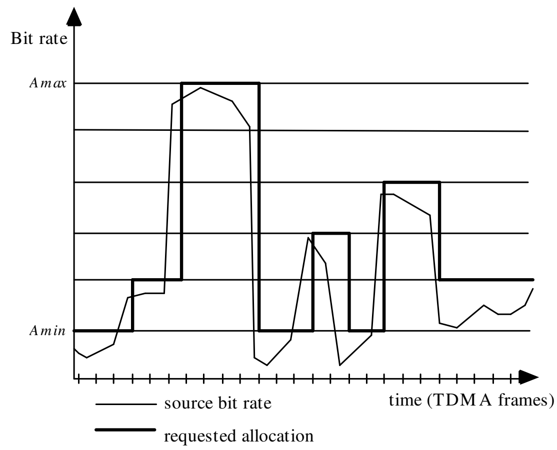

Figure 1. Allocation levels for an input traffic with varying throughput

Note that a real implementation of this algorithm would have to account for possible communication errors between the earth station and the channel dispatcher. In fact, VnL as described has no feedback mechanism that monitors the input queue, but only considers what we have called virtual queues, that is, what the input queue would be if the granted allocation were equal to the requested one.

3 SIMULATION ENVIRONMENT

In order to reliably evaluate the performance of VnL, we fed it with a synthetically generated MPEG-2 data. We assume that a GOP is made up of twelve 40ms long frames, and that the output of the MPEG-2 coder is smoothed at the GOP level, i.e., that after the coder there is a 480ms pre-buffering before transmission on the network. To generate VBR traffic, we considered the trace-driven transmission of the movie "The sheltering sky" produced by an MPEG-2 encoder. The model of the MPEG video source used is sketched in (Celandroni et al. 1997), where it was already used, and described in (Chimienti et al. 1996; Conti and Gregori 1997) in more detail, where the accuracy of this model was investigated. Figure 2 shows the probability density function of the generated throughput. We run the simulations using Fracas[3], a high speed, lightweight simulator useful for simulating framed channel allocation schemes (Celandroni, Ferro and Potortì 1999). In order to obtain statistically significant results, we also wrote a module, using the Python language, which implements independent replications by repeatedly calling Fracas with different seeds for the random number generators, until the requested confidence interval for the results has been obtained.

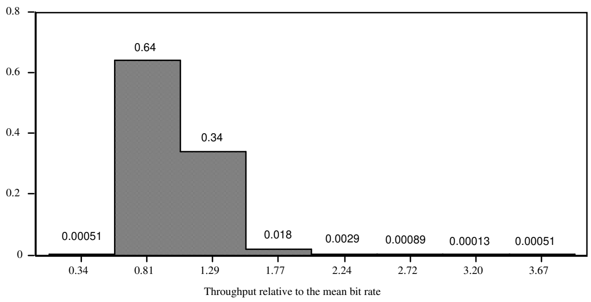

Figure 2. Probability density function of the bit rate produced by the Markov generator

The most significant specifications of the simulation runs are:

- a TDMA system is used, where the frame length is 20 ms, the virtual queues are probed once per frame, and an allocation request is issued at every frame;
- the VBR source is simulated by using the Markov generator with a mean and a maximum throughput equal to 3 Mbps and 11.7 Mbps, respectively;
- a single traffic station is loaded with the VBR traffic generator;
- the statistics collected include mean unused space, maximum packet delay, and packet delay quantiles of 0.9, 0.99 and 0.999;
- the minimum allocation *Amin* has been varied between 1 and 5 Mbps, in steps of 0.1 Mbps, while the maximum allocation *Amax* lies in the range between 5 and 9.5 Mbps, in steps of 1.5 Mbps;
- the number of allocation levels *nlev* has been set to 2, 3, 4, 10 and 100; the 100 level case is a practical approximation of a continuous variation in the allocation level.

All the simulation results were obtained with a 95% confidence level. The confidence intervals for the unused allocation were ±2%, while the confidence intervals for the delays were ±5%.

4 SIMULATION RESULTS

The performance of the proposed method was evaluated by measuring two statistics: the *end-to-end packet delay*, and the *unused allocation space*. The end-to-end packet delay is variable because of variable queuing delays. The minimum delay is set to 250 ms, which is the conventional round-trip time of a geostationary satellite. The quantiles of the packet delay capture the dynamic behaviour of VnL, and show how the variability of the input traffic affects its end-to-end delay. The unused allocation space is the satellite link share allocated to a station for transmitting VBR traffic and used by the station for transmitting non-real time traffic, if any. This quantity gives the efficiency of the allocation method, and should be made as small as possible. It is mostly a steady state characteristic. In fact, a similar characteristic was studied in (Celandroni et al. 1997) by purely analytical methods, yielding the same results as those obtained here for VnL when the number of levels is 2.

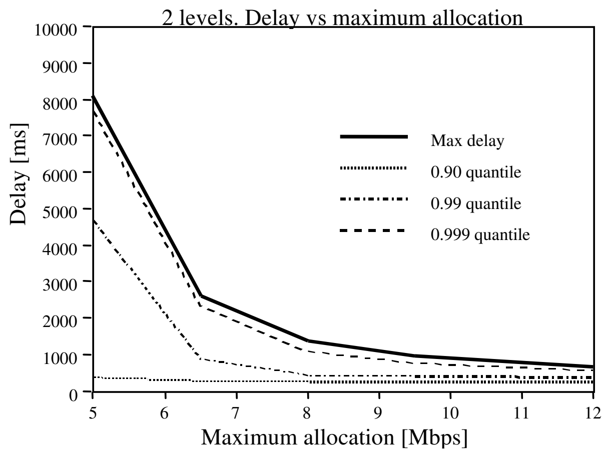

Figure 3. Maximum data delay vs. maximum allocation. Two allocation levels with a minimum allocation equal to 3 Mbps.

The packet delay is essentially the sum of three addenda. The first one is simply the latency of the satellite link. The second is the allocation delay when

[3] FRAmed Channel Access Simulator, developed at CNUCE/C.N.R. Pisa (I).

switching between levels in response to variations in the throughput of the VBR flow entering the earth station. During the two round trip times between the request for a larger allocation and the relative authorisation (which is always granted, because the bandwidth has been booked in advance), the traffic is enqued at the station, and the queue is emptied only after the allocation delay. There is always this effect when going up levels, and it depends on the *Amin*, *Amax*, and *nlev* parameters. The third cause of delay is the insufficiency of the booked allocation, *Amax*, which we set up to a set of values ranging from 5 Mbps to 9.5 Mbps, lower than the peak VBR data throughput, which is 11.7 Mbps. Again, data spends some time in the queue, waiting to be transmitted. This third effect, which decreases as *Amax* increases, disappears when the maximum allocation is equal to the peak throughput of the input traffic, and it is independent both of the minimum allocation *Amin* and of the number of levels *nlev*.

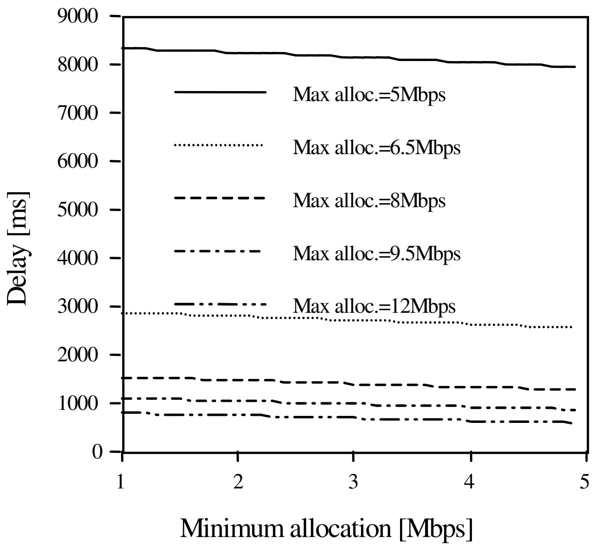

Figure 4: Maximum data delay vs. minimum allocation. 100 allocation levels with maximum allocations equal to 5, 6.5, 8, 9.5, and 12 Mbps.

In order to eliminate this effect, and to examine the switching delay more closely, we also made some simulation runs with a maximum allocation of 12 Mbps, which is greater than the peak. Figure 3 shows the relationship between the maximum data delay and the maximum allocation when the minimum allocation is 3 Mbps. This figure is the same for any of the listed levels, thus showing that the delay is independent of the number of levels. In 99% of cases a maximum allocation of 8 Mbps is sufficient to fit the input traffic, since the data delay is about the same as that obtained with a maximum allocation of 12 Mbps, while a maximum allocation of 6.5 Mbps, or lower, satisfies the input traffic only in 90% of cases.

Figure 4 plots the maximum data delay versus the minimum allocation for 100 levels. The runs were made with a maximum allocation of 5, 6.5, 8, 9.5, and 12 Mbps, with the same confidence intervals as the previous results. The data delay also has a very slight dependence on the minimum allocation, and in fact the delay curves are basically flat, whatever the number of levels, with maximum variations of about 5%.

When the booked allocation *Amax* is greater than the peak VBR traffic throughput, the delay is influenced by the level switch delay alone, while in the cases presented above the delay is mainly influenced by *Amax*, and only secondarily by the level switches.

A characteristic common to the plots of the delays obtained by varying the minimum allocation is the presence of a slight raising in the central part, which is highlighted in Figure 5.

This effect is most clearly visible with a small number of allocation levels, and is due to a greater frequency of level switches, because of the characteristics of the input traffic. Looking at the figures that show the delay vs. the minimum allocation, and, more clearly (as different scales are used) in the figures with the maximum allocation equal to 12 Mbps, we can see that the raising point corresponds to *Amin* values such that the peaks of the input distribution do not fall entirely between two allocation levels. This effect is justified by the fact that the short-term autocorrelation of the output states of the Markov chain is quite high. In fact, when in the second state, whose steady state probability is 0.64, the probability of staying in the same state in the next GOP is 0.93; when in the third state, whose steady state probability is 0.34, the probability of staying is 0.85. Thus, for a low number of levels and a low *Amin* (lower than 1.7 Mbps), the allocation requested almost always corresponds to one of the highest levels, thus decreasing the number of level transitions and consequently the end-to-end delay. When the minimum allocation increases, so does the probability of transition, and its influence on the delay. Such an influence again decreases for a greater *Amin*. Notably, this effect tends to disappear when the number of levels is high.

The maximum delay lines do not show the same risings, and decrease as *Amin* increases. This is due to a situation that only occurs at the beginning of the movie. In fact, the implementation of the VnL-DA algorithm assigns *Amin* as the first allocation, while the movie (and thus our model) starts with a throughput which is very close to the maximum. Therefore, at the beginning of the simulation there is always a very high delay, which is inversely proportional to the minimum allocation *Amin*. When *Amin* is equal to the peak throughput of the input traffic, the maximum delay is extinguished, in the sense that it becomes equal to the round trip time.

It is possible to estimate the unused space by using the statistical properties of the synthetic input traffic. It is well approximated by the difference between the input traffic and the relative allocation request, weighted with the probability of a given input traffic value. Denoting by t the input throughput, we have

$$u = \int_{\min(t)}^{\max(t)} [A(t) - t] P(t)$$

where $A(t)$ is the allocation relative to a given value of the input throughput.

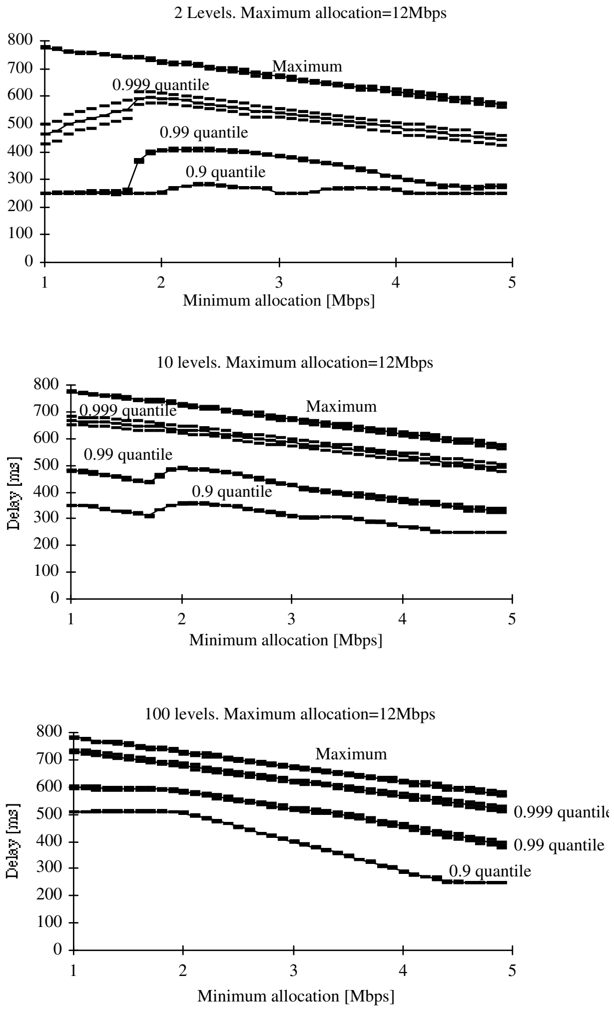

Figure 5. Maximum data delay vs. minimum allocation with 2, 10, 100 allocation levels.

The unused allocation space is plotted in Figure 6 for various *Amax* values, as a function of *Amin*.

Computed values are depicted as solid lines, while the simulation results are reported as the upper and lower bounds of 95% confidence intervals. The matching is excellent, in spite of the analytical model being only an approximation of the allocation algorithm. While the number of levels is low, the unused space very much depends on the booked allocation, which is lower for low *Amax*. However, the above considerations on the packet delay mean that the cases where *Amax* is less than 6.5 Mbps should be discarded.

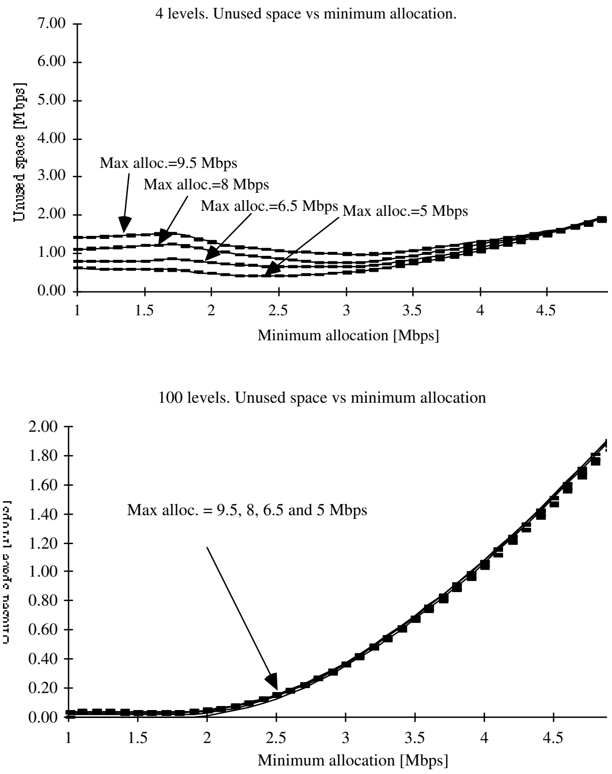

Figure 6. Four and hundred levels. Unused space vs. minimum allocation for various values of the maximum allocation with 95% confidence intervals.

The curves flatten with increasing numbers of levels. Since the point of minimum is strongly dependent on the input distribution, it is preferable to have flat curves, because it makes the system performance less dependent on the particular MPEG model used to tune it.

5 CONCLUSIONS

The simulation study shows that neither the number of levels *nlev* nor *Amin* play a significant role as far as the packet delay is concerned. Moreover, setting the booked bandwidth *Amax* to values greater than 8 Mbps does not produce any appreciable improvement. Therefore, the major gain of using many levels is not so much in the improvement of the optimal point but in the greater flatness of the lines that describe the unused space. At higher numbers of levels the dependence on *Amax* tends to disappear, and the efficiency of the algorithm improves as *Amin* is smaller. This means that the dependence on the characteristics of the input generator is practically lost. For 100 levels, which is an approximation of an infinite number of levels, the unused space tends to zero for low *Amin*, as was expected. The channel efficiency is very high, with very little unused allocation space when many allocation levels are used, and virtually no allocation waste when the number of levels is in the order of 100.

We expected to find an optimum number of allocation levels, because we thought that the queuing delays introduced by allocation level switching would have made it impractical to use a high number of levels. Indeed, queuing delays with hundred levels are higher than delays with ten levels, but the relative difference is so small that it is practically negligible. In short, it is convenient to use as many allocation levels as possible, at least up to a value of around 100. If using a high number of allocation levels is impossible because of link layer limits on the minimum allocable unit, a minimum number of four levels should be used, which provides better performance and less dependence on the input pattern than the V2L method proposed in (Celandroni et al. 1997).

REFERENCES

Celandroni, N.; E. Ferro; N. James; and F. Potortì, 1992, "FODA/IBEA: a flexible fade countermeasure system in user oriented networks", *International Journal of Satellite Communications*, Vol. 10, No. 6, pp. 309-323.

Celandroni, N.; M. Conti; E. Ferro; E. Gregori; and F. Potortì, 1997, "Bandwidth assignment on a satellite channel for VBR traffic", *International Journal of Satellite Communications*, Vol. 15, pp.237-246.

Celandroni, N.; E. Ferro; and F. Potortì, 1999, "A simulation tool to validate and compare satellite TDMA access schemes", *Telecommunication Systems*, Vol. 12, No. 1, pp. 21-37.

Chimienti, A.; M. Conti; E. Gregori; M. Lucenteforte; and R. Picco, 1996, "MPEG-2 Variable Bit rate coding algorithm: analysis and modeling", in *SPIE proceedings*, Vol. 2952, N. Otha editor, pp.597-607.

Conti, M. and E. Gregori, 1997, "Traffic characterization and bandwidth allocation for MPEG-2 VBR Video Traffic", in *Proceedings of GLOBECOM'97*, pp. 443-447.

Dornier Deutsche Aerospace, 1992, "Advanced Business Communications via satellite, System Description".

Jahangir I. M. and Le-Ngoc, T., 1994, "Performance analysis of Combined Free/Demand Assignment Multiple Access (CFDAMA) Protocol for Packet Satellite", in *Proceedings of ICC 94*, pp. 869-873.

Le-Ngoc, T. and Krishnamurthy, S.V., 1995, "Performance of Combined Free/Demand Assignment Multi-Access (CFDAMA) Protocol with Pre-Assigned Request Slots in Integrated Voice/Data Satellite Communications", in *Proceedings of the International Conference on Communications, ICC 95*, pp. 1572-1576.

Zein, T.; G. Maral; T. Brefort; and M. Tondriaux, 1995, "Performance of the Combined/Fixed Reservation Assignment (CFRA) scheme for aggregated traffic", in *Proceedings of the COST-226 Final Symposium*, pp.183-198.