

# Performance Evaluations of Partial Order Connections

M. Fournier, C. Chassot, M. Diaz, A. Lozes

LAAS-CNRS

7 avenue du colonel Roche 31077 Toulouse Cédex, France,  
fournier, chassot, diaz, alozes@laas.fr

## Abstract

Technological improvements in Computer and Telecommunication areas lead to the emergence of new distributed multimedia and co-operative applications, whose features and requirements are both multiple and diversified. In order to tackle their needs, two major approaches are currently developed, one arguing for an improvement of the application software, the other one promoting a sophisticated network support. Following this last approach, a new concept of connection, *the partial order connection* (POC), has been proposed and formalized. This paper provides simulation results of the POC concept obtained using the OPNET software tool. Particularity, it is shown how a partial order / partial reliability management at the transport layer impacts on end to end transit delay and on the memory used at both sending and receiving transport entities.

## Keywords

Multimedia Application, Transport Service, Transport Protocol, Partial Order Connection, Performance Evaluation, OPNET

## 1 INTRODUCTION

During the last decade, technological improvements in Computer and Telecommunication areas led to the emergence of new distributed *multimedia* and *co-operative* applications, including transmission and processing of all data types (text, fixed or animated image, voice, video sequences, etc.). Among the requirements of such applications, the following ones are now of importance:

- high speed transmission links are obviously needed;
- temporal constraints have to be enforced for interactive communications such as visioconferencing systems;
- multimedia synchronization is required: exchanged data have to be orchestrated the ones with the others;
- finally and at the opposite of file data transfers, distributed multimedia applications tolerate an imperfect transfer of the data they involved, without being unacceptable for users.

A full analysis of distributed multimedia applications requirements may be found in [23].

**Current solutions and their limits:** Recently, many studies have been performed to design light weight transport protocols (such as NETBLT [6], VMTP [4] or XTP [17]) more suited to support multimedia data transfers than TCP or TP4 protocols; however, these proposals do not provide a sufficient solution with respect to multimedia requirements. Usually, multimedia synchronization issues (both spatial and temporal ones) are not addressed and temporal constraints management mechanisms are implementation dependent. A new generation of high speed, multimedia protocols has to be designed on top of high speed links using ISDN, Internet protocols or ATM. Presently, several communication architectures are envisaged within different projects which may be classified as follows.

**The "Application Aware Networking" approach:** Up to now, two major architectural approaches have been proposed to design new distributed communication systems; both extend the *Application Layer Framing* (ALF) concept introduced in [7] that argues the interest of Application Data Units (ADU) preservation at the communication system level:

- the first approach assumes that the network software has to be as simple as possible without providing any Quality of Service (QoS) guarantee. The application is supposed to be the most appropriate to adapt to network fluctuations by integrating in its software the most adequate transport mechanisms. This *Network Aware Application* (NAA) approach is mainly supported by the Internet community, and is currently pursued within the Esprit project HIPPARCH. Recent investigations and results are given in [5] [14] [13].
- at the opposite, the second approach suggests to increase the networking system complexity to handle new requirements defined at the applicative level. The major interest of this approach is to make the user software simpler at the cost of a more sophisticated network support. This *Application Aware Networking* (AAN) approach has been or is currently pursued within several projects: OSI 95 [8], RACE CIO [16], BERKOM [9], QoS-A [3], TENET [15] and CESAME [11].

This paper, which follows the AAN approach, will evaluate the performance of a new kind of connection, the *partial order connection* (POC), introduced in [1] and [10]. A POC defines a new transport QoS related to 'order' and 'reliability' parameters; when implementing this new concept, a transport protocol allows bandwidth, memory and transit delay savings. The major part of this paper aims at showing (in a simulation environment) the interest of the POC concept for two parameters: the number of used buffers and the applicative transit delay. This paper is divided into five sections. Basic principles of the POC concept are first recalled in section 2. Tested partial order transport protocol is composed of different mechanisms which are briefly exposed in section 3. In section 4, the communication model (including network support) is presented

first. Simulation context is then exposed: the selected parameters are given and their different values during simulations are detailed; simulation results are finally given and analyzed. Conclusions are stated in section 5.

## 2 PARTIAL ORDER CONNECTION: CONCEPT AND ANALYSIS

At the present time, commonly used transport protocols are either based on the connection-oriented (CO) mode or on the connectionless (CL) one:

- on one hand, CO protocols (such as TCP) provide their users with full reliability and total order, at the cost of increased delay and reduced throughput;
- on the other hand, CL protocols (such as UDP) introduce no increase in transit delay or reduction in throughput but provide neither order nor reliability guarantee.

This classification shows a design gap between CO and CL protocols that suggests a conceptual extension of the classical connection concept, using *order* and *reliability* parameters. This extension has led to *the partial order connection*, for which CO and CL protocols are two special cases. POC has been introduced in [1] and [10]. The basic principles of the POC concept and its suitability for multimedia applications are presented in 2.1. In order to quantify and then compare the complexity of different partial order services, quantification metrics have also been introduced in [1] and [10]; in order to classify the different protocols whose benefits are evaluated in section 4, a brief overview of the used metric is presented in 2.2.

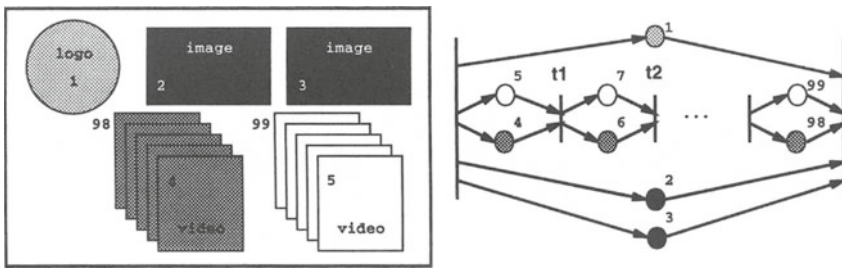
### 2.1 POC Concept

***A conceptual extension of the connection concept:*** A POC is an end-to-end connection that allows its users to define and use for transferring data any partial order/reliability service from no order/no reliability (typically a UDP-like service) to total (sequential) order/total reliability (typically a TCP-like service). In a POC, 'order' and 'reliability' appear as two specific QoS parameters specified by the service user in the connection set-up request. Once known by the sending entity, 'order' and 'reliability' are translated into protocol parameters used to start-up associated protocol mechanisms. In a POC, service data units (SDUs) can be delivered to the receiving user in an order that is different from the sending order: the acceptable difference between the submission sequence and the different but acceptable delivery sequences precisely results from the definition of the selected partial order.

***A concept suitable for multimedia applications:*** Up to now, a few models have been proposed in order to provide a formal representation of temporal constraints in distributed systems; most of them are based on Petri nets [22] [19] [2]. One of this model, the Timed Stream Petri Net (TSPN) model [12] [21], provides a formal description of multimedia synchronization scenarios in asynchronous distributed systems. In this model, a place is associated with a data presentation (a still image or a sound fragment for instance); logical

dependency relationships existing either within a flow or between the different flows of the application are modeled using transitions of the net. The Petri net deduced from a TSPN model provides a logical representation of temporal synchronization constraints, which is a partial order.

Consider for instance the *multimedia* object pictured on the left part of Figure 1; it is composed of different *monomedia* objects (a logo, two fixed images and two video sequences numbered from 1 to 99) which will have to be displayed at the screen of distant user; the picture gives the expected object display at the receiving side. The right part of the figure provides the partial order, say  $P$ , deduced from the application defined TSPN model, that illustrates (among other) that logo 1 and image 2 may be distributed at the receiving side independently the one of the other as they are in parallel both in the display and in the model; logical synchronization constraints within the two video sequences are expressed by the intermediate transitions  $t_i$ .



**Figure 1** Partial order deduced from a TSPN model of a multimedia object

Provided with a  $P$  partial order transport connection, the receiving user may then be delivered monomedia objects (typically SDU) in any sequence consistent with both spatial and temporal synchronization constraints. Moreover, as it is shown through different theoretical examples in [1], these different delivery sequences lead to transfer speed-up and save resources at both sending and receiving sides. Simulations given in section 4 will confirm these first results.

## 2.2 Quantification metric

A simple 'reliability' measure consists (for instance) in the maximum number of losses the protocol may tolerate without recovering. Divided by the number of sent objects during the observation period, this measure is still between 0 and 1. Such an 'order' measure is not obvious and several metrics have been proposed in [1] and [10]. The one we select to keep a representative subset of all protocols (according to order and reliability) is the following one [1]:  $m(Z) = \frac{\ln(e(Z))}{\ln(\text{card}(Z)!)}; \text{ with } Z: \text{ number of objects / period.}$

### 3 PROTOCOL DESIGN

Performance evaluations exposed in section 4 have been realized using a protocol based on a selective positive acknowledgment mechanism. This protocol has been initially modeled using the formal description technic Estelle [18]; the Estelle code may be found in [1].

At the sending side, PDUs are numbered and then buffered in the event of future retransmissions. A timer is associated to each sent PDU. A flow control is also ensured by means of a sending window.

At the receiving side, any received PDU is acknowledged and its "deliverability" with respect to order and reliability QoS parameters is checked; three cases are considered:

- The PDU is deliverable; it is then delivered to the user.
- The PDU is not deliverable (order and reliability constraints are not respected); it is acknowledged and then buffered until these constraints are verified.
- The PDU has already been received (it is a duplicate); it is then rejected.

In addition to order and reliability parameters, two other parameters are managed by the protocol: the sending window length and the retransmission timer. In order to optimize retransmissions management, the value of the timer has to be updated according to the Round Trip Time (RTT) value. When the underlying network only provides a "best effort" service (typically, the present Internet), it is not possible to have a fixed timer value (RTT variations are not predictable and may be very important); however, in order to make our simulations simpler, we fixed the retransmission timer value to  $1.1 \cdot \text{RTT}$ . Sending window size has been fixed to 10 PDUs. Note that these two parameters can be modified for other simulations.

### 4 PERFORMANCE EVALUATIONS

Protocol design and performance evaluations have been made using the OPNET tool [20]. OPNET allows one to design a hierarchical and structured communication architecture (an OPNET model) composed of three different design levels, network, node and process levels, representing for instance an ATM network, a gateway (or a workstation) and its behaviour respectively. In the following, our model is first presented and its parameters are detailed (4.1). The simulation context, i.e. parameters and their values, is detailed in (4.2). Their impact on both sending/receiving buffers and applicative transit delay is exposed and analyzed.

#### 4.1 Model description

At the network level, the model is composed of three parts, the sender, the receiver and the network (see figure 2). The sender is composed of a data

generator and the sending POC entity. The receiver is composed of the receiving POC entity and a data consumer.

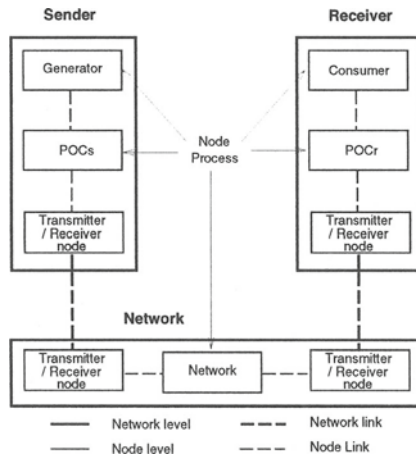


Figure 2 Partial order Model

**Sender parameters:** The sender parameters are the packet generation date and the sending period duration. Time origin is given by the first packet sending date.

**Receiver parameters:** The receiver parameters are 'order' and 'reliability' QoS. Reliability is expressed by a real between 0 and 1, 0 meaning unreliable service and 1 reliable service. Order is expressed by an integer list representing the immediate predecessors of each object (see example given in figure 3). For instance, object 1 of the period (composed of 4 objects) is preceded by 2 objects (predecessor number = 2): -1 et 0 (predecessor list = -1 et 0).

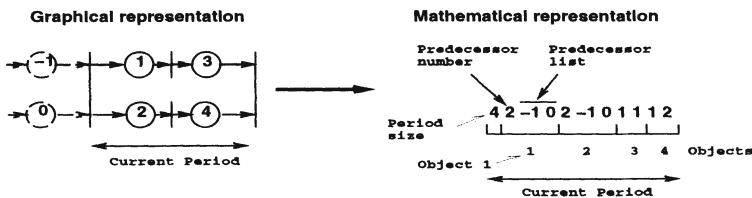


Figure 3 Order representation

**Network parameters:** Two parameters are considered for the network: transit delay and loss rate.

- Transit delay is expressed in milliseconds and represents the time between one outgoing PDU (at the sending side) and one incoming POC PDU (at the receiving side).

- Loss rate is expressed by a real number between 0 and 100. Losses are produced by a random generator; generated numbers are extracted from an uniform distribution between 0 and a maximum value set up by the user.

The protocol allows simulation of different order and reliability with variable transit delay and/or loss rate values. Note that simulations are reproducible because the initial value of the OPNET random generator (used to produce random losses) is a simulation parameter.

### 4.2 Simulation context

The simulation context defines parameters values that have been used in our simulations. The two major parameters are the order and reliability QoS parameters. Other parameters characterize both the network (loss rate, transit delay) and the transport protocol (sending window size, retransmission timer value). Simulation duration is the last defined parameter.

#### (a) QoS

**Partial order:** Ten partial orders have been tested. All are serial/parallel Petri nets compositions of eight objects. Their representation on the order axis given on figure 5 results from the metric  $m$  described in section 2.2. The ten partial orders have been chosen to be equitably distributed on the order axis. Details of these orders are given on figure 4.

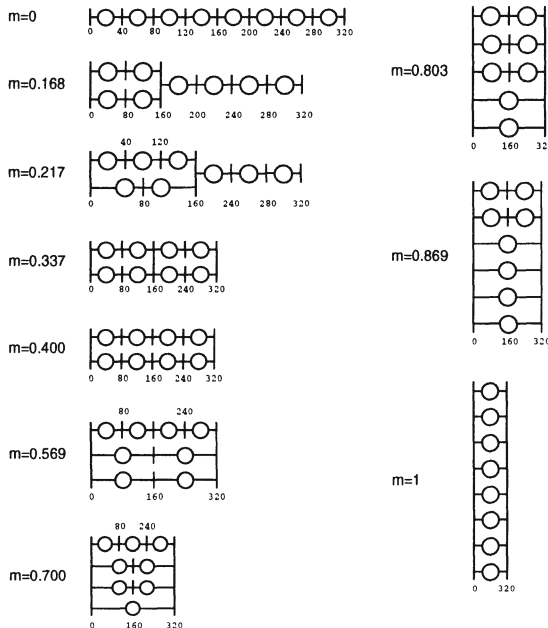


Figure 4 Set of serial parallel compositions

**Partial reliability:** Reliability is represented between 0 and 1, which respectively represents unreliable and reliable service. For instance, a 0.9 reliability indicates that 10% of user data may be lost by the transport service provider. We chose reliability between 0.5 and 1, this variation being sufficient enough to show reliability impact on both buffers and transit delay. Set of tested values is 1, 0.99, 0.95, 0.9, 0.75 and 0.5.

**QoS representation:** As it has been described in section 2.2, it is possible to classify any protocol in a two axis system ('order', 'reliability') such as the one given on figure 5. Protocols which will be evaluated in the following are localized in this figure by means of character "o".

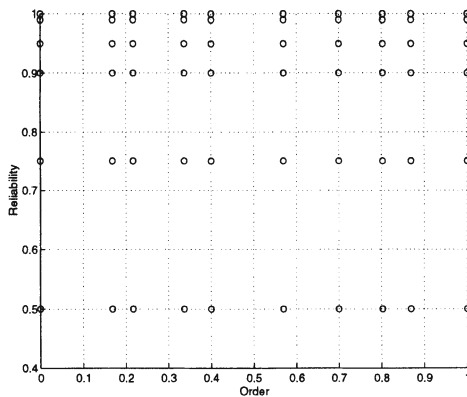


Figure 5 Protocols set

### (b) Simulation parameters

**Transit delay:** Transit delay parameter allows one to simulate different networks. Our goal being first to measure the impact of the QoS parameters, we fixed the transit delay value to 50ms, representing a  $RTT$  of 100ms.

**Loss rate:** Order and reliability impact appears only when losses occur. It is then necessary to introduce them in the network; four loss rates have been tested: 0, 5, 10 and 25%. Even if a network with a 0% loss rate is not realistic, associated simulation results will be given here as a reference level.

**Retransmission timer and sending window size:** Retransmission timer has been fixed to  $1.1 * RTT$ . The sending window size is 10 PDUs, a size which appeared during simulations being sufficient to preserve data flow continuity.

### (c) Simulation duration

An important point to determine before experiment start up is the optimal simulation duration. Indeed, simulations have to be made in a stationary rate, not during a transitory one. On figure 6 where both buffers and transit delay evolution are presented, we see that the transitory phenomena is relatively short compared to the 1000s simulation duration. Illustrated curves have been obtained with maximal order/reliability constraints (that is total order/reliability) and



a 10 % loss rate. We assumed for all curves that the same simulation duration was sufficient to reach 'permanent rate' results.

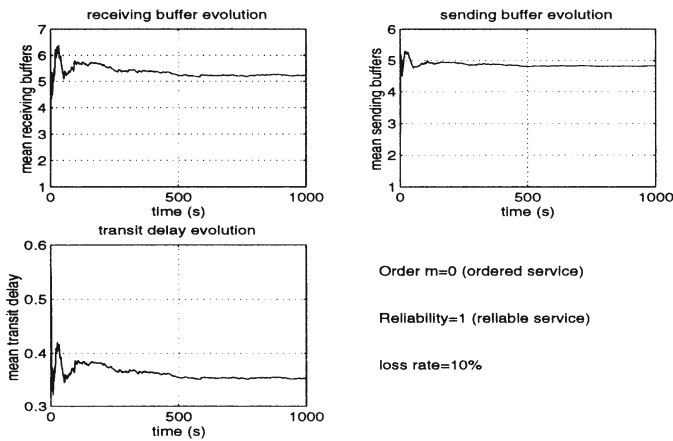


Figure 6 Transitory and permanent rates

### 4.3 Partial Order Connection evaluation

#### (a) Order impact on sending/receiving buffers

In this subsection, evaluation of mean sending/receiving used buffers is computed for each of the ten tested partial orders, depending on four network loss rates (0, 5, 10 and 25 %) and three particular reliability values:

- 0, which represents a totally reliable service.
- 0.95, which represents an unreliable service with a 5% acceptable loss rate.
- 0.5, which represents an unreliable service with a 50% acceptable loss rate.

Sending (respectively receiving) buffers are given on figure 7 (respectively on figure 8). Only 'permanent rate' results are provided on these figures. Each figure is composed of four sub-figures (one for each network loss rate), each of them representing three curves (one for each reliability).

**Sending buffers (figure 7):** As any received or declared lost PDU is acknowledged by the receiving POC entity (out of partial order received PDU are stored), note that for a given (reliability, loss rate) couple, order impact on used sending buffers is null (same used buffers number for each of the ten selected orders). However, it is to be noticed that used sending buffers number depends on the (reliability, loss rate) couple value. This last point will be detailed in subsection (c) ('Reliability impact on sending/receiving buffers').

**Receiving buffers (figure 8):** Order impact on receiving buffers appears to be more significant. Indeed, as any out of order received PDU is stored at the receiving side, the less the order constraints are, the less the receiving buffers are full. This point is clearly shown through the decreasing slope of the different curves. As far as this lastest point is concerned, one can also notice that slope

of the different curves also depends on the (reliability, loss rate) couple value. This point will be analyzed in subsection (c).

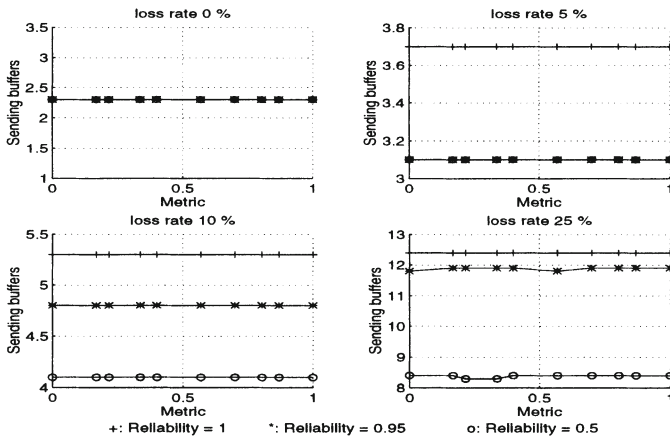


Figure 7 Sending buffers

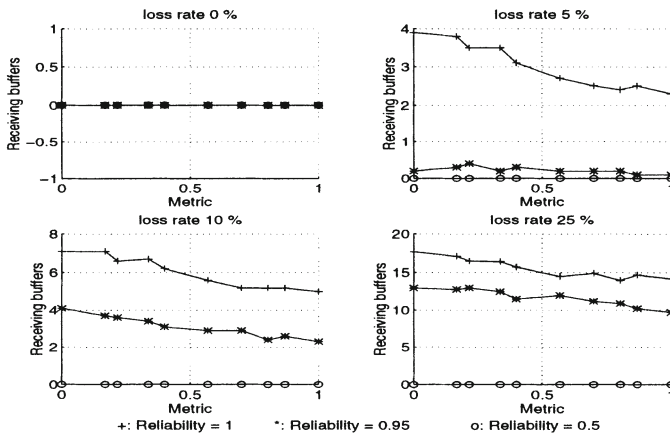


Figure 8 Receiving buffers

**(b) Order impact on the transit delay**

Order impact on the applicative transit delay is shown on figure 9. Except on the first curve (loss rate = 0%), the following conclusions may be stated:

- All curves are decreasing with order (that is when order constraints are decreasing). For instance, for the curve obtained for a 10% loss rate and a 0.95 reliability, a 20% saving on the transit delay can be observed for the partial order whose metric is 0.569 compared with the total order ( $m = 0$ ). Instead of having to buffer an out of sequence (i.e. out of total order) PDU and wait for the arrival of the retransmitted PDU(s) (lost by the network

during their first transfer) that precede it in the total order, the receiving POC entity may deliver such a PDU as soon as it has been received provided its delivery matches the selected partial order.

- Another point has also to be noticed: order impact on the transit delay is less significant when the reliability parameter gets closer to or is greater than the network loss rate (see all the curves obtained for a QoS reliability or the one given for a 0.95 reliability and a 5% loss rate). Indeed, in these cases, lots of retransmissions are avoided (the selected reliability service still being ensured), making the previous point less significant.

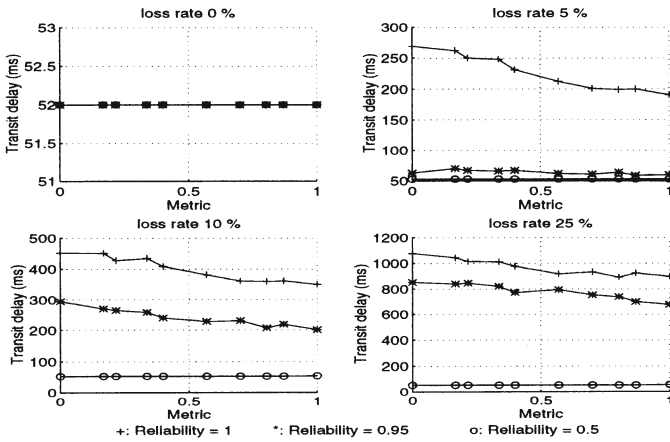


Figure 9 Transit delay evolution

(c) Reliability impact on sending/receiving buffers

In this subsection, reliability impact on used sending/receiving buffers is evaluated. The different loss rates which have been kept are the same as in the previous two subsections: 0, 5, 10 and 25%. Only three representative orders have been selected:

- 0, which represents total order (serial compositions only).
- 0.569, which represents an intermediate partial order between total order and null order (serial and parallel compositions).
- 1, which represents total disorder (parallel compositions only).

Sending (respectively receiving) buffers evolution is given on figure 10 (respectively on figure 11).

**Sending buffers (figure 10):** Partial reliability management allows the receiving POC entity to acknowledge lost PDUs (no retransmission will be performed) if these losses still match the selected reliability service; remember that a PDU is considered to have been lost by the network when a PDU that follows it in the partial order has been received. It then appears that when relaxing

the reliability constraints, the sending used buffer are freed sooner, which is expressed on curves of figure 10 by the decrease of the mean used sending buffer numbers. Two other points have also to be noticed:

- Reliability impact on sending buffers is significant only when the reliability parameter is greater than the network loss rate; else, the protocol has not to process retransmissions since all losses generated by the network do not damage the required reliability service. A clear illustration of this last point is given for a 5% loss rate. Reliability impact is null if the corresponding service is less than 0.95; beyond this value, buffers saving significantly appears (about 20% for 0.95 reliability service compared with a totally reliable service).
- One can also notice that less buffers are used for a totally ordered service compared with a partial order one. This observation results from the protocol loss declaration mechanism. Indeed, this mechanism is activated by the receiving POC entity when an undeliverable PDU (regard with the selected partial order) is received and all not yet received PDUs that precede it in the partial order may be declared lost still ensuring the required reliability service. In this case, these PDUs are declared lost and then acknowledged by the receiving POC entity. Total order having more order constraints than other tested orders, previous acknowledgments are generated earlier, making sending buffers to be freed at the earliest time.

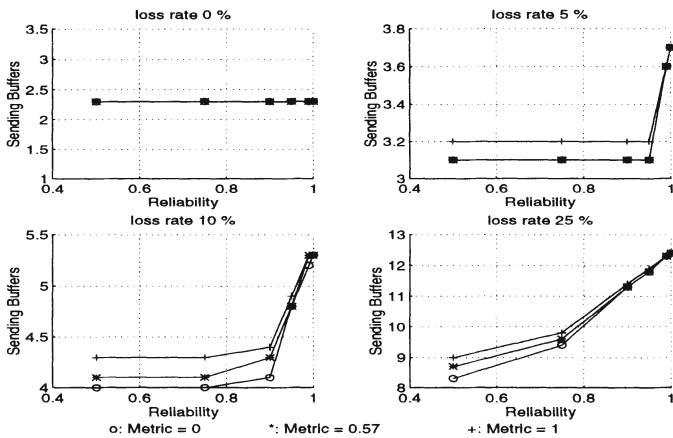


Figure 10 Sending buffers

**Receiving buffers (figure 11):** Analogous conclusions can be stated about reliability impact on receiving buffers. Particularly, it clearly appears that this impact becomes significant when losses generated by the network are greater than the user acceptable loss rate. For instance, if we look at the curves obtained for a 10% loss rate, about 50% of the used buffers are saved when a 0.95

reliability is provided compared with a total reliability (order=0.569); for a reliability lesser than 0.9, this gain no more appears.

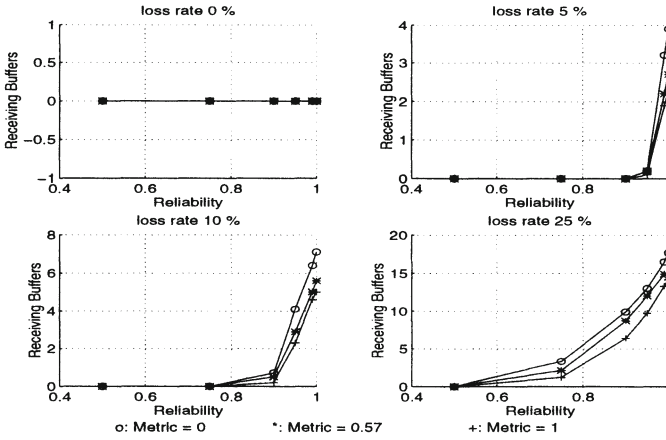


Figure 11 Receiving buffers

(d) Reliability impact on the transit delay

Reliability impact on the transit delay is shown on figure 12.

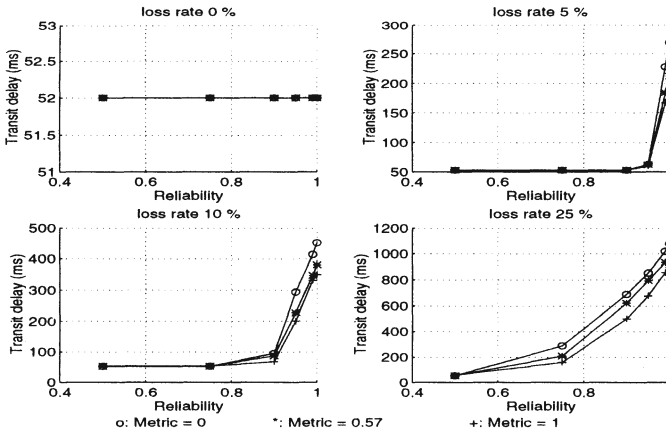


Figure 12 Transit delay

- For all curves, the lowest transit delay values are obtained when the network loss rate is lesser than the user acceptable loss rate; in this case, the protocol has not to process loss recovery and then does not increase the delay.
- Out of this particular context, benefits generated by a partial reliability management clearly appear. For instance, transit delay measure obtained with a 10% network loss rate is 40% lesser for a 0.95 reliability than for a total one (order=0.569). Such a gain (180 ms in the previous example)

allows service users to manage temporal synchronization constraints more easily, still ensuring reliability guarantees.

## 5 CONCLUSION

This paper has presented a few results performed to characterize a new concept of connection, the partial order connection, in a simulation environment (OPNET). The goal of this paper was to provide an evaluation of the benefits, related to transit delay and memory, a transport protocol implementing the POC concept generates. First, the basic principles of a partial order connection have been recalled; to classify and then compare the different partial order protocols tested in the paper, a metric,  $m$ , has also been detailed. Second, main mechanisms our POC protocol is constituted of have been presented to address the two major QoS parameters, 'order' and 'reliability'. The OPNET simulation environment has then been introduced and the communication model has been described. Simulation results have provided a first estimate of the benefits a POC may generate within an imperfect network environment, compared with a classical TCP-like connection. The following general conclusions may be stated.

Concerning order, it first appears that the less the order constraints are, the less buffers are needed and the lower applicative the transit delay is; however, note that the order parameter is entirely application dependent and cannot be modified, which is not exactly the case for the reliability (as users may modify it). When the network loss rate is lesser than the user required reliability, reliability management at the transport level is not useful, this QoS being ensured by the network itself. In the general case, that is when the network does not provide a sufficient enough reliability with respect to the application needs, very significant benefits have been shown when a partial reliability management is implemented at the transport level. Particularly, transit delay improvements when total reliability is not needed is extremely interesting for time sensitive or real time distributed applications, as the synchronization constraints of such applications are very difficult to enforce at the application layer.

## REFERENCES

- [1] P.D. Amer, C. Chassot, C. Connolly, P. Conrad, and M. Diaz. (1994) Partial order transport service for multimedia and other applications. *IEEE/ACM Transactions on Networking*, vol.2, n°5.
- [2] B. Berthomieu and M. Diaz. (1991) Modeling and verification of time dependent systems using time petri nets. *IEEE Transactions on Software Engineering*.
- [3] A. Campbell, G. Coulson, and D. Hutchison. (1994) A quality of service architecture. *ACM Computer Communications Review*.
- [4] D.R. Charinton and C.L. Williamson. (1989) Vmtp as the transport layer for high-performance distributed systems. *IEEE Communications Magazine*, pages 37-44.

- [5] I. Chriment. (1994) Impact of alf on communication subsystems design and performance. In *Proceedings of the 1st Int. Workshop on High Performance Protocol Architectures*, INRIA, Sophia Antipolis.
- [6] D.D. Clark, M.L. Lambert, and L. Zhang. (1987) Netblt : a bulk data transfer protocol. rfc 998.
- [7] D.D. Clark and D.L. Tennenhouse. (1990) Architectural considerations for a new generation of protocols. In *Proceedings of ACM SIGCOMM*.
- [8] A. Danthine, editor. (1994) *The OSI95 Transport Service with Multimedia Support*. Springer Verlag. Research Reports ESPRIT, Project 5341, OSI95, Vol.1.
- [9] L. Delgrossi and J. Sandvoss. (1993) The berkcom-ii multimedia transport system.
- [10] M. Diaz, A. Lozes, C. Chassot, and P.D. Amer. (1994) Partial order connections. a new concept for high speed and multimedia services and protocols. *Annals of Telecommunications, Tome 49, n° 5-6*.
- [11] M. Diaz and G. Pays. (1994) The cesame project : Formal design of high speed multimedia cooperative systems. *Annals of Telecommunications, Tome 49, n° 5-6*.
- [12] M. Diaz and P. Sénac.(1994) Time stream petri nets for multimedia information. In *LNCSS R. Valette (Ed), Int. Conf. on Application and Theory of Petri Nets, Zaragoza*.
- [13] C. Diot. (1995) Adaptive applications and qos guaranties. In *Invited paper to IEEE Multimedia Networking, Aizu, Japan*.
- [14] C. Diot, C. Huitema, and T. Turletti. (1995) Network conscious applications. In *HPCS workshop, Mystic*.
- [15] TENET group. (1994) *Recent and Current Research*. University of California, Berkeley, USA.
- [16] L. Henckel. (1993) Multimedia communication platform : Specification of the enhanced broadcast transport service. In *CIO RACE Project 2060*.
- [17] Protocol Engines Incorporated. (1992).
- [18] ISO/TC97/SC21/WG1-FDT/SG-B. (1987) Estelle, a formal description technique based on an extended state transition model. *Draft International Standard ISO/TC97/SC21-DP9074*.
- [19] T. Little and A. Ghafoor. (1990) Synchronization and storage models for multimedia objects. *IEEE Journal on Selected Areas in Comm., 8(3)*.
- [20] Inc. MIL 3. (1994) *OPNET documentation*. MIL 3 Incorporated / 34000 International Drive / Washington, DC 20008. OPNET documentation.
- [21] P. Sénac, M. Diaz, and P. de Saqui-Sannes. (1994) Toward a formal specification of multimedia synchronization scenarios. *Annals of Telecommunications, Tome 49, n° 5-6*.
- [22] B. Walter. (1983) Timed petri-nets for modeling and analysing protocols with time. In *Proceedings of PSTV, III, IFIP*.
- [23] N. Williams and G.S. Blair. (1994) Distributed multimedia applications : a review. *Computer Communications, vol.17, n° 2*.