PERFORMANCE ANALYSIS OF THE CLASS 4 TRANSPORT PROTOCOL

Andrzej PACH*, Patrick BROWN, and Brigitte DELOSME
Centre National d'Etudes des Télécommunications,
Issy les Moulineaux, France.

This paper deals with the evaluation of the mean reassembly time and transit delay of a message at the transport layer. Bounds on these values are given when the network connection is fixed and the packet loss probability is low. Detailed simulations of TP4 with various parameters show the influence of timeout on the performance indices.

1. INTRODUCTION

Transport protocols such as the CCITT-ISO class 4 transport protocol [1] have been designed to improve the reliability of data transfer between two end points over a packet-switched network or internetwork. Reliability is specially required by the transport service user for connection-oriented applications, like e.g. file transfer. Besides end-to-end flow control, the class 4 transport protocol (for short TP4) ensures packet corruption and loss recovery. Performance characteristics of the transport service will depend on the grade of service offered by the network layer.

The key performance measures we focus on are the reassembly time and the transit delay of a message, or transport service data unit (TSDU). The reassembly time is defined as the time elapsed between the correct reception of the first and the last missing packet (DT TPDUs) completing the TSDU. The transit delay is the time elapsed between the TSDU sending request and the availability of the TSDU at the destination transport service user.

In this paper we investigate the behaviour of TP4 for various network conditions in function of protocol parameters (window size, timeout), when transferring multipacket messages over a fixed network path. First, we give bounds which allow us to estimate the mean reassembly time and the mean transit delay of a message when TP4 operates in low packet loss probability conditions. Next, through a detailed simulation of TP4, we aim to give a way to choose window size and timeout values in order to obtain the best performances over the present network.

2. THE MODEL

The TP4 data transfer phase is modelled as depicted in Fig. 1. For simplicity it illustrates only the simplex transfer of user data between two end transport entities.

The TP4 user in A sends data in TSDUs which are divided in packets, causing a bulk arrival of DT TPDUs at the transport station. DT TPDUs are stored until they may be transferred to the network, i.e. when the transmit window is open.

* A. Pach was on leave at CNET during this work, he is with the Institute of Telecommunications, University of Mining and Metallurgy, Cracow, Poland.
Thus, the transmit buffer can be modelled by means of two queues in tandem with blocking at the second queue. The DT TPDPUs are removed from this queue when acknowledged by AK TPDUs issued by the receiving transport entity B. A single AK TPU may acknowledge more than one DT TPDU as well as when a timeout expires more than one DT TPDU may be retransmitted. Elapsing the timeout for a given DT TPDU may be caused by its loss, loss of the associated AK TPDU, or a round-trip delay greater than the timeout. At the network access node we may have blocking, and in the other nodes we may lose packets (e.g. due to buffer congestion). The receiving transport entity arranges the accepted DT TPDPUs in a reassembly buffer according to their sequence numbers, and rejects duplicated ones. It acknowledges all DT TPDPUs received in sequence (including duplicates). DT TPDPUs received out of order are stored but acknowledged only when the sequence of missing DT TPDPUs in front of them is completed. When all the DT TPDPUs a TSDU consists of are gathered, the TSDU is made available to the user.

![Queueing model of a class 4 transport protocol](image1)

**Fig. 1 - Queueing model of a class 4 transport protocol**

This study considers a store and forward network, where packets are transmitted sequentially over a physical path which does not vary during the transport connection. Then we may model classically the data flow within the network by means of a tandem queueing system with background traffic streams at each node, see Fig. 2. For our purpose we assume that one DT TPDU corresponds with one packet in the network layer.

![Queueing model of a network connection](image2)

**Fig. 2 - Queueing model of a network connection**

5.3B.5.2
As defined in simulations, system parameters are: the number of nodes in the path (in both directions), node buffer sizes, link capacities, packet service times, background traffics, and propagation delays along the path. Input variables are TP4 window size and timeout value, as well as message length in packets and message arrival rate. For given system parameters, input variables will define network service quality as seen by the end users: blocking probability between transport and network layer, probability of packet loss (due to node buffer saturation) along the path, and round-trip delay. Input variables will also define transport protocol efficiency we characterize mainly by the mean reassembly time and transit delay of a TSDU. These values can be estimated by analytical means under some restrictive assumptions.

3. BOUNDS COMPUTATION

We are interested in computing key performance measures for a transport service user, namely:

1. the mean reassembly time of a TSDU, or message, and
2. the message mean transit delay.

Due to the difficulty of the general problem, our analytical study is based only on the model in Fig. 2. This assumes that the transport protocol operates over a network without loss and access blocking, and that the window size has been taken large enough.

A set of recursive equations describes the behaviour of the system at each node, when a TSDU consisting of \( L \) packets traverses a \( N \) hops path in the network:

\[
X_{i+1} = \sum_{j=2}^{L} T_{i}^{(j)} + \sum_{k=1}^{K(X_{i})} T_{b}^{(k)} + I(X_{i}) , \quad i=1,2,\ldots,N, \tag{1}
\]

where \( X_{i} \) is the gap between receipts of the first and the last message packets at node \( i \), \( T_{i}^{(j)} \) and \( T_{b}^{(k)} \) are the service times of message packet \( j \) and background packet \( k \) (resp.), \( K(X_{i}) \) is the number of background packet arrivals and \( I(X_{i}) \) is the idle period of the server, during \( X_{i} \).

From (1), we derive a lower and an upper bound on \( X = X_{N+1} \) under the following general assumptions:

i) background and message packets arrival processes are independent;

ii) background traffic is stationary and ergodic.

These bounds, denoted by \( X_{L} \) and \( X_{U} \) (resp.), are computed from node to node by the relations (see Ref. [2] for proof and comments):

\[
X_{L+1} = X_{L} = X_{U} = X_{U} = (L-1)T_{1} , \quad \text{and for } i=2,3,\ldots,N
\]

\[
X_{L+1} = \begin{cases} 
X_{L} + \rho_{i} X_{L} & \text{if } (1-\rho_{i}) X_{L} < (L-1) T_{i}, \\
X_{L} & \text{if not},
\end{cases} \tag{2}
\]

\[
X_{U+1} = X_{U} + (L-1) \left[ T_{i} - (1-\rho_{i}) \min\{ T_{m_{i}}, T_{m_{i-1}} \} \right], \tag{3}
\]

where \( \rho_{i} = \lambda_{i} T_{b_{i}} \) is the workload generated by the background traffic at node \( i \) and \( T_{m_{i}} \) is the transmission time or minimum service time of a message packet; several transmissions and retransmission delays may be experienced by a packet due to bit errors on the links.

5.3B.5.3
Bounds are presented in Fig. 3 along with results of the tandem queueing model simulation, for a homogeneous network (i.e., $\rho_i = \rho, T_i = T, T_b = T$), constant service times for message packets and various service time distributions for the background traffic. In this particular case, we have:

$$X_1 = (L-1) \frac{1-\rho}{1-\rho} T < X < (L-1) \frac{1+(N-1)\rho}{1-\rho} T.$$  \hspace{1cm} (4)

It can be noticed that the value of $X_1$ expressed in (4) is given in the literature as an approximation of the message mean reassembly time instead of its lower bound. Though the mean reassembly time tends to $X_1$ when $L \to \infty$ or when $\rho \to 1$, it can be significantly larger for short messages and usual load values.

![Fig. 3 - Message mean reassembly time bounds and simulation results.](image)

The message mean transit delay $\overline{D}$ is the sum of the network mean transit delay of the first packet of the message, $\overline{D}^{(1)}$, and of $X$. $\overline{D}^{(1)}$ is given by:

$$\overline{D}^{(1)} = \sum_{i=1}^{N} \overline{W}_i^{(1)} + T_i^{(1)} + t_i,$$ \hspace{1cm} (5)

where $t_i$ is a constant for node $i$ which includes all deterministic components such as node processing time and link propagation delay, and $\overline{W}_i^{(1)}$ is the mean waiting time of the message first packet. In order to obtain $\overline{W}_i^{(1)}$, we use a $M/G/1$ queueing model; arrivals of messages and background packets to the first node are assumed Poissonian, with intensities $\lambda$ (in bulks of $L$ packets) and $\lambda_1$ respectively, and we assume that the overall traffic arriving at node $i$ $(i > 2)$ is Poissonian with intensity $\lambda_i + \lambda L$. Using the P-K mean value formula we get:

$$\overline{D} = \frac{L(L-1)\lambda T_b^2}{2(1 - \rho_1)} + \sum_{i=1}^{N} \frac{\lambda_i T_b^2 + \lambda L T_i^2}{2(1 - \rho_i)} + \sum_{i=1}^{N} \left[ T_i + t_i \right] + X,$$ \hspace{1cm} (6)

where $\rho_i = \lambda_i T_b + \lambda L T_i$.

Bounds on $\overline{D}$ are easily derived from the bounds on $X$ given by (2) and (3).
4. SIMULATION RESULTS AND DISCUSSION

A detailed simulation of the TP4 model presented in Sect. 2 shows first that, if the packet loss probability is low, the mean values of interest are included between the computed bounds when the window size and timeout are correctly chosen. This is illustrated in Figs. 4, for a homogeneous network (with \( p=0.3 \), \( \lambda=0.02 \) and \( T_m=1 \)). We observe the existence of minimal window size and optimal timeout.

![Graphs showing simulation results.](image)

Figs. 4 - Simulation results in function of window and timeout parameters vs computed bounds (\( K=15 \)).

In various loss conditions (obtained by taking different node buffer sizes \( K \)), extensive simulations have shown that one can find window size and timeout values which maximize transport protocol efficiency, if the message flow does not saturate the network links. We will not discuss here about the choice of the window size - the best choice for the user is to take the largest window accepted by the receiving end -, but we will focus on the choice of the timeout.

We may first notice that the network grade of service does not affect strongly the optimal values of the performance indices seen by the TP4 service user. This is shown in Figs 5.a-5.d, where the measured delays have a maximum increase of 5% (Fig. 5.c) to 50% (Fig. 5.b) from low loss to high loss network conditions. We note also that the optimal value of each index corresponds with the same timeout value.

Another important result when optimising TP4 is that, whatever the performance measure we consider (see Fig. 5), there is an optimal timeout value which minimizes all the indices. In fact, the best timeout value from the user point of view is slightly smaller than the optimal timeout as regards to the transport and network efficiency.

It can be seen also that the optimal timeout is smaller when the loss probability is higher; as more packets are going to be lost, we have better to send duplicates in advance. But, though the optimal timeout may be smaller than the round-trip delay, the important outcome is that using a timeout equal to the transport optimum results in the best use of the network. This is shown, e.g., in Figs. 5.e,g,h, where network loss probability and traffic are stable and minimal for timeouts larger than or equal to the optimal value.
4 nodes, \( L = 5 \), \( \lambda = 0.02 \), \( \lambda_i = 0.5 \), window= 10 (1000 when \( K=500 \))
\( T_i = T_{D_i} = 1 \) (const.), \( T_{\text{send ack}} = 1 \) (expon.), \( t_i = 0.25 \)

5.a TSDU-Mean-Transit-Delay

5.b TSDU-Mean-Reassembly-Time

5.c DT-TPDU-Mean-Transit-Delay

5.d DT-TPDU-Mean-Life-Time

5.e DT-TPDU-Loss-Probability

5.f Mean-Number-of-DT-TPDU-Transmissions
The retransmissions due to a timeout smaller than the optimum are not going to improve the transport protocol efficiency; the performance indices for the service user may be close to the optimum, but the transport protocol has more work to do at both ends, sending more packets and receiving more duplicates or acknowledgements (see Figs. 5.f-5.h). Furthermore, the resulting overload may significantly reduce the network performances, in spite of the control due to network access blocking.

Then, if the timeout is to be determined by an adaptive scheme, we suggest to start transmissions with a large timeout and to shorten its value to the optimum, e.g. as long as an index such as the mean number of DT TPDU transmissions does not increase (see Fig. 5.f). Another useful indication of the timeout optimality is the intensity of AK TPDU reception (see Fig. 5.h). The optimum is reached when this intensity equals the offered intensity of message packets (i.e. $\lambda L$). This index gives a direct means to decide whether the timeout must be reduced or increased.

5. CONCLUSION

The presented bounds on the mean reassembly time and transit delay of a TSDU, when packet loss probability is low, and a detailed simulation of TP4 under different packet loss conditions have given significant indications on the transport protocol performances. These results help to determine suitable values of the window size and timeout parameters. Further studies under development concern mainly the derivation of analytical models taking into account the packet loss probability and flow control in the network.

REFERENCES