

Modeling the Steady State Throughput of SCPS-TP

Feng Liu¹ and Shilong Liu²

¹School of Electronics and Information Engineering, Beihang University, Beijing, China

²Beijing Key Laboratory for Network-based Cooperative Air Traffic Management (No. BZ0272),
Beijing Laboratory for General Aviation Technology, Beijing, P.R. China

Abstract—Space Communication Protocol Specification(SCPS)-Transport Protocol(SCPS-TP) is proposed for space environment, in which TCP/IP protocols do not work very well. In this paper we develop a simple model to analyze the steady state throughput of TP protocol. The throughput model is achieved with an approximate method based on Padhye model, which has two parameters: the round trip time(RTT) and the packet loss rate. We assessed our model by conducting a series of simulations using the NS2.35 simulator. The result shows our model matches the simulation very well. Then we compared the performance of TCP and TP in various environment, and the result confirmed that TP conducts better than TCP in Space environment.

Keywords-space communication; SCPS; TP; throughput model; simulation

I. INTRODUCTION

Nowadays space mission is getting more and more popular. Communication in space environment is one of the most important work. But traditional TCP/IP protocol stack does not perform very well in space environment which has long delay and high loss rate. Consultative Committee for Space Data Systems(CCSDS) proposed a set of specifications to meet the space communication requirement. SCPS-TP is the transport layer protocol, which guarantee end-to-end reliable communication. TP protocol is a variety of TCP. It adopts Vegas mechanism to avoid congestion and Selective Negative ACK to retrieve from error.

II. THROUGHPUT MODEL

Actually TP's performance is affected by two main aspect: the vegas congestion mechanism and Selective Negative Acknowledgment(SNACK) function. Referring to Padhye Model for TCP Reno, we calculate the steady state throughput with two parameters: round trip time(RTT) and packet loss rate(p). Mary K vernon used the same method to calculate the throughput of tcp vegas. But vernon only consider the situation that there is less than or equal to one packet lost in one congestion window. In space communication environment, one congestion window may occur packet loss multiple times. When multi-loss happens, the TP receiver will invoke the SNACK function, which will report all the lost packets to TP sender after a time constant(denoted as Ts).

As with previous models[1-5], we calculate TP throughput in rounds, where a window of packets is transmitted per round. The duration of every round is independent to window size and is assumed to be equal to RTT. Then we model the TP flow in the following three aspect.

A. No Loss Model

In this section, TP performs the same as TCP Vegas. The throughput is derived by vernon as follows:

$$\Lambda_{no-loss} = \frac{\beta}{R-baseRTT} \quad (1)$$

where R denotes the expectation of RTT, baseRTT denotes the base RTT in the flow, and β is a preset constant number.

B. No Time-out Model

When TP flow go through a space link, packet loss is inevitable. Such loss may be bit error or queue drop.

In this section we assume that all losses will be identified, so that time-out(TO) will not be triggered. Given this assumption, when a loss occurs, TP will decrease the congestion window by 1/4, and get into fast re-transmit/ fast recovery(fr/fr) period. Only new packets is received, TP get into congestion avoidance period. When loss occurs, TP get back into fr/fr period again, and so on. We call the two period(fr/fr and congestion avoidance) transport period(TPP).

As is illustrated in Figure I, the TP flow is consisted of one slow start period and a series of TPP periods which are independent with each other.

While the slow start period can be ignored in bulk data transfer. TP flow's throughput is approximately equal to TPP's throughput.

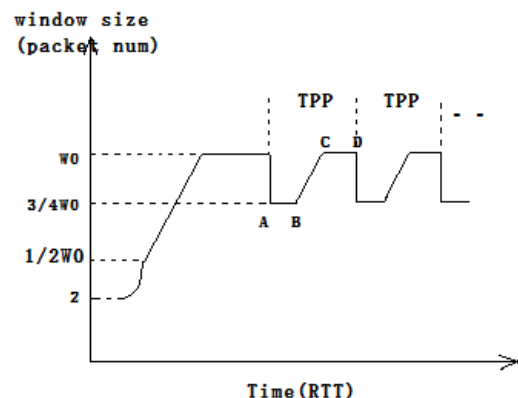


FIGURE I. THE EVOLUTION OF WINDOW SIZE WHEN THERE IS NO LOSS

To derive TPP's throughput, we need to analyze the three stage of TPP:(a) the stage from point A to point B, (b) the stage from point B to point C, (c) the stage from point C to point D, as is illustrated in Figure I.

After slow start period, when a loss occurs, TP get into fr/fr period, which is stage AB. Then TP receiver will send SNACK option after T_s . In this stage, the TP sender's window will stop sliding until receiving an ACK packet, which is triggered by new sent packet. We assume that SNACK will acknowledge all losses to the sender. Then we can get the number of packets(P_{AB}) transmitted during stage AB is equal to the window size(W_0):

$$P_{AB} = W_0 \quad (2)$$

Then we need to solve the duration of stage AB, which has two conditions:

- There is no loss except for the loss at A point. The duration of AB(denotes as D_{AB}) is equal to R (the expectation of RTT).
- There is more than one loss in AB. The duration of AB is $R+T_s$.

The probability of former is $(1-p)^{W_0-1}$. The probability of latter is $1-(1-p)^{W_0-1}$. Thus,

$$\begin{aligned} D_{AB} &= (1-p)^{W_0-1} \times R + (1-(1-p)^{W_0-1}) \times (R + T_s) \\ &= R + (1-(1-p)^{W_0-1}) \times T_s \end{aligned} \quad (3)$$

After stage AB, TP get into stage BC, where the window size increase linearly from $3/4W_0$ to W_0 , with one packet per RTT. So the expected packet number transmitted in stage BC, P_{BC} , is,

$$P_{BC} = \sum_{i=3/4W_0}^{W_0} i = \frac{7W_0^2}{32} \quad (4)$$

The expected duration of BC, D_{BC} , is,

$$D_{BC} = \frac{W_0}{4} \times R \quad (5)$$

Then TP get into stage CD, where the window size holds still. We notice that from point B to point D, there is no loss. Thus the number of packets transmitted in BD, P_{BD} , is,

$$D_{BC} = \frac{W_0}{4} \times R \quad (6)$$

The expected duration of CD is the number of packets transmitted in this period over W_0 . Thus,

$$D_{CD} = \frac{P_{BD}-P_{BC}}{W_0} \times R \quad (7)$$

Finally, the expected number of packet re-transmitted in TPP is:

$$P_{TPP} = P_{AB} + P_{BC} + P_{CD} = W_0 + \frac{1}{p} \quad (8)$$

The expected duration of TPP is:

$$D_{TPP} = D_{AB} + D_{BC} + D_{CD} \quad (9)$$

The expected throughput of TPP is:

$$\begin{aligned} \Lambda_{no-timeout} &= \frac{P_{TPP}}{D_{TPP}} \\ &= \frac{W_0 + \frac{1}{p}}{R + (1-(1-p)^{W_0-1}) \times T_s + \frac{W_0}{4} \times R + \left(\frac{1}{p \times W_0} - \frac{7W_0}{32} + \frac{1}{8} \right) \times R} \end{aligned} \quad (10)$$

C. Full Model

The other aspect of TP flow is time-out(TO).In this section, loss happened, but they are not identified by the sender. A TO occurs, when there is not enough duplicated ACKs return to the sender to trigger re-transmission after loss happened. Traditionally three duplicated ACKs will trigger re-transmission, but because of Vegas congestion prediction and SNACK function, TP can trigger re-transmission by a single duplicated ACK.

When a packet time-out happens, TP will remain idle for a period of T_0 , and then reset window size to two and get into slow start period. T_0 is equal to two times the smooth RTT plus four times the RTT variance.

We can see that the evolution of the flow is periodic, as is illustrated in Figure II. The behavior of the flow can be divided into a series of intervals. Each interval begins with slow start period and ends up with TO period. All the intervals are statistically independent with each other.

To derive the expected throughput of one interval, we calculate the expected number of packets transmitted and the duration of such a interval. The interval can be divided into three stages: (1)the no loss stage from point A to D, (2)a series of n TPP stage which is analyzed in section II-B, (3)the time-out stage. Thus, the throughput of TO is:

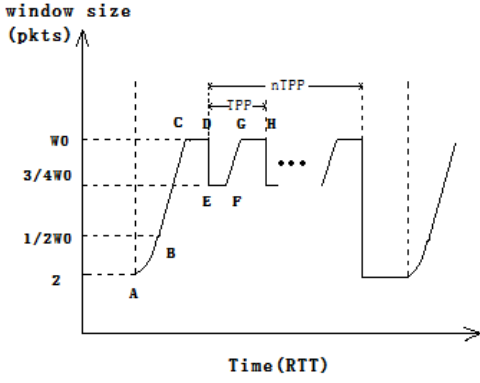


FIGURE II. THE EVOLUTION OF WINDOW SIZE WHEN TO OCCURS

$$\Lambda_{TO} = \frac{P_{AD} + nP_{TPP} + P_{TO}}{D_{AD} + nD_{TPP} + D_{TO}} \quad (11)$$

Where P_X denotes the expected number of packets in period X , and D_X denotes the expected duration of period X . Then we analyze each period respectively.

1) *Start period between point A and D.* In this period, TP restarts transmission at A point and ends up with a packet loss at D point. So the expectation of transmitted packets' number is:

$$P_{AD} = \frac{1}{p} \quad (12)$$

Then we calculate the expected duration of this period, D_{AD} . In the stage from point A to point B, the window size is doubled every other RTT from two to the ssthresh($1/2W_0$).

$$P_{AB} = 2 \left(2 + 4 + 8 + \dots + \frac{W_0}{4} \right) = 2^{\log W_0} - 4 \quad (13)$$

$$D_{AB} = 2 \times \log \left(\frac{W_0}{4} \right) \times R \quad (14)$$

In the stage from point B to point C, the window size increase one packet size every RTT from $1/2W_0$ to W_0 .

$$P_{BC} = \sum_{i=1/2W_0}^{W_0} i = \frac{3W_0^2}{8} \quad (15)$$

$$D_{BC} = \frac{W_0}{2} \times R \quad (16)$$

In the last stage from point C to point D, the window size remain unchanged(W_0). We have:

$$P_{CD} = P_{AD} - P_{AB} - P_{BC} = \frac{1}{p} - 2^{\log W_0} + 4 - \frac{3W_0^2}{8} \quad (17)$$

$$D_{CD} = \frac{P_{CD}}{W_0} = \frac{1}{pW_0} + \frac{4 - 2^{\log W_0}}{W_0} - \frac{3W_0}{8} \quad (18)$$

2) *A series of n TPP period.* In section II-B we analyzed the expected throughput of a TPP period. Then we need to derive an expression of n , the expected number of TPP in one interval. We note from Figure II that the probability of getting into TO, p_{TO} , is $1/(1+n)$. Thus,

$$n = \frac{1 - p_{TO}}{p_{TO}} \quad (19)$$

There are two conditions to trigger time-out: (1) all the packets in a window are lost, (2) i out of w packets are lost, the sender is acknowledged the loss condition by SNACK function and re-transmit the i packets, which are all lost later.

$$p_{TO} = p^{W_0} + \sum_{i=1}^{W_0} \left(C_{W_0}^i \times p^i \times (1-p)^{W_0-i} \times p^i \right) \quad (20)$$

$$\approx W_0 \times p^2 \times (1-p)^{W_0-1} + \delta(p^4)$$

- (3) Time-Out period

In this period, no packet is transmitted, thus,

$$P_{TO} = 0 \quad (21)$$

After a TO period, TP send two new packets, then slow start may begins and another TO may follows too. Consider the following four cases: (i) both packets are lost, with the probability $p_1=p^2$, (ii) only the first packet is lost, with the probability $p_2=p(1-p)$, (iii) only the second packet is lost, with the probability $p_3=p(1-p)$, (iv) nether of the packets is lost, with the probability $p_4=(1-p)^2$.

Obviously, case(i) will lead to one more TO, and case(iv) will lead to slow start. Because of SNACK function, we assume that TP can detect packet loss in case(ii) and case(iii). Let M be the number of triggered TO times. Since the first TO is given,

$$p[M = k] = p^{2k} \quad (22)$$

The duration of first TO is T_0 . For each new TO the duration will be doubled until the 64 times the T_0 . Ignoring the higher order of p^2 , we get:

$$D_{TO} = T_0 + p^2 \times 2T_0 \quad (23)$$

Finally, The expected throughput of full model is:

$$\Lambda_{TO} = \frac{\frac{1}{p} + n \left(W_0 + \frac{1}{p} \right)}{D_{AB} + D_{BC} + D_{CD} + nD_{TPP} + T_0 + p^2 \times 2T_0} \quad (24)$$

Where the expression of D_{AB} , D_{BC} , D_{CD} , D_{TPP} and n can be computed from equation(15), (17), (19), (9), (20) and (21).

III. EXPERIMENT RESULTS

To validate the throughput full model presented in section II, according to the blue book for TP[6], we developed TP protocol on ns-2 simulator[7], and conducted a series of simulations. We also compared the performance of TCP and TP.

Our simulation is based on the simple dumb-bell network. Some previous simulation assumed a small buffer at the bottleneck link[1,3]; some assumed a larger buffer[2]. To obtain a more common result and to simplify the variables, we adopted a small buffer(10pks), and few background traffic.

Firstly, we conduct a series of experiments to compare the theoretical throughput and the simulation throughput. The delay of the link is 204ms, the bit error rate changed from 0.01% to 1%, the local network bandwidth is 100Mbps, the bottleneck link bandwidth is 10Mbps, and the bottleneck link buffer is set to 10(pkts).

As is illustrated in Figure III, our model fit the simulation very well. Only when the loss rate is very high, the theoretical throughput is slightly higher than simulation.

Then we compared the throughput of TCP and TP in various conditions.

Figure IV shows the throughput of TCP and TP with the packet loss rate changing from 0.1% to 1%, where the baseRTT is 204ms. Because of the buffer's limit, TCP packet loss rate can't be lower. we can see in high loss environment

Figure V shows the throughput with various of RTT, where the packet loss rate keeps 1%. We can see TP performs better than TCP in high packet loss rate environment, and TP is less sensitive to RTT compared to TCP.

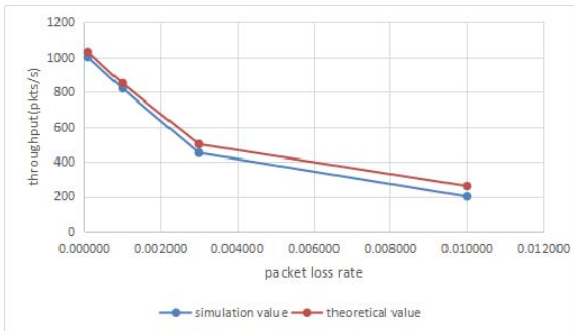


FIGURE III. THROUGHPUT BETWEEN SIMULATION AND THEORY

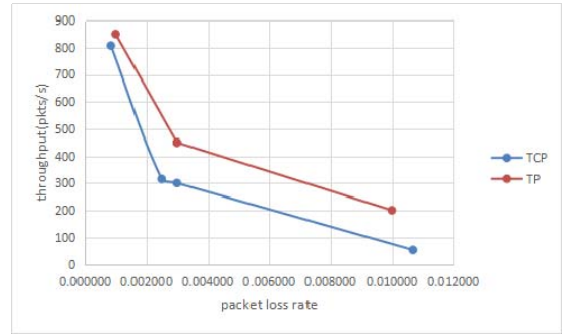


FIGURE IV. THROUGHPUT WITH VARIOUS PACKET LOSS RATE

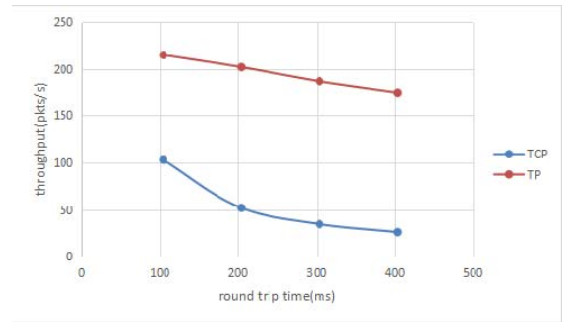


FIGURE V. THROUGHPUT WITH VARIOUS ROUND TRIP TIME

IV. CONCLUSION

This paper has prospered a throughput model for SCPS-TP protocol and has valued this model by a series of experiments on ns-2 simulator. Simulation also shows that in high loss rate or high RTT environment, TP performs better than TCP.

REFERENCES

- [1] U. Hengartner, J. Bolliger, and T. Gross. "TCP Vegas revisited." In INFOCOM 00, Tel Aviv, March 2000.
- [2] Samios, Charalampos, and M. K. Vernon. "Modeling the throughput of TCP Vegas." *Acm Sigmetrics Performance Evaluation Review* 31.1(2003):71-81.
- [3] Altman E, Avrachenkov K, Barakat C. "A stochastic model of TCP/IP with stationary random losses" [J].*IEEE ACM Trans on Networking*, 2005, 13(2) : 356-369
- [4] Bullot H, Ttrel L, Jones R H. "Evaluation of Advanced TCP Stacks on Fast Long Distance Production Networks"[C] *Proc. of the 2nd Workshop on PFLDnet*. Argonne, Canada: [s. n.], 2004.
- [5] Kumazoe K, Hori Y, Tsum M, et al. "Transport Protocols for Fast Long Distance Networks: Comparison of Their Performance in JGN"[C] *Proc. of International Symposium on Applications and the Internet Workshops*. Tokyo, Japan: [s. n.], 2004.
- [6] CCSDS 714.0-B-2 <http://public.ccsds.org/publications/archive/714x0b2.pdf>
- [7] Ns-2 simulator, <http://www.isi.edu/nsnam/ns>.