Primers of Using Analytical Tools to Capture TCP Behavior

[1]Biljana Stojcevska, [2]Oliver B. Popov

[1]Institute of Informatics, Faculty of Natural Sciences and Mathematics
Saints Cyril and Methodius University, PO Box 162, MK-1000 Skopje
Republic of Macedonia
[2]Department of Information Technology and Media, Mid Sweden University
SE - 851 70, Sundsvall, Sweden
{biljanas@ii.edu.mk oliver.popov@miun.se}

Abstract: Arguably, understanding the behaviour of TCP is essential to understanding the behaviour of the whole Internet hence (1) the majority of traffic flows use it for their transportation, (2) it has been around since the inception of the global net thus showing remarkable scalability and robustness, (3) it has been a subject of many modifications in order to absorb technological innovations, and (4) through its self-clocking disposition has displayed flexibility and fairness. The heuristic nature of the TCP extensions and the inherit complexity of the Internet have provided the justification behind using testbeds and simulations in order to describe and study the nature of TCP and the effect of the protocol on the Net. However, having in mind that rudimentary TCP protocol is actually FSM, as well as acknowledging the need for analytical modelling that will supplement the empirical work and thus induce the necessary predictability, has lately induced an intensive research for mathematical paradigms applicable and usable in TCP studies. The article enumerates some of these efforts and compares their effectiveness.

Keywords: analytical modeling, TCP analysis, IP networks

1 Introduction

TCP dominates the transport layer on the Internet 9. Hence, understanding the way Internet traffic behavior is influenced by the protocol, its major services and the mechanisms through which they are implemented is crucial to the overall system performance and stability.

The protocol breaks the incoming byte stream from the higher-level protocols into segments and sends them as separate IP packets. The flow of segments is governed by a sliding window mechanism at the sender and the receiver.

Three different regimes describe mainly the entire TCP operation and these are termed as Slow Start, Congestion Avoidance and Additive Increase Multiplicative Decrease (AIMD). The Slow-Start 4 algorithm is used at connection start and or/restart (and is equivalent to system reset). In this case, the congestion window

increases by a single segment every time a new acknowledgement is received. When the window reaches the size of the congestion threshold, there is a switch to Congestion Avoidance phase, when the window increases by 1/*cwnd* on the receipt of every new acknowledgement. The growth of the window is linear, which translates to one segment in each RTT. In fact, Congestion Avoidance forces a connection to path adaptation and provides conditions for fairness (by preventing flow starvation). Finally, the TCP sender implements AIMD 2 for dynamic adjustment of the congestion window, enables additional factors for convergence to fairness and makes some network stability attainable.

The emergence of new technologies posited the need for TCP evolution and eventual modifications such as Fast Retransmission, Recovery, SACK, Snoop, Peach to enumerate a few. It is an ongoing process and it should continue in the future having in mind both the prevalent role of TCP as a transport protocol on IP networks.

Optimal or near optimal TCP performance is usually a condition for similar performance by networks and the Internet in general. Consequently this makes the TCP research area one of the most active in the real of Internet analysis and modeling. The complexity of the protocol and the many key services it provides warrant a holistic approach to its study based on the combination of empirical, simulation, and analytical tools. In the case of the later, by using analytical or mathematical models of the protocol it is possible to define (1) sound and relevant performance metrics, and (2) the optimal operating conditions.

2 Window behavior

The basic elements of the protocol that must be included in a model are the sliding window process and the packet loss. The congestion window is characterized by its size, which is commonly initialized by the receiver (window size) and then governed by the slow start and congestion avoidance algorithms.

Every TCP connection starts with a “slow start”, and upon reaching the congestion threshold proceeds with congestion avoidance. Figure 1 illustrates the congestion window evolution.

Let w be the size of the congestion window. TCP sends w bytes to the destination every RTT seconds. Once the TCP connection enters the steady state, i.e. gets into the congestion avoidance phase, it continuously probes the network bandwidth and increases w by one segment on each RTT. This leads to congestion at one point triggering a packet loss, which halves the size of the congestion window. The process continuous with the congestion avoidance phase by increasing w until another packet loss occurs.

Figure 2 depicts a standard behavior of a single TCP connection. The *warm-up* period is spent in slow-start. After the congestion window threshold is established, the protocol goes into congestion avoidance and starts its *steady state*.

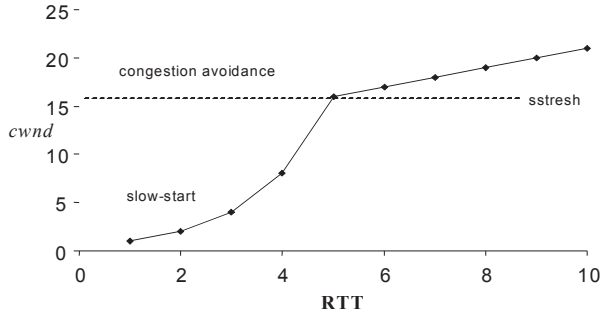

Figure 1. Slow-start and congestion avoidance

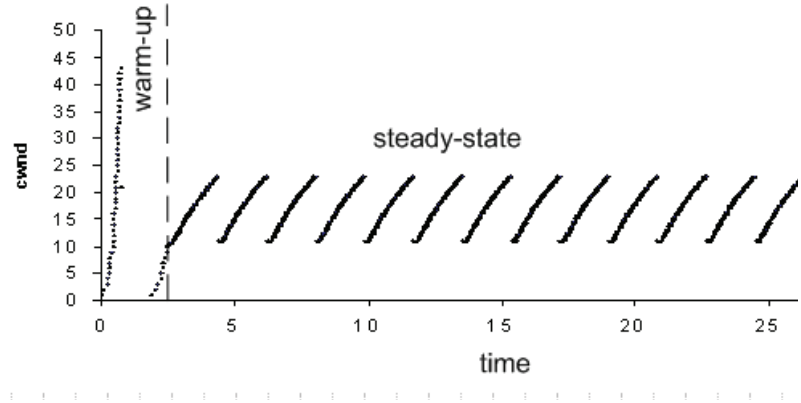

Figure 2: Warm-up to steady state transition

If we denote by *W(t)* the size of the window at time *t* when a packet loss happened, then we can conclude that TCP sends its segments at rates varying between the values

$$\frac{W(t)}{2 \times RTT} \text{ and } \frac{W(t)}{RTT}.$$

Hence, the simplified model of the TCP throughput would be:

$$average\ throughput = \frac{0.75 \times W(t)}{RTT} \quad (1)$$

Usually, TCP connections compete for bandwidth with other network traffic. The information about the traffic load is communicated solely through packet losses to which TCP responds by decreasing its window. Since the traffic behaviour is probabilistic in nature, the common way to model the uncertainty is through stochastic processes that use the probability *p* of loosing a packet.

3 TCP Models

Simplified periodic model

As indicated, the congestion avoidance algorithm describes completely the TCP steady state phase. In 6 the macroscopic behavior of a single TCP connection is modeled. Let us consider a simplified case, where both *RTT* and the probability *p* for random packet loss are constant. Under these assumptions, the link delivers approximately *1/p* packets before a packet loss occurs. If the maximum value of *cwnd* is *W*, then the minimal value would be *W/2*.

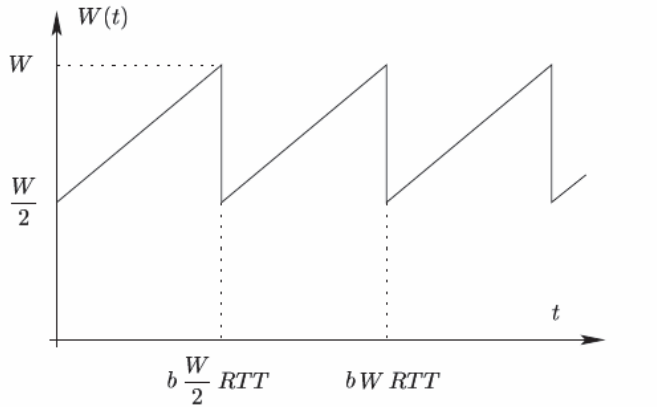

Figure 3. cwnd behaviour

Since the window opens for one segment on each *RTT*, each cycle of the steady state of the connection must last for *RTT*W/2* seconds. The total delivered data in each cycle is:

$$\left(\frac{W}{2}\right)^2 + \frac{1}{2}\left(\frac{W}{2}\right)^2 = \frac{1}{p} \tag{2}$$

Solving (2) by *W* yields

$$W = \sqrt{\frac{8}{3p}} \tag{3}$$

In this case, the bandwidth of the connection *BW* is:

$$BW = \frac{data\ per\ cycle}{time\ per\ cycle} = \frac{MSS \times \frac{3}{8} W^2}{RTT \times \frac{W}{2}} = \frac{MSS}{RTT} \frac{\sqrt{\frac{3}{2}}}{\sqrt{p}} = \frac{MSS}{RTT} \frac{C}{\sqrt{p}} \quad (4)$$

where $C = \sqrt{\frac{3}{2}}$ is named a *constant of proportionality*. It is one of the fundamental results, also known as $C/\sqrt{p}$ *law* or the *inverse square-root p law* that holds for many other TCP models.

For *BW* it is in fact the upper bound , where *C* is usually less than 1. The value of *BW* given by

$$BW < \frac{MSS}{RTT} \frac{1}{\sqrt{p}} \quad (5)$$

reflects realistic situations though the model does not account for timeouts that induce slow-start phase and thus decrease the throughput.

Stochastic modeling

The simplified model was refined to account for retransmission timeouts 8 such as the dependence of congestion avoidance on ACK behaviour (cases that cover duplicate ACK detection or timeout). The congestion-avoidance behaviour is considered in term of "rounds", which start with sending a window of *W* segments, and end with the first received ACK for that window. If *b* denotes the number of acknowledged segments then the value of b is either 1 or 2 (when delayed acknowledgements are employed). So, after sending *W* segments, the next value of the congestion window will be *W'=W+1/b*.

In this case, the assumption is that loss is indicated by the receipt of three DUPACKs. For a connection starting at *t=0*, let *Nt* be the number of packets transmitted in the interval *[0, t]*, and *Bt=Nt/t* the corresponding throughput. The long-term steady-state of the TCP connection throughput is:

$$B = \lim_{t \to \infty} B_t = \lim_{t \to \infty} \frac{N_t}{t} \quad (6)$$

The goal is to establish a relationship *B (p)* between the throughput of the TCP connection and the probability *p* that a packet is lost.

TDP denotes a period between two triple-duplicate loss indications (Figure 4):

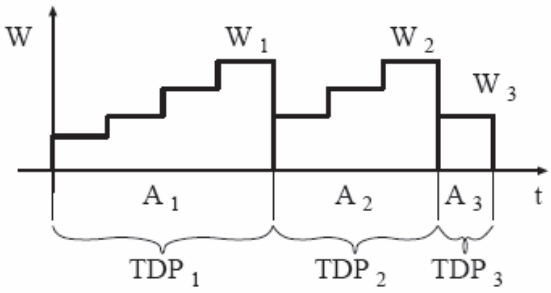

Figure 4: TDP periods

Relative to *i*-th *TD* period, let *Yi* is the number of packets sent, *Ai* is its duration, while *Wi* is the window size at the end of it. Then it is shown that *{Wi}* is a Markov regenerative process with rewards *{Yi}* and

$$B = E[Y]/E[A] \tag{7}$$

Let αi denote the number of first packet lost in *TDPi*, and *Xi* the round where this loss occurs.

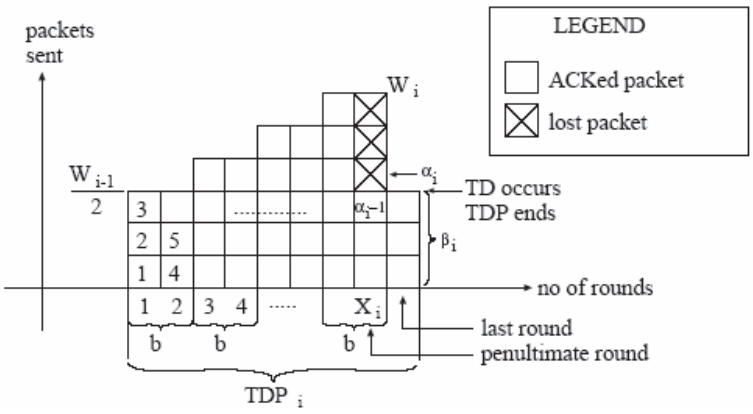

Figure 5: TDP with timeout

When in round *Xi+1*, $Yi = \alpha i + Wi - 1$ segments are sent we have:

$$E[Y] = E[\alpha] + E[W] - 1 \tag{8}$$

The assumption is that the random process $\{\alpha_i\}$ is a sequence of independent and identically distributed (i.i.d.) random variables. Hence, the probability that $\alpha_i = k$ and exactly *k-1* packets are successfully acknowledged before a loss occurs is:

$P[\alpha i = k] = (1-p)^{k-1}p$, where $k = 1, 2, \ldots$ (9)

and

$$E[\alpha] = \sum_{k=1}^{\infty}(1-p)^{k-1}p^{k} = \frac{1}{p}, \quad (10)$$

therefore:

$$E[Y] = \frac{1-p}{p} + E[W], \quad (11)$$

The connection between $E[W]$ and $E[A]$ is established through $E[A]=(E[X]+1)E[r]$, where r_{ij} is defined as the duration of the j-th round, so

$$A_i = \sum_{j=1}^{X_i+1} r_{ij}. \quad (12)$$

It is shown that

$$E[W] = \frac{2}{b}E[X] \quad (12)$$

holds by assuming that $\{Xi\}$ and $\{Wi\}$ are mutually independent sequences of i.i.d. random variables. The derivation of the expressions for $E[A]$ and $E[X]$ leads to the expression for $B(p)$:

$$B(p) = \frac{\frac{1-p}{p} + \frac{2+b}{3b} + \sqrt{\frac{8(1-p)}{3bp} + \left(\frac{2+b}{3b}\right)^2}}{RTT\left(\frac{2+b}{6} + \sqrt{\frac{2b(1-p)}{3p} + \left(\frac{2+b}{6}\right)^2 + 1}\right)} \quad (13)$$

This can be expressed as:

$$B(p) = \frac{1}{RTT}\sqrt{\frac{3}{2bp}} + o\left(\frac{1}{\sqrt{p}}\right) \quad (14)$$

So, for small values of p and $b = 1$ (14) reduces to (4).

The model is extended to include timeouts also. Let $\hat{Q}(w)$ denote the probability that a TDP loss indication in a window size w is due to a timeout. The complete model obtained in 8 is:

$$B(p)=\begin{cases}\dfrac{\dfrac{1-p}{p}+E[W]+\hat{Q}(E[W])\dfrac{1}{1-p}}{RTT(\dfrac{b}{2}E[W_u]+1)+\hat{Q}(E[W])T_0\dfrac{f(p)}{1-p}} & if\ E[W_u]<W_{\max}\\ \dfrac{\dfrac{1-p}{p}+W_{\max}+\hat{Q}(W_{\max})\dfrac{1}{1-p}}{RTT(\dfrac{b}{8}W_{\max}+\dfrac{1-p}{pW_{\max}}+2)+\hat{Q}(W_{\max})T_0\dfrac{f(p)}{1-p}} & otherwise\end{cases}$$

(15)

where *Wu* is the unconstrained window size, *Wmax* the maximum congestion window size, and T_0 is the time when timeout begins and $f(p)=1+p+2p^2+4p^3+8p^4+16p^5+32p^6$.

AQM interaction

The relation between the throughput of a TCP connection and the loss process has been established along with the assumption of independence between the later and the data flow. What follows is a discourse on the interplay between Active Queue Management (AQM) and the operation of TCP.

AQM is designed to prevent network congestion before the router buffers are overflown. The implementation on network layer aims implicitly to notify the TCP senders to slow down, viz. reduce their sending rates. Currently, there is number of AQM schemas either proposed or being studied in the IP community. While each one may induce some improvements in network performance, there is no consensus about a single one having in general a clear advantage over the others.

The default AQM scheme recommended by IETF is Random Early Detection (RED) 3, which improves the performance of the network by introducing a proactive form of congestion control. RED prevents global synchronization of the sources (may happen with the traditional drop-tail buffers). The algorithm discards packets arriving at the queue randomly with a given probability. The basic idea is to increase the dropping probability as the mean queue size increases. Figure 6 shows the dropping function considered throughout the section. It is a linear function between two thresholds, a lower one t_{min} and an upper one t_{max}. The form of the function *p(x)* is:

$$p(x)=\begin{cases}0 & 0\le x<t_{\min}\\ \dfrac{x-t_{\min}}{t_{\max}-t_{\min}}p_{\max} & t_{\min}\le x\le t_{\max}\\ 1 & t_{\max<x}\end{cases}\qquad(16)$$

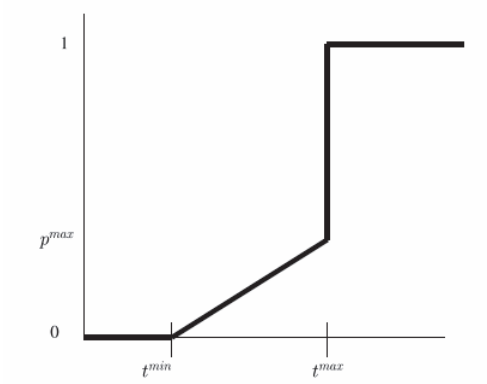

Figure 6: RED dropping function

By using AQM, the network can be modeled as a control system that is governed by the AQM feedback to the network (Figure 7).

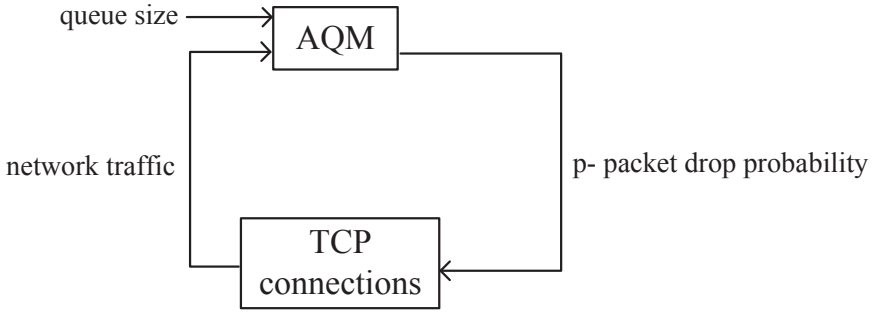

Figure 7: Control system network model

The round-trip delay of a single TCP source can be expressed as:

$$R(t) = d_p + \frac{q(t)}{C} \tag{17}$$

7 where d_p is the propagation delay, $q(t)$ the number of queued packets, and C is the capacity of all nodes along the path. Assuming that the number of packet drops can be modeled as a Poison process $N(t)$. When the packet-loss information arrives at rate $\lambda(t)$, then the TCP window dynamics is:

$$dW(t) = \frac{dt}{R(q(t))} - \frac{W(t)}{2} dN(t) \tag{18}$$

The derivative of the expectation of W is:

$$dE[W] = E\left[\frac{dt}{R(q)}\right] - \frac{E[W]\lambda(t)}{2} dt \tag{19}$$

The loss information reaches the source approximately after one round-trip delay *(τ)*. If the total node traffic load is *x(t)* and the packet drop rate is *p(x(t))*, then:

$$\lambda(t) = p(\bar{x}(t-\tau))E\left[\frac{W(t-\tau)}{R(q(t-\tau))}\right] \tag{20}$$

By using (20) and rewriting (19), the window dynamics is modeled as:

$$\frac{d\overline{W}}{dt} = \frac{1}{R(\bar{q})} - \frac{\overline{W}\,\overline{W}(t-\tau)}{2R(\bar{q}(t-\tau))}\,p(\bar{x}(t-\tau)) \tag{21}$$

The equation captures the interaction between the packet drop function *p(.)* and the congestion window dynamics.

$$\frac{d\bar{x}}{dt} = \frac{\log_c(1-\alpha)}{\delta}\bar{x}(t) - \frac{\log_c(1-\alpha)}{\delta}\bar{q}(t) \tag{22}$$

$$\frac{d\bar{q}(t)}{dt} \approx -C + \sum_{i=1}^{N}\frac{\overline{W}_i}{R_i(\bar{q})} \tag{23}$$

Equations (21), (22), and (23) form a system of differential equations with unknowns $(\bar{x}, \bar{q}, \overline{W}_i)$ whose solution gives an estimate of the average transient behavior of the system.

Based on the model, it was possible to evaluate the performance of RED using simulation. The results confirm that RED has a problem with its mean queue averaging mechanism.

4 Conclusion

The paper presents some analytical models for TCP congestion control. Though the study is for now confined to a set of relatively simple and straightforward cases of the TCP actions, it explains well the dynamics of the window behavior. For a single TCP connection, first the necessary expressions between the TCP bandwidth and the network loss probability are formulated. Then, the examination is extended to a presence of connection timeouts. The model is enriched with the interaction between the concept of AQM and TCP operation.

References

1. Allman, M., Paxson, V., Stevens, W., “TCP Congestion Control”, *RFC 2581*, Apr 1999
2. Chiu, D. M., Jain, R. “Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Networks”, *Computer Networks and ISDN Systems*, Vol. 17, pp. 1-14, 1991
3. Floyd, S., Jacobson, V., “Random Early Detection Gateways for Congestion Avoidance”. *IEEE/ACM Transactions on Networkin*g, 1(4):397–413, Aug 1993
4. Jacobson, V., “Congestion Avoidance and Control”, *Proc. of the ACM SIGCOMM Symposium on Communications Architectures and Protocol*s, pp. 314–329, Stanford, USA, 14-17 Aug 1988
5. Kurose, J., F., Ross, K., W., “Computer Networking-A Top-Down Approach Featuring the Internet”, Second Edition, *Pearson Education*, 2003
6. Mathis, M., Semke, J., Mahdavi, J., Ott, T., "The macroscopic behaviour of the TCP congestion avoidance algorithm". In *Computer Communication Review*, 1997.
7. Misra, V., Gong, W., Towsley D., F., "Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED". In *Proceedings of SIGCOMM*, pp. 151–160, 2000.
8. Padhye, J., Firoiu, V., Towsley, D., Kurose J., "Modeling TCP throughput: A simple model and its empirical validation". *Proceedings of the ACM SIGCOMM ’98*, pages 303–314, 1998.
9. Postel, J., “Transmission Control Protocol”, *RFC 793*, Sep 1981