Optimizing TCP Retransmission Timeout

Alex Kesselman[1,⋆] and Yishay Mansour[2]

[1] Max Planck Institut für Informatik,
Saarbrücken, Germany
akessel@mpi-sb.mpg.de

[2] School of Computer Science, Tel-Aviv University, Israel
mansour@cs.tau.ac.il

Abstract. Delay spikes on Internet paths can cause spurious TCP timeouts leading to significant throughput degradation. However, if TCP is too slow to detect that a retransmission is necessary, it can stay idle for a long time instead of transmitting. The goal is to find a Retransmission Timeout (RTO) value that balances the throughput degradation between both of these cases. In the current TCP implementations, RTO is a function of the Round Trip Time (RTT) alone. We show that the optimal RTO that maximizes the TCP throughput need to depend also on the TCP window size. Intuitively, the larger the TCP window size, the longer the optimal RTO. We derive the optimal RTO for several RTT distributions. An important advantage of our algorithm is that it can be easily implemented based on the existing TCP timeout mechanism.

1 Introduction

In most cases the Internet does not provide any explicit information about the network conditions. Thus, it is up to the transport protocol to form its own estimates, and then to use them to adapt as efficiently as possible. For these reasons congestion avoidance and control have become critical to the use of the Internet. Jacobson [12] pioneered the concepts of TCP congestion avoidance and control based on additive increase/multiplicative decrease (AIMD). This scheme allows to avoid the congestion collapse as shown by Floyd and Fall [9]. TCP was later augmented with fast retransmission and fast recovery algorithms to avoid inefficiency caused by retransmission timeouts [13, 25].

Despite the conventional wisdom of relying less on timeout-based mechanisms, it has been indicated that a large number of lost packets in the Internet is recovered by retransmission timeouts [3, 21]. The problem is that delays on Internet paths can be highly variable resulting for instance from route flipping [4, 2]. On the one hand, underestimation of RTT leads to a premature retransmission timeout in case there is no loss or the retransmission could be handled by the fast retransmission mechanism. On the other hand, overestimation of RTT leads to a late retransmission timeout, in case there is a loss that cannot be captured by the fast retransmission mechanism. Therefore, it is crucial for the TCP performance to find a Retransmission Timeout (RTO) value that is an equilibrium point balancing between both of these cases.

[⋆] The work of the first author was supported by AvH-Stiftung.

P. Lorenz and P. Dini (Eds.): ICN 2005, LNCS 3421, pp. 133–140, 2005.

Dolev et al. [6] study retransmission timeout setting for a simple transmission protocol by means of competitive analysis. Ludwig and Katz [17] propose the Eifel algorithm to eliminate the unnecessary retransmissions that can result from a spurious retransmission timeout. Gurtov and Ludwig [11] present an enhanced version of the Eifel algorithm and show its performance benefits on paths with a high bandwidth-delay product. Ekstr and Ludwig [7] propose a new algorithm for calculating the RTO, named the Peak-Hopper-RTO (PH-RTO), which improves upon the performance of TCP in high loss environments. Significant efforts have been also devoted to modeling such a complex protocol as TCP [16, 15, 18, 20].

In this paper we study how to find the optimal RTO maximizing the TCP throughput using the model of [20]. Our main contribution is to show that the optimal RTO need to depend on the TCP window size. We derive the optimal RTO as a function of RTT and the TCP window size for a general and some specific distributions of RTT. Intuitively, the larger the TCP window size, the longer the optimal RTO. We note that the heavy-tailed Pareto distribution has been shown to approximate most accurately the actual RTT distribution in the Internet [1, 2, 10, 5]. An important advantage of our algorithm is that it can be easily implemented on the top of the existing TCP timeout mechanism.

RFC 3649 [8] proposes a modification of TCP congestion control that adapts the increase strategy and makes it more aggressive for high bandwidth links (i.e. for large window sizes). In this work we demonstrate that for such scenarios TCP throughput could be further increased by selecting a larger RTO. Our results are strengthened by simulations in [11], which show that proper handling of spurious timeouts on paths with a high bandwidth-delay product can increase TCP throughput by up to 250%.

The rest of the paper is organized as follows. Summary of our results appears in Section 2. In Section 3 we describe the TCP model. Section 4 contains an analytic derivation of the optimal RTO. A general RTT distribution and some specific distributions are considered in Section 5 and Section 6, respectively.

2 Summary of Results

In this section we give an overview of our main results while the formal definitions and proofs are deferred to the following sections. We assume that RTT behaves like a random variable and derive the optimal retransmission timeout as a function of the mean and the variance of RTT and the TCP window size.

The input parameters to our algorithm are the RTT mean μ, the RTT variance σ^2 and the TCP window size W. (We assume that both μ and σ are finite.) Our goal is to find the optimal RTO maximizing the TCP throughput. We show that it is an increasing function on W.

First we obtain some upper bounds on the optimal RTO for a general RTT distribution. These bounds may be considered as worst-case bounds since they hold for any distribution. The results are presented in Table 1. We show that for any RTT distribution the optimal RTO is bounded from above by $W\sqrt{\log W}/3$ times the mean of RTT. Provided that higher moments of RTT exist, we establish bounds which are mostly driven by those moments while the effect of the window size becomes insignificant. Notice that when RTT is a fixed constant, we obtain an upper bound which tends to RTT.

Table 1. General distribution

Moment	RTO – Upper Bound
First moment	$\frac{1}{\sqrt{3}} W \sqrt{\log W} E[RTT]$
k'th moment	$\left(\frac{W^2 \log W}{3}\right)^{\frac{1}{k+1}} \left(E[RTT]E[RTT^k]\right)^{\frac{1}{k+1}}$

Table 2. Specific distributions

RTT Distribution	RTO – Optimal Value
Normal	$\mu + \sigma \cdot O\left(\sqrt{\ln W + \ln \frac{\mu}{\sigma}}\right)$
Exponential	$\mu \cdot O(\ln W)$
Pareto	$\left(\frac{W^2 \log W \mu}{3}\right)^{1-1/\mu}$

Next we derive the optimal RTO for some specific distributions. The corresponding results are presented in Table 2. Basically, we would like the probability of a premature retransmission timeout to be very small. The rational is that the throughput degradation due to a premature retransmission timeout is much higher than that due to a late retransmission timeout. Our model sets the probability of a premature retransmission timeout at about $1/W^2$, for optimizing the TCP throughput.

In case RTT is distributed according to the Normal distribution, one would expect the optimal RTO to be a sum of the mean plus the standard deviation times some factor, as our analysis indeed shows. The factor of μ/σ is due to the fact that when $RTO = \mu + \sigma \cdot A$, the expected number of rounds wasted as a result of a late retransmission timeout is $A \cdot \sigma/\mu$. This setting is similar to the RTO calculation of Jacobson [12] while the main difference is the dependence on the window size.

For the Exponential RTT distribution, we show that the optimal RTO is proportional to the mean of RTT and the logarithm of the window size. The logarithmic factor of the window size follows from the form of the density function.

Finally, we consider the heavy-tailed Pareto distribution of RTT and establish that the optimal RTO is the mean of RTT multiplied by a power of the window size. Such a dependence is due to the heavy-tail property of the Pareto distribution.

3 TCP Model

We adopt the model of [20] that is based on Reno version of TCP. The TCP's congestion avoidance behavior is modeled in terms of "rounds." The duration of a round is equal to the round trip time and is assumed to be independent of the window size. We define a round of TCP to be a time period starting from transmitting a window of W packets back-to-back and ending upon receiving the acknowledgments for these packets.

We make the following simplifying assumptions. There is always data pending at the sender, such that the sender can always transmit data as permitted by the congestion window while the receiver's advertised window is sufficiently large to never constrain the congestion window. Every packet is assumed to be individually acknowledged (the delayed acknowledgment algorithm is not in effect). A packet is lost in a round independently of any packets lost in other rounds. However, packet losses are correlated among the back-to-back transmissions within a round: if a packet is lost, all the consequent packets transmitted until the end of that round are also lost[1]. We define packet loss probability p to be the probability that a packet is lost, given that either it is the first packet in a round or the preceding packet in the round is not lost.

We call the congestion avoidance phase a *steady state*. We assume that timeout expiration does not occur during a slow start phase and concentrate on the timeout setting in a steady state. We also assume that the mean and the variance of RTT are available or could be estimated.

We approximate the packet loss probability as a function of the TCP window size in a steady state, which is a simplification of [20], as $p \approx \frac{1}{W^2}$. We note that the model of [20] captures the effect of TCP's timeout mechanism on throughput.

4 TCP Timeout Optimization

In this section we consider optimization of the retransmission timeout. The goal is to maximize the throughput of TCP. Notice that the optimal RTO is the actual RTT, which is unknown to our algorithm. Thus, the online decision must be based only on the available estimates of the mean and the variance of RTT. We try to find the value of RTO that balances throughput degradation between a premature retransmission timeout and a late retransmission timeout, which are so called “bad events” (that will be formally defined later). Recall that in our model bad events occur only in a steady state.

When a bad event happens, we consider the *convergence period* T during which TCP reaches a steady state. We compare the throughput of TCP during T with that of an optimal algorithm that uses the actual RTT as its RTO and sends in average W packets every round. We call to the number of extra packets sent by the optimal algorithm during T *throughput degradation*. The goal is to minimize the expected throughput degradation due to bad events.

First we will derive the expected duration of the convergence period. In the case of a premature retransmission timeout, it takes exactly $\log W$ rounds for TCP to reach a steady state since the TCP window grows exponentially during a slow start phase. In the case of a late retransmission timeout, TCP is idle instead of transmitting during $RTO - RTT$ time units. Thus, the expectation of the length of T in rounds is:

$$E[length(T) \,|\, RTO > RTT] = \frac{1}{P[RTO > RTT]} \int_0^{RTO} (RTO - RTT) dRTT.$$

We approximate the expected number of rounds using the Law of Large Numbers as $E[length(T)]/\mu$:

$$E[\#\text{ rounds in } T \,|\, RTO > RTT] \approx \frac{1}{P[RTO > RTT]} \int_0^{RTO} \frac{RTO - RTT}{\mu} dRTT.$$

[1] Such situation naturally occurs when the tail-drop policy is deployed by the bottleneck router.

Assuming that there is a sequence of one or more losses in a given round, the probability of retransmission timeout is $\min(1, \frac{3}{W})$ [20]. In the sequel, we assume that $W > 3$. Next we will define the bad events more formally.

Premature retransmission timeout. We say that a timeout occurred *prematurely* if no packet in the round is lost or the loss can be captured by the fast retransmission mechanism. Note that RTO must be smaller than RTT. The probability of this event is:

$$\begin{aligned} P_1 &= P[RTO < RTT] \cdot \left((1-p)^W + \left(1 - (1-p)^W\right) \left(1 - \frac{3}{W}\right) \right) \\ &\approx P[RTO < RTT]. \end{aligned}$$

The throughput degradation due to this event is: $L_1 = W \log W$. Observe that during the slow start phase, TCP sends at most W packets. We obtain that the expected throughput degradation as a result of a premature retransmission timeout is:

$$P_1 \cdot L_1 = P[RTO < RTT] \cdot W \log W.$$

Late retransmission timeout. We say that a timeout occurred *lately* if some packets in the round are lost and the loss cannot be captured by the fast retransmission mechanism. Note that RTO must be larger than RTT. The probability of this event is:

$$P_2 = P[RTO > RTT] \cdot \left(1 - (1-p)^W\right) \frac{3}{W} \approx P[RTO > RTT] \frac{3}{W^2}.$$

The throughput degradation due to this event is:

$$L_2 = W \frac{1}{P[RTO > RTT]} \cdot \int_0^{RTO} \frac{RTO - RTT}{\mu} dRTT.$$

We get that the expected throughput degradation as a result of a late retransmission timeout is:

$$P_2 \cdot L_2 = \frac{3}{W} \int_0^{RTO} \frac{RTO - RTT}{\mu} dRTT.$$

The optimal RTO, RTO^*, minimizes the expected throughput degradation, that is:

$$P_1(RTO) \cdot L_1(RTO) + P_2(RTO) \cdot L_2(RTO).$$

Thus, given the probability distribution of RTT, the optimal RTO minimizes:

$$P[RTO < RTT] \cdot W \log W + \frac{3}{W} \int_0^{RTO} \frac{RTO - RTT}{\mu} dRTT.$$

For simplicity, we will derive an approximation for the optimal RTO, the balanced RTO^{**}, for which the expected throughput degradation is the same for both of the bad events:

$$P[RTO < RTT] \cdot W \log W = \frac{3}{W} \int_0^{RTO} \frac{RTO - RTT}{\mu} dRTT). \qquad (1)$$

Note that in the worst case the expected throughput degradation for the balanced RTO is at most twice as large as that for the optimal RTO.

5 General Distribution

In this section we study what is the worst case effect of the TCP window size on the maximal value of the optimal RTO. We derive upper bounds on the optimal RTO that hold for any distribution of RTT. In our analysis we use a simplified form of (1): $P[RTO < RTT]W\log W = \frac{3}{W}\frac{RTO}{\mu}$.

First we show that for any RTT distribution with finite mean, the optimal RTO is bounded from above by $W\sqrt{\log W}/3$ times the mean of RTT. Applying Markov inequality to (1) we get: $\frac{\mu}{RTO}W\log W \geq \frac{3}{W}\frac{RTO}{\mu}$, and thus $RTO \leq \frac{1}{3}W\sqrt{\log W}\mu$.

In case higher moments of RTT exist, applying the general form of Chebyshev inequality and using (1) we obtain an upper bound that depends on both those moments and the window size: $\frac{E[RTT^k]}{RTO^k}W\log W \geq \frac{3}{W}\frac{RTO}{\mu}$, and we obtain

$$RTO \leq \left(\frac{1}{3}W^2\log W\right)^{\frac{1}{k+1}}\left(E[RTT]E[RTT^k]\right)^{\frac{1}{k+1}}.$$

Notice that when RTT is almost constant, that is $E[RTT^k] \approx \mu^k$, for sufficiently large k the resulting upper bound tends to μ.

6 Specific Distributions

In this section we study the case in which RTT is distributed according to a given known distribution and derive the optimal value of RTO for some well-known distributions.

6.1 Normal Distribution

In this section we consider the Normal distribution of RTT with the mean μ and the variance σ^2, the density function $f(x) = \frac{1}{\sigma\sqrt{2\pi}}e^{-(x-\mu)^2/2\sigma^2}$ and distribution function $F(x) = \Phi(\frac{x-\mu}{\sigma})$. To avoid negative values, we can take RTT to be $\max(D, N(\mu,\sigma))$ for some $D < \mu$, which does not really affect the analysis that is concentrated on the tail of RTT values larger than μ. Since the Normal distribution is invariable under transforming the mean, one would expect the RTO bound to be a sum of the mean plus the standard deviation times some factor, which is indeed the case as we show.

Substituting to (1), $P[RTO < RTT] = 1 - \Phi(\frac{RTO-\mu}{\sigma})$, $E[RTT] = \mu$, $d(RTT) = \frac{1}{\sigma\sqrt{2\pi}}e^{-(x-\mu)^2/2\sigma^2}dx$ and $y = \frac{x-\mu}{\sigma}$ we obtain:

$$\left(1 - \int_0^r \frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}dy\right)W\log W =$$

$$\frac{3}{W\mu}\left(RTO\int_0^r \frac{1}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}dy - E[RTT \mid RTT < RTO]\right).$$

Provided that RTT is sufficiently large, we can assume that $E[RTT|RTT < RTO] \approx \mu$. Having done some calculations, we derive the following RTO: $RTO = \mu + \sigma \cdot O\left(\sqrt{\ln W + \ln\frac{\mu}{\sigma}}\right)$.

The interesting factor is $O(\sqrt{\ln W})$, which guarantees that the probability of a premature retransmission timeout is small.

6.2 Exponential Distribution

In this section we consider the Exponential distribution of RTT with the rate parameter λ, the mean $E[x] = 1/\lambda$, the density function $f(x) = \lambda e^{-\lambda x}$ and the distribution function $F(x) = 1 - e^{-\lambda x}$. We show that the optimal RTO is proportional to the mean of RTT and the logarithm of the TCP window size.

Substituting to (1), $P[RTO < RTT] = e^{-\lambda RTO}$, $E[RTT] = 1/\lambda$ and $d(RTT) = \lambda e^{-\lambda x} dx$ we get:

$$e^{-\lambda RTO} W \log W = \frac{3}{W} \lambda \left(\left(1 - e^{-\lambda RTO}\right) RTO - \int_0^{RTO} x \lambda e^{-\lambda x} dx \right).$$

This gives us the following RTO: $RTO \approx \frac{1}{\lambda} \ln \left(\frac{W^2 \log W}{3} \right) = \frac{1}{\lambda} O(\ln W)$. The logarithm of W achieves the effect of setting the premature retransmission timeout probability to be order of $1/W^2$.

6.3 Pareto Distribution

In this section we consider the heavy-tailed Pareto distribution of RTT with the shape parameter $a > 1$, the mean $E[x] = \frac{a}{a-1}$, the density function $f(x) = \frac{a}{x^{a+1}}$ and the distribution function $F(x) = 1 - (\frac{1}{x})^a$. We show that the optimal RTO is the mean of RTT multiplied by a power of the window size, which is due to the heavy-tail property of the Pareto.

Substituting to (1), $P[RTO < RTT] = \left(\frac{1}{RTO}\right)^a$, $E[RTT] = \frac{a}{a-1}$ and $d(RTT) = \frac{a}{x^{a+1}} dx$ gives us:

$$\left(\frac{1}{RTO}\right)^a W \log W = \frac{3}{W} \frac{a-1}{a} \cdot \left(\left(1 - \left(\frac{1}{RTO} \right)^a \right) RTO - \int_1^{RTO} RTT \frac{a}{x^{a+1}} dx \right).$$

Solving this equation derives the following RTO: $RTO \approx \left(\frac{W^2 \log W \mu}{3} \right)^{1-1/\mu}$. An interesting setting is $a = 2$ where $E[RTT] = 2$. In this case we get that $RTO \approx W\sqrt{\log W}$, which justifies the form of the bound we have for an arbitrary distribution.

References

1. A. Acharya and J. Saltz, "A Study of Internet Round-trip Delay," *Technical Report CS-TR-3736*, University of Maryland, December 1996.
2. M. Allman and V. Paxson, "On Estimating End-to-End Network Path Properties," *In Proceedings of SIGCOMM '99*, pp. 263-274.
3. H. Balakrishnan, S. Seshan, M. Stemm, and R. H. Katz, "Analyzing Stability in Wide-Area Network Per-formance," *In Proceedings of SIGMETRICS'97.*
4. J. C. Bolot, "Characterizing End-to-End Packet Delay and Loss in the Internet," *Journal of High Speed Networks*, 2(3), September 1993.
5. C. J. Bovy, H. T. Mertodimedjo, G. Hooghiemstra, H. Uijterwaal and P. Van Mieghem, "Analysis of End to end Delay Measurements in Internet," *In Proceedings of PAM 2002*, March 2002.

6. S. Dolev, M. Kate and J. L. Welch, "A Competitive Analysis for Retransmission Timeout," *15th International Conference on Distributed Computing Systems*, pp. 450-455, 1995.
7. H. Ekstr and R. Ludwig, "The Peak-Hopper: A New End-to- End Retransmission Timer for Reliable Unicast Transport," *In Proceedings of IEEE INFOCOM 04*.
8. S. Floyd, "HighSpeed TCP for Large Congestion Windows," *RFC 3649*, December 2003.
9. S. Floyd and K. Fall, "Promoting the Use of End-to-end Congestion Control in the Internet," *IEEE/ACM Transactions on Networking*, August 1999.
10. K. Fujimoto, S. Ata and M. Murata, "Statistical analysis of packet delays in the internet and its application to playout control for streaming applications," *IEICE Transactions on Communications*, E84-B, pp. 1504-1512, June 2001.
11. A. Gurtov, R. Ludwig, "Responding to Spurious Timeouts in TCP," *In Proceedings of IEEE INFOCOM'03*.
12. V. Jacobson, "Congestion Avoidance and Control," *In Proceedings of SIGCOMM'88*.
13. V. Jacobson, "Modified TCP congestion avoidance algorithm," *end2end-interest mailing list*, April 30, 1990.
14. P. Karn and C. Partridge, "Improving Round-TripTime Estimates in Reliable Transport Protocols," *In Proceedings of SIGCOMM '87*, pp. 2-7, August 1987.
15. A. Kumar, "Comparative Performance Analysis of Versions of TCP in a Local Network with a Lossy Link," *IEEE/ACM Transactions on Networking*, 6(4):485-498, August 1998.
16. T. V. Lakshman and U. Madhow, "The Performance of TCP/IP for Networks with High Bandwidth-Delay Products and Random Loss," *IEEE/ACM Transactions on Networking*, 3(3):336-350, June 1997.
17. R. Ludwig, and R. H. Katz, "The Eifel Algorithm: Making TCP Robust Against Spurious Retransmissions," *ACM Computer Communication Review*, 30(1), January 2000.
18. M. Mathis, J. Semske, J. Mahdavi, and T. Ott, "The macroscopic behavior of the TCP congestion avoidance algorithm," *Computer Communication Review*, 27(3), July 1997.
19. T. Ott, J. Kemperman, and M. Mathis, "The stationary behavior of ideal TCP congestion avoidance," November, 1996.
20. J. Padhye, V. Firoiu, D. Towsley and J. Kurose, "Modeling TCP Throughput: A Simple Model and its Empirical Validation," *In Proceedings of SIGCOMM'98*.
21. V. Paxson, "End-to-End Internet Packet Dynamics," *In Proceedings of SIGCOMM'97*.
22. V. Paxon and M. Allman, "Computing TCP's Retransmission Timer," *RFC 2988*, November 2000.
23. J. Postel, "Transmission Control Protocol," *RFC 793*, September 1981.
24. P. Sarolahti and A. Kuznetsov, "Congestion Control in Linux TCP," *In Proceedings of the USENIX Annual Technical Conference*, June 2002.
25. W. R. Stevens, "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms," *RFC 2001*, January 1997.
26. L. Zhang, "Why TCP timers don' t work well," *In Proceedings of SIGCOMM'86*.