International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 12 (2017) pp. 3365-3369

Lower Bound for Mean Object Transfer Latency in the Narrowband IoT Environment

Y. –J. Lee
Department of Technology Education, Korea National University of Education,
250 Taesungtapyon-ro, Heungduk-ku, Cheongju, Chungbuk, South Korea.

ORCID: 0000-0003-4555-3285

Abstract

This paper presents analytical model and algorithm to find the lower bound of mean web object transfer latency, which is one of important measures of quality of Service (QoS) in the Internet. The proposed mean latency model assumes the multiple packet losses and the narrowband IoT (Internet of Things) environment including multi-hop wireless network, where fast retransmission is not possible due to small window. Our model also considers the initial congestion window size and the multiple packet loss in one congestion window. Computational experiments show that for a given packet loss rate, round trip time and initial congestion window size mainly affect the mean web object transfer latency. The model can be applied to determine the web object size satisfying mean response time that end user requires and select the neighbor node with the least transfer latency in the narrowband IoT environment.

Keywords: Web object latency, TCP congestion control, Packet loss, Narrowband IoT

INTRODUCTION

The web object transfer latency is one of many important measures in the Internet since it is main factor to affect end-end delay. Web applications mainly make use of transmission control protocol (TCP), which uses a single stream preserving byte order in the stream by assigning a sequence number to each packet.

When any loss occurs in a certain packet, the subsequent packets must wait to be delivered to users until the lost packet will be retransmitted. This retransmission increases transfer latency. Typically, object transfer latency is affected by data size and transmission time according to transmission rate of link as well as by TCP congestion control mechanism. The common functions of TCP congestion control mechanism are slow start, congestion avoidance, timeout, and fast retransmission [1].

Previous related works on analytical models of data transfer delay over TCP are as following: Padhye [2] considered large amount of data transfer on steady state over TCP. Most of TCP connections for web application, however, are short for small amount of data instead of large one in current Internet environment. Connection setup or slow-start time dominates the performance of web in this environment.

Noticing this phenomenon, Cardwell [3] extended the above steady state model but he did not consider delay of TCP after time-out. Jiong [4] enhanced the model of [3] considering slow-start time after timeout of retransmission.

However, since the above models assumed wideband network, they are not able to be applied to the narrowband IoT environment, which this paper considers. That is why narrowband IoT environment and multi-hop wireless network do not allow fast retransmission of data due to the very small size of window [5].

Lee [6,7] proposed the web object latency model for HTTP over TCP, T-TCP, and SCTP in the data network and dealt with the SCTP handover scheme over mobile Internet [8,9]. However they and other previous works did not consider the effect of the initial window size and the multiple packet losses in a single window in the IoT environment.

Our model estimates the lower bound (LB) when all the packet losses occur in the last window during slow start for the transfer completion. The LB of mean object transfer latency can be found by using our iterative algorithm based on the packet loss rate, the initial congestion window size, the link bandwidth, and round trip time (RTT). Our model and algorithm can be used in estimating the mean object delay and determining the web object size satisfying end-user's required latency in the narrowband IoT environment and multi-hop wireless network.

The rest of the paper is organized as follows. Section 2 describes modeling for mean web object transfer latency. Section 3 discusses iterative algorithms and computational experiences for mean web object transfer latency. Section 4 concludes with future works.

MODELLING FOR WEB OBJECT TRANSFER LATENCY

In order to simplify the model we assume that sizes of web objects are identical and packets are transmitted in terms of

congestion window size unit. Let the size of a web object to transfer be θ Bytes and sender maximum segment size (SMSS) be *smss* Bytes. Then, the number of packets to transfer for an object is $N=\lceil \theta/smss \rceil$. When the probability of a packet loss is *p*, the expected number of total packet losses is $\alpha=\lceil n\ p \rceil$ by binomial distribution.

Congestion control period for the narrowband network is composed of the slow start period and congestion avoidance period, because fast retransmit and fast recovery are not possible. Therefore, any packet loss occurs during either slow start or congestion avoidance period.

We can identify where any k^{th} (k=1,2.. a) packet loss occurs by comparing the maximum number of packets (A_k, k=1,2.. a) to be transferred until the threshold (TH_k, k=1,2,..,a) at which congestion avoidance starts, with the expected number of packets (X_k: k=1,2,..,a) to be transmitted before the packet loss. That is, if $X_k \le A_k$, packet is lost during slow start period, otherwise ($X_k > A_k$), packet is lost during congestion avoidance period.

For the data to be transmitted before k^{th} packet loss, N_k (N_k=N for k=1), the expected number of packets sent (including the lost packet) until the packet loss is given by

$$X_k = \frac{1-(1-p)^{N_k}}{p} + (1-p)^{N_k} + 1 \qquad k = 1,2,\cdots,\alpha \tag{1}$$

We first consider the worst case in which the amount of retransmission can increase, even if we consider the multiple losses within one congestion window. In this case, the estimation of expected number of packets sent before the packet loss (X_k) in (1) is fairly distributed over the transmission time. That is, packet errors are spread over time depending on the data amount to be transferred and the packet loss.

The initial value of congestion window (*cwnd*) (*IW*) is suggested as 2×*smss*, 3×*smss*, and 4×*smss* [1]. Initial slow start threshold (TH_1) is set arbitrarily high (∞) and TH_k ($k \ge 2$) are set to

$$TH_k = \max(\frac{FlightSize}{2},\ 2\times smss) \qquad k = 2,3,..,\alpha \tag{2}$$

Here, *FlightSize* represents the amount of data that has been sent but not yet cumulatively acknowledged. In our paper, we set *FlightSize* to *cwnd* by considering worst case.

A_k (k=1,2,...,a) is the maximum number of packets to be sent until TH_k. Since $TH_1=\infty$, A_1 is also equal to ∞. Thus, X_1 is less than A_1. This means that first packet loss (k =1) must occur during slow start.

Packets are transmitted in the manner *IW*, 2×*IW*, 4×*IW*, 8×*IW*,…(*IW*=2,3,4) for k=1 and *IW*×1, *IW*×2, *IW*×4, *IW*×8,…(*IW*=1) for $k \ge 2$, respectively. Therefore, A_k is given by

$$A_k = \begin{cases} 2^{\left\lceil \log_2^{TH_{k+1}} \right\rceil} - IW & if\ \ TH_k = 2^j \\ 2^{\left\lceil \log_2^{TH_{k+1}} \right\rceil} - IW + TH_k & if\ \ TH_k \ne 2^j \end{cases} \tag{3}$$

Generally, we need n windows in order to completely send the object. Generally n can be expressed in terms of the transmission data amount (Y) and initial window size (*IW*).

$$\begin{aligned} n &= \min\left\{q : IW\left(2^0 + 2^1 + \cdots + 2^{q-1}\right)\right\} \ge Y \\ &= \left\lceil \log_2(1+\frac{Y}{IW}) \right\rceil \end{aligned} \tag{4}$$

Because X_k is sent until k^{th} packet loss, the window number (n_k) which includes X_k is given by

$$n_k = \left\lceil \log_2(1+\frac{X_k}{IW}) \right\rceil \qquad \begin{cases} IW = 2,3,4 & for\ \ k=1 \\ IW = 1 & for\ \ k \ge 2 \end{cases} \tag{5}$$

Congestion window size (C_k) corresponding to the window number (n_k) is

$$C_k = IW \times 2^{n_k - 1} \qquad \begin{cases} IW = 2,3,4 & for\ \ k=1 \\ IW = 1 & for\ \ k \ge 2 \end{cases} \tag{6}$$

The maximum number of packets sent until n_k^{th} window is given by

$$B_k = IW\left(\sum_{j=0}^{n_k-1} 2^j\right) = IW \times (2^{n_k} - 1) \qquad \begin{cases} IW = 2,3,4 & for\ \ k=1 \\ IW = 1 & for\ \ k \ge 2 \end{cases} \tag{7}$$

By considering the initial congestion window size (*IW*) additionally, the number of receiver stalls (m) when the object contains an infinite number of segments [8] is given by

$$\begin{aligned} m &= \max\{q : \frac{smss}{\mu} + rtt - \frac{smss}{\mu} \times IW \times 2^{q-1} \ge 0\} \\ &= \left\lfloor \log_2\left(1+\frac{\mu \times rtt}{smss}\right) \right\rfloor + 1 - \log_2 IW \end{aligned} \tag{8}$$

Therefore, when the transmission data amount (Y) and the initial congestion window size (*IW*) are given, slow start time is

$$\begin{aligned} &ST(Y) = \rho(rtt + \frac{smss}{\mu}) - IW(2^\rho - 1)\frac{smss}{\mu} \\ &where \\ &\rho = \min(m, n-1) \\ &= \min\left(\left\lceil \log_2\left(1+\frac{Y}{IW}\right)\right\rceil - 1,\ \left\lfloor \log_2\left(1+\frac{\mu \times rtt}{smss}\right)\right\rfloor + 1 - \log_2 IW\right) \end{aligned} \tag{9}$$

Here, μ and *rtt* represent the link bandwidth and round trip time between sender and receiver respectively.

Now we consider the amount of transmission data (Y), retransmission data (R), and the remaining data for transfer before the next packet loss (N_{k+1}) when multiple packet losses occur in one window after k^{th} packet loss. From Equation (2), (3), and (7), it is clear that $X_k \le A_k$, $X_k \le N_k$, and $X_k \le B_k$.

Therefore, mean web object transfer latency when the k^{th} packet loss occurs during slow start is the sum of slow start time of Y, transmission time of Y, and retransmission timeout as in (10).

$$OT_k^{slow} = ST(Y) + \frac{Y \times smss}{\mu} + R_{to} \quad (10)$$

Retransmission timeout (R_{to}) is mostly given by $3/2 \times rtt$, which can be adjusted according to real environment. At the next step, we compute X_{k+1} in Eq. (1) by using N_{k+1}.

Mean web object transfer latency when the k^{th} packet loss occurs during congestion avoidance is the sum of slow start time of A_k, additional (M-1) round trip time, transmission time of X_k, and retransmission timeout. Here, R_{to} is $3/2 \times rtt$ and can be adjustable.

$$OT_k^{cong} = ST(A_k) + (M-1) \times rtt + \frac{X_k \times smss}{\mu} + R_{to} \quad (11)$$

After α packet losses are processed in either during slow start or congestion avoidance by using the above mentioned model, data for transferring might be still remained. At this time, the remaining data ($N_{\alpha+1}$) is greater than 0 and since there is no more packet loss, timeout (R_{to}) in Equation (10) and (11) is not necessary. We can therefore simply find the transfer latency, LT^{slow} during slow start (if $N_{\alpha+1} \le A_{\alpha+1}$) or LT^{cong} during conges $LT^{slow}(N_{\alpha+1})$ *and* $LT^{cong}(N_{\alpha+1})$ are given by

$$LT^{slow}(N_{\alpha+1}) = ST(N_{\alpha+1}) + \frac{N_{\alpha+1} \times smss}{\mu} \quad if\ N_{\alpha+1} \le A_{\alpha+1} \quad (12)$$

$$LT^{cong}(N_{\alpha+1}) = ST(A_{\alpha+1}) + (M-1) \times rtt + \frac{N_{\alpha+1} \times smss}{\mu} \quad otherwise$$

After packet losses, α packets can be transferred either during slow start or during congestion avoidance without error in the last phase. Thus, the LB of mean web object transfer latency with the number of packets, $N = \lceil \theta/smss \rceil$ (object size = θ and sender MSS = $smss$) is given by Eq. (18).

$$\underline{OT}_\theta^{smss} = ST(N) + \gamma \times LT^{slow}(\alpha) + (1-\gamma) \times LT^{cong}(\alpha) + \frac{(N+\alpha) \times smss}{\mu} + R_{to} \quad where\ \gamma = 0\ or\ 1 \quad (13)$$

ITERATIVE ALGORITHMS FOR WEB OBJECT TRANSFER LATENCY

Based on the above model and recent TCP congestion control standard (RFC2581) [1], we can build an algorithm to estimate the LB of mean latency for web object transfer as in Figure 1.

ALGORITHM 1. LB of mean latency for web object transfer

01: **INPUT:** p, θ, $smss$, rtt, μ

02: **OUPUT:** $\underline{OT}_\theta^{smss}$

03: **BEGIN**

04. Compute the total number of packets included in a web object,

$N = \lceil \theta/smss \rceil$

05: Compute the expected number of packet losses, $\alpha = \lceil np \rceil$

06. Set $N_1 = N$, $TH_1 = A_1 = \infty$

07: Set $\underline{OT}_\theta^{smss} = 0$ and $k = 0$

08: **while (1)**

09: **if** (k >= 1 or p = 0)

10: **begin**

11: Compute A_{k+1} using Eq. (3) satisfying TH_{k+1}

12: **if** $\alpha \le A_{k+1}$

13: Set $\underline{OT}_\theta^{smss} = \underline{OT}_\theta^{smss} + ST(\alpha) + \alpha \times smss/\mu$

14: **else**

15: Set $\underline{OT}_\theta^{smss} = \underline{OT}_\theta^{smss} + ST(A_{k+1}) + (M\text{-}1) \times rtt + \alpha \times smss/\mu$

16: **end**

17: **break;**

18: k++;

19: Compute the expected number of packets sent including the lost packet until the packet loss, $X_k = N - \alpha + 1$;

20: Compute the number of packets sent until slow start (A_k) by using and (3)

21: **if** ($X_k \le A_k$)

22: **begin**

23: Compute the window number (n_k) and *cwnd* (C_k) using (5) and (6), respectively

24: Set $TH_{k+1} = \max\{C_k/2, 2\}$ by Eq. (11)

25: Set $\underline{OT}_\theta^{smss} = \underline{OT}_\theta^{smss} + ST(N) + N \times smss/\mu + 3/2 \times rtt$

26: **end**

27: **end while**

28: Find the LB of mean latency for web object transfer ($\underline{OT}_\theta^{smss}$)

29: **END**

Figure 1: Mean web object transfer latency algorithm for LB

We compute the LB of mean web object transfer

latency. Generally, mean web object size (θ) is known to be 13.5KB and sender MSS (*smss*) is 536B for WAN. If *smss* ≤ 1095B, the initial value of *cwnd* (*IW*) is set to 4×*smss* as an upper bound [1].

First, we compute mean web object transfer latency by varying the web object size (θ) when the packet loss rate (p) is 0.0, 0.01and 0.05. Round trip time (*rtt*) is 0.1 second and transmission rate of link (μ) is 10Mbps.

Second, we fix the round trip time (*rtt*) and the initial congestion window (*IW*) for k =1 as 256ms and 2×*smss*, respectively. And then, we change the packet loss rate (p) from 0 to 0.2.

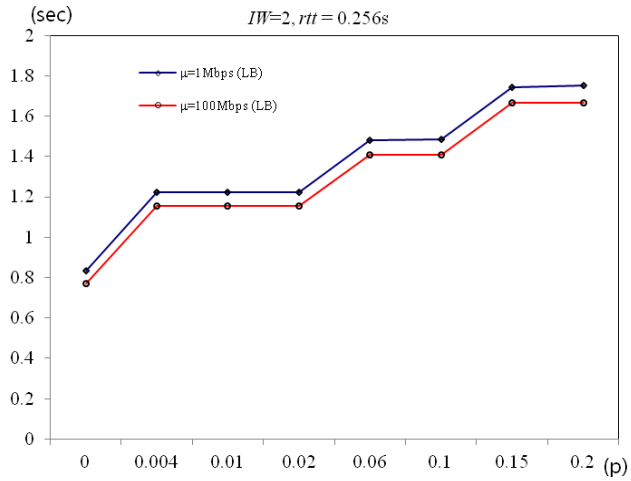

Figure 2: Mean web object transfer latency for varying the link speed (μ)

Figure 2 depicts the LB of mean web object transfer latency when the transmission rate of link (μ) is 1Mbps and 100Mbps. In particular, we can investigate that there is no much reductions in mean transfer latency although the transmission rate of link (μ) is increased from 1Mbps to 100Mbps.

Third, we fix the transmission rate of link (μ) and the initial window (*IW*) as 10Mbps and 2×*smss*, respectively. We also vary the packet loss rate (p) from 0 to 0.2.

Figure 3 depicts the LB of mean web object transfer latency when the round trip time (*rtt*) is 0.1second and 1.5second. As we can see in the figure, mean latency is largely affected by *rtt*. This is why slow start time significantly increases when the round trip time is relatively large. The LB when *rtt* = 0.1s is larger than the LB when *rtt* is 1s for $p \leq 0.6$.

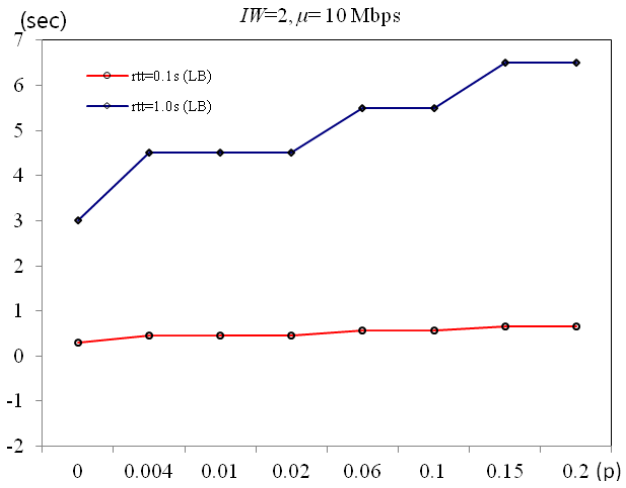

Figure 3: Mean web object transfer latency for varying the round trip time (*rtt*)

Finally, we vary the packet loss rate (p) from 0 to 0.2 after fixing the transmission rate of link (μ) and the round trip time (*rtt*), as 10Mbps and as 0.256sec, respectively. We change the initial window size (IW) from 1×*smss to* 4×*smss*.

Figure 4 shows the LB of mean web object transfer latency in this case.

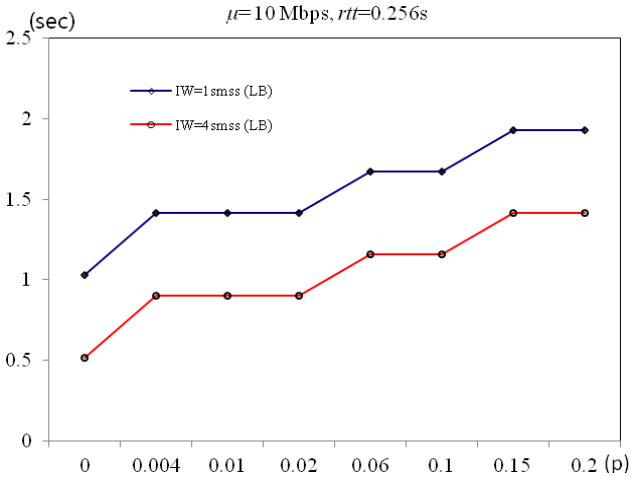

Figure 4: Mean web object transfer latency for varying the initial window size (*IW*)

CONCLUSIONS

In this paper, we present the analytical model and algorithm to find the lower bound (LB) of the mean web object transfer latency in the narrowband IoT environment and multi-hop wireless network. LB is the mean latency when all the packet losses occur in the last one window during slow start, respectively. Our model iteratively finds the latency based on

the packet loss rate and the number of packets to be transmitted. It also considers the initial value of congestion window and multiple packet losses in one window. The proposed algorithm can easily find the mean latency when the packet loss rate, web object size, SMSS, RTT, and the link rate are given.

Computational experiences show that at given the packet loss rate, round trip time and initial window size mainly affect the web object transfer latency. Our algorithm can be applied to determine the web object size satisfying end user's target response time and selecting the neighbor node with the least transfer delay in the narrowband IoT environment. Our work assumed single user and dealt with extreme packet loss cases-Lower Bound in order to simplify the model. Future works include more accurate model considering the probability distribution of burst errors in multiple user environment.

REFERENCES

[1] Allman, M., Paxson, V., and Blanton, E., 2009, "TCP congestion control," RFC-2581.

[2] Padhye, J., Firoiu, V., Towsley, D., and Kurose, J., 2000, "Modeling TCP reno performance: A simple model and its empirical validation," ACM Transactions on Networking, 8(2), pp. 133-145.

[3] Cardwell, N, Savage, S., and Anderson, Y., 2000, "Modeling TCP latency,", Proc. IEEE Infocom Conference, pp. 1742-1751.

[4] Jiong, Z., Shu-jing, Z., and Qi-gang, 2002, "An adapted full model for TCP latency," Proc. IEEE TENCON Conference, pp. 801-804.

[5] Oliveria, D., and Braun, R., 2005, "A dynamic adative acknowledgement strategy for TCP over multihop wireless networks," Proc. IEEE INFOCOM Conference, pp. 1863-1874.

[6] Lee, Y., and Atiquzzaman, M., 2009, "Mean waiting delay for web object transfer in wireless environment," Proc. IEEE International Conference on Communications, pp. 1-5.

[7] Lee, Y., 2015, "Novel quality of service measure for web transaction in multiple user access environments," International Journal of Applied Engineering Research, 10(16), pp. 37439-37444.

[8] Lee, Y., 2016, "Some considerations for SCTP handover scheme in mobile network," International Journal of Applied Engineering Research, 11(11), pp. 7526-7531.

[9] Lee, Y., 2016, "Location management agent for SCTP handover in mobile network," International Journal of Applied Engineering Research, 11(11), pp. 7532-7536.