Comparison of Weighted-Average and Median Filters for Wireless Retransmission Timeout Estimation

Auc Fai Chan, John Leis
Faculty of Engineering and Surveying
University of Southern Queensland
Toowoomba Queensland 4350

***Abstract*—Estimation of retransmission timeout (RTO) in TCP networks is important in order to maximize throughput. If the RTO is too small, packets will be unnecessarily declared lost, thus resulting in a retransmission. If RTO is too large, applications will wait too long before declaring a packet to be lost, thus reducing throughput. The conventional method due to Karn and Jacobson for estimating RTO is using a smoothing filter [1], [2], and this was later refined to utilize mean deviation measurements [1], [2]. In wireless links, the delay can often be impulsive, and weighted-average filters perform poorly. This paper presents results for mobile wireless transmission, and investigates the use of a median filter for timeout estimation in the presence of competing traffic flows. We demonstrate the applicability of this approach to estimating TCP round-trip times, and UDP packet jitter.**

***Index Terms*—median filter TCP congestion wireless**

I. Introduction

As the Internet expands, the number of TCP segments to be transmitted increases greatly, and consequently, there is congestion. Researchers have designed two different approaches to address the problem of transmission congestion. The first method is to modify the existing TCP/IP, examples of which are sliding windows and Random Early Drop [3]. The second method is to design a new protocol, such as Snoop Protocol [4]. The aim of this paper is to accurately estimate round-trip time (RTT) (which in turn affects RTO). All these will be described in following sections.

II. Transmission Control Protocol/Internet Protocol (TCP/IP)

In early 1970s, the Transmission Control Protocol/Internet Protocol (TCP/IP) was developed by Defense Advanced Research Projects Agency (DARPA), an agency inside the U.S. Department of Defense [5].

TCP resides in the transport layer of the conventional seven-layer Open Systems Interconnection (OSI) model. It is above the IP layer and below the upper layers. TCP is not loaded into router (also called gateway in some books) to support user data transfer. It resides in the host computer.

TCP provides the following services to upper layers:

- Connection-oriented data management;
- Reliable data transfer;
- Resequencing;
- Flow control (sliding window); and
- Precedence (i.e. priority) and security [6].

III. Acknowledgement (ACK) and Sliding Window

TCP uses a technique called positive acknowledgement with retransmission to make a connection reliable.

When a packet of data arrives at the receiver site, the receiver site transmits a positive acknowledgement (ACK) message to the sender site. The sender site will transmit a new packet of data only when the ACK message is received. The sender will start a timer when it transmits a packet of data.

When the timer expires before ACK message is received, the sender site will transmit the packet of data again.

To prevent duplicate packets being sent due to long delays in the network, the sequence number of the packet of data is also sent back with the ACK message.

Sliding window protocol improves TCP by letting the sender side transmit certain number of packets of data without waiting for ACK messages. The window size controls the number of packets of data to be sent before any ACK messages are received.

The success of TCP retransmission algorithm depends on the management of RTT [7].

As the number of computers inside the network grows, the problem of ACK messages and retransmissions becomes complicated, and there will be congestion in the network. Congestion is the focus of this research. The objective of this research is to apply the median filter to improve the performance of TCP to alleviate congestion.

IV. Median Filter

The following example demonstrates the operation of median filter [8].

Example Apply median filter with a window size of 3 to the array x(n), where x(n)=[2 80 6 3]. y[1]=Median[2 2 80]=2 y[2]=Median[2 80 6]=Median[2 6 80]=6 y[3]=Median[80 6 3]=Median[3 6 80]=6 y[4]=Median[6 3 3]=Median[3 3 6]=3 Therefore z(n)=[2 6 6 3] The original signal x(n)=[2 80 6 3], after being filtered by median filter, becomes z(n)=[2 6 6 3]

The properties of median filter are stated below [9], [10]:

1) The filter belongs to the class of Order Statistic (OS) filter.
2) Order Statistic filters operate according to the following steps:
   - The original signal x(n) is sorted in ascending order to produce y(n).
   - The median of y(n) is selected

   x(n) → Rank (sorting number in ascending order) → y(n) → Apply appropriate OS-filter-coefficient → z(n)

   x(n) = original signal
   y(n) = ranked signal
   z(n) = filtered signal
3) For median filter, the filter-coefficient matrix ,a, can be expressed as a = [0..0 1 0..0].
4) The filter has a non-linear frequency response due to its filter-coefficient matrix.
5) Noisy step signal is rounded at the step.
6) The filter will not attenuate noise well, if the noise comes from a noisy ramp signal
7) The filter eliminates impulse noise. However, if the impulse noise is close to an edge, the noise may be removed but the signal edge moves toward the impulse noise (edge jitter).
8) The filter preserves signal edges.

V. Median-filtering results from a wireless point-to-point connection

A simple point-to-point wireless connection is established between a notebook computer and a desktop computer. The distance between two computers is around one metre, and the transmission rate is 24Mbps. A network analyzer software is activated on the desktop computer to monitor and gather the network traffic. To simulate a wireless connection disturbance, the notebook computer is carried and moved around, while the desktop computer is streaming a video clip from the notebook computer.

The captured data from the network analyzer software is exported to an Excel file.

A MATLAB program is written to analyze a series of round-trip-time (RTT) values obtained experimentally from a wireless connection. The series of RTT will later be sent to a median filter for evaluating the performance of median filter on RTT. The median filter window size used for testing are three, five, seven and nine.

From Fig.2 to Fig.5, when the median filter window size increases, those RTT values which are greatly different from the rest are filtered away. However, the RTT values are getting smaller as the median filter window size increases.

From this observation, median filter can improve RTT, which in turn improves the congestion window size management. With consistently stable RTT values, a wireless connection can have fairly constant transfer rate.

VI. Results

According to RFC 793 [11], the RTT can be estimated by

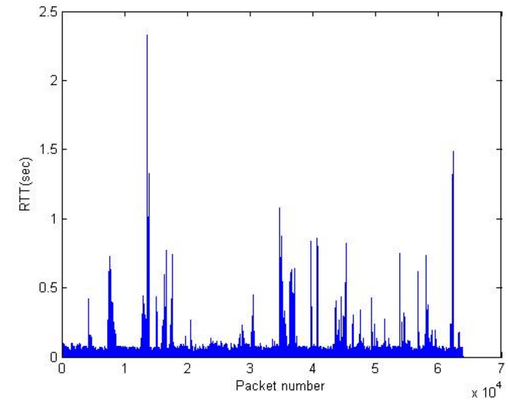

Fig. 1. Unfiltered RTT from a wireless connection.

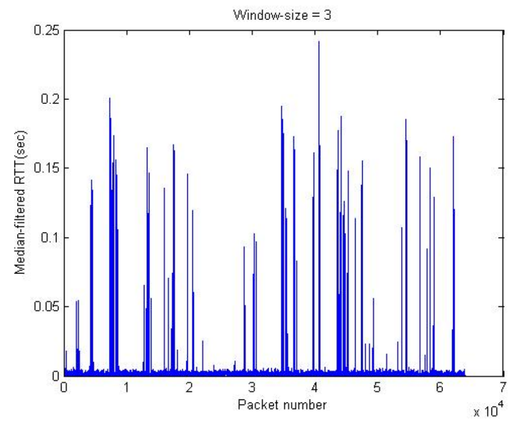

Fig. 2. Median-filtered RTT with window size equal to three.

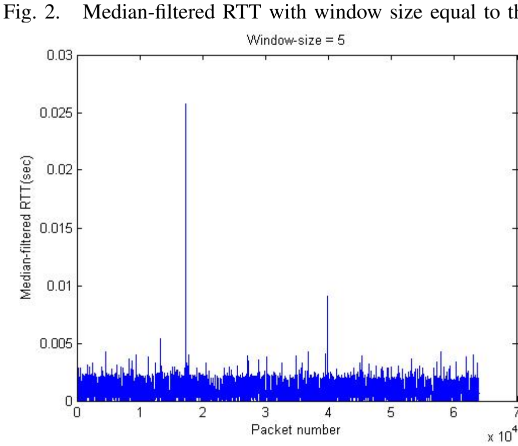

Fig. 3. Median-filtered RTT with window size equal to five.

$$SRTT = \alpha SRTT + (1 - \alpha)M \quad (1)$$

SRTT Smoothed RTT
α Smoothing factor
M Measured RTT

A MATLAB program is written to obtain a series of round-trip-time (RTT) by using (1).

If Fig. 6 is compared with Fig.5, the set of RTT estimated by median filter has less fluctuation than that estimated by weighted-average filter.

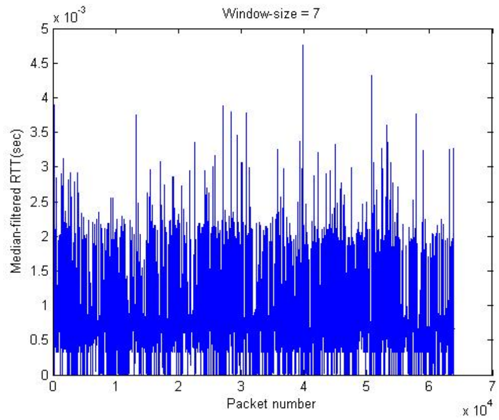

Fig. 4. Median-filtered RTT with window size equal to seven.

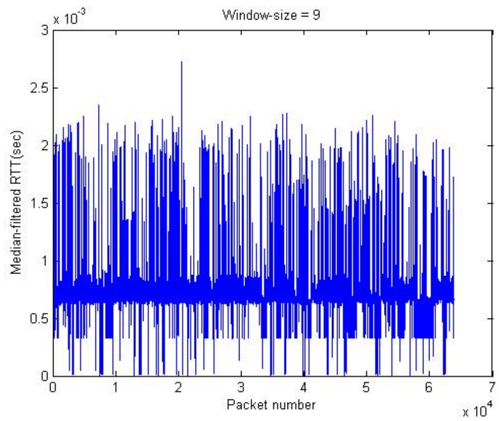

Fig. 5. Median-filtered RTT with window size equal to nine.

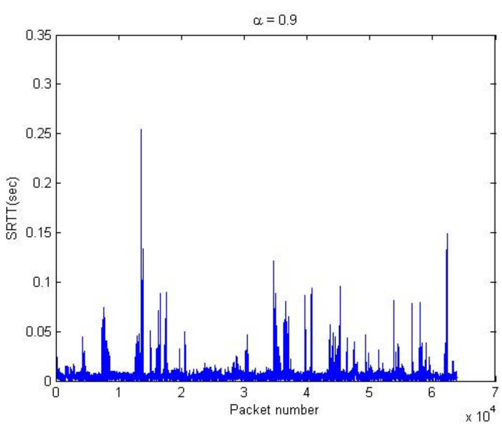

Fig. 6. Smoothed RTT by using (1)

A further investigation on the performance between median filter and weighted-average filter is carried out by using RTO.For simplicity, the unfiltered RTO and median-filtered RTO are taken to be 1.1 times of their respective RTT values. For the weighted-average filtered RTO, it is found by [12].

$$SRTO = SRTT + 4D \tag{2}$$

SRTO Smoothed RTO
SRTT Smoothed RTT
D Variance

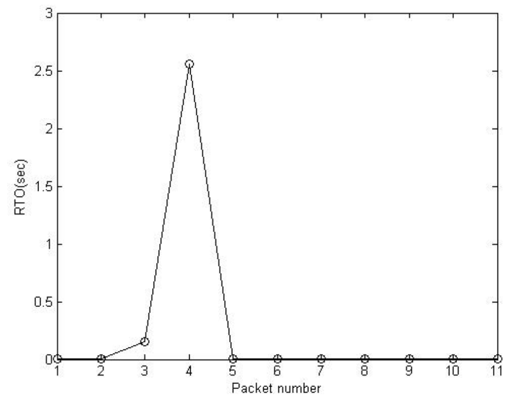

Fig. 7. Unfiltered RTO from a wireless connection.

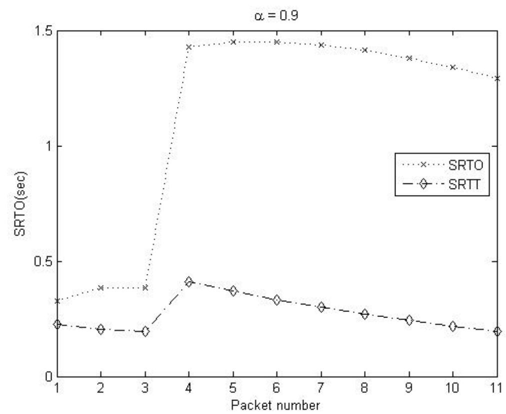

Fig. 8. Smoothed RTO using (2) and (3)

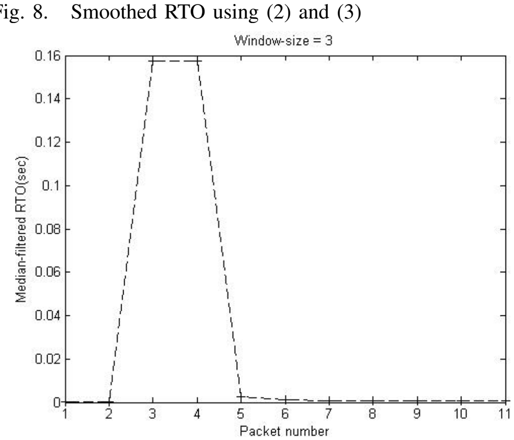

Fig. 9. Median-filtered RTO with window size equals to three.

$$D = \alpha D + (1 - \alpha)|SRTT - M| \tag{3}$$

D Variance
α Smoothing factor
SRTT Smoothed RTT
M Measured RTT

From the unfiltered RTT result, only one set of samples are chosen for the RTO comparison. Sample number 13655 to 13665 are chosen for testing. This set of samples represents a situation where the transmission rate is bursty. Results for

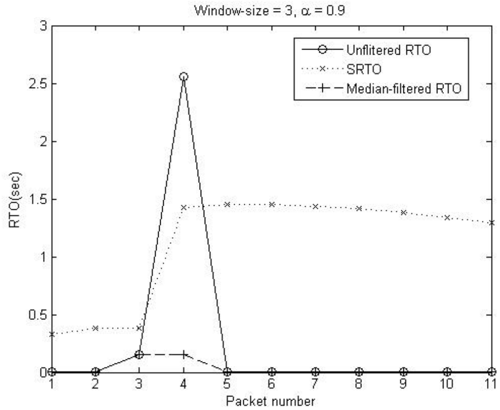

Fig. 10. Different RTOs.

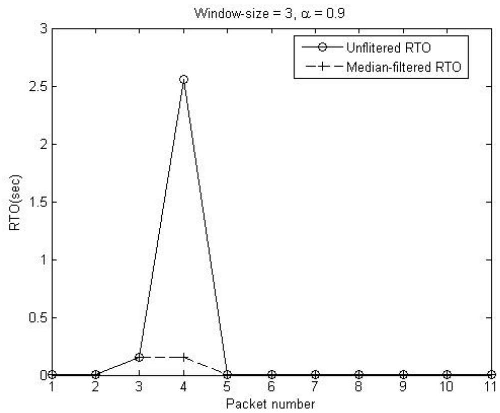

Fig. 11. Unfiltered RTO and median-filtered RTO.

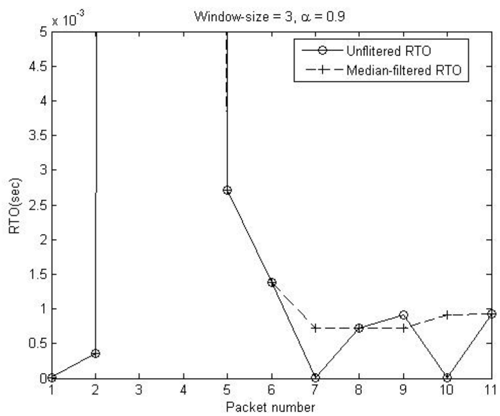

Fig. 12. Unfiltered RTO and median-filtered RTO (Zoom-in view).

sample number 13655 to 13665 are shown in Fig. 7 to Fig. 12.

From Fig. 10 to Fig. 12, weighted-average filter generates RTO timings much higher than those by median filter.

The histogram of RTT values are shown from Fig. 13 to Fig. 20. From Fig. 13, the histogram of the unfiltered RTT does not have a Gaussian distribution. That is, large RTT values exist on the right hand size of Fig. 13. However, these large RTT values are not in huge amount.

In Fig. 14, the histogram of SRTT shows a much gradual change in RTT values, compared with Fig. 13. However, the RTT values are not spread evenly.

TABLE I
EFFECT ON DIFFERENT WINDOW SIZES.

$Window\ Size$	$Mode(in\ millisecond)$	$Outlier = Mode \pm 10\%$
3	1.20	99.5%
5	0.64	51.3%
7	0.69	28.7%
9	0.67	30.8%
10	0.67	31.2%
11	0.67	26.5%

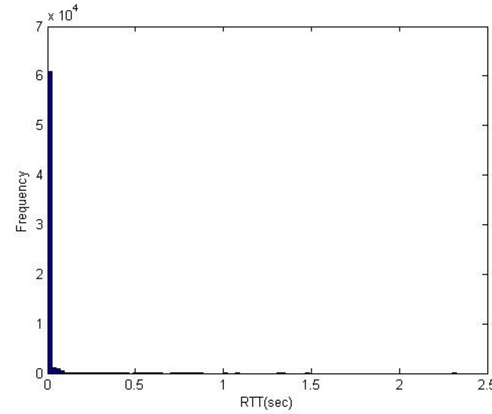

Fig. 13. Histogram of unfiltered RTT from a wireless connection.

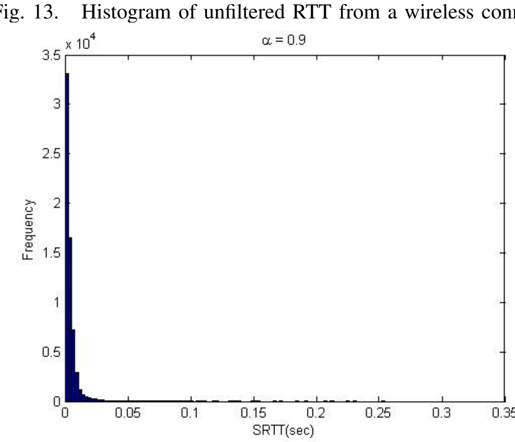

Fig. 14. Histogram of smoothed RTT by using (1)

From Fig. 15 to Fig. 20, pdf of median-filtered RTT with different window sizes are shown. From the window sizes, the pdf distribution can be significantly improved. That is, the large RTT values are being distributed to some smaller values.

In order to compare the results from different window sizes, a table is created. From table 1, mode and outlier from different window size are shown.

The mode is defined as the RTT value that occurs most in that set of filtered RTT values. Outlier is defined as the RTT value that is $\pm 10\%$ of the mode.

From table 1, the number of outliers decreases as the window size increases. This shows that exceptional large RTT values are successfully removed. However, the mode has no significant change after the window size is three.

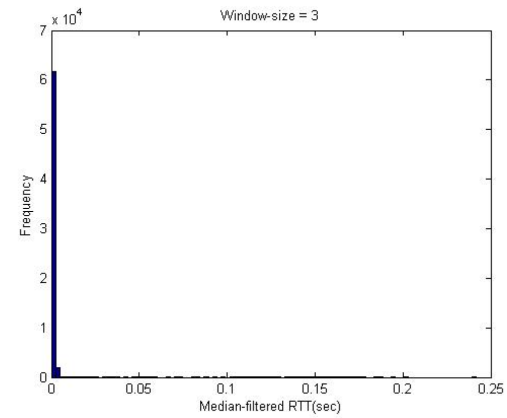

Fig. 15. Histogram of median-filtered RTT with window size equals to three.

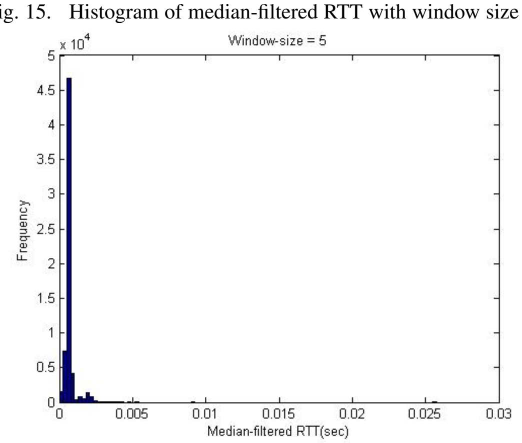

Fig. 16. Histogram of median-filtered RTT with window size equals to five.

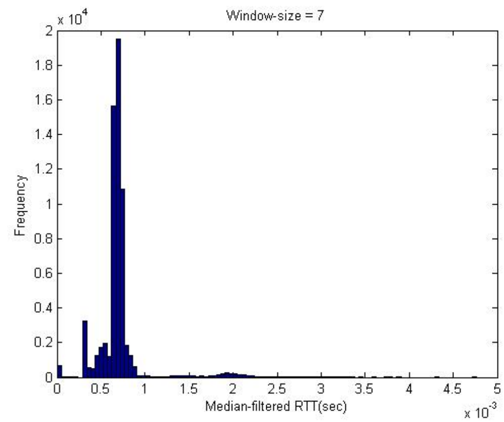

Fig. 17. Histogram of median-filtered RTT with window size equals to seven.

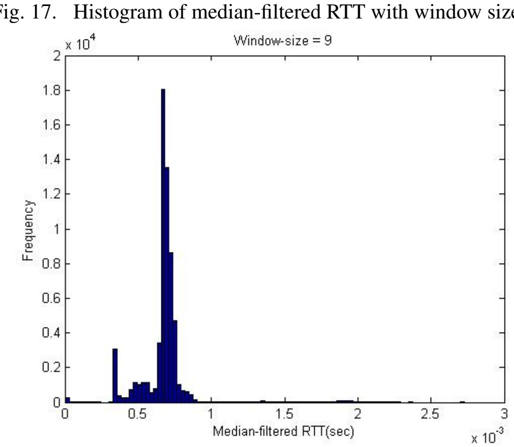

Fig. 18. Histogram of median-filtered RTT with window size equals to nine.

From these observations, median filter can perform better than the weighted-average filter because consistent RTT and small RTO are obtained, which are desirable factors for high connection throughput.

VII. Computation and implementation consideration

The experimental results presented in this paper indicates that median filter is useful for RTO estimation yet it is hard to be efficiently implemented for real-time usage. This problem is due to the fact that a median number can only be found from a series of numbers which are sorted in ascending or descending order. Sorting is a time consuming process and the time to sort a series of number is directly proportional to the window size of the median filter [13].

As the bit rate of wireless networks is increasing, the time required for the median computation becomes critical. From the previous section, it is shown that a large window size is desirable. However, this also increases the amount of time in finding the median.

From Fig. 21, the time required for median filtering from 63886 data samples is obtained by using an Intel®Core™2 E6750 with vPro™technology running at 2.66GHz. Two 800MHz-1GB RAMs are used. The software that is being used to rub median filtering is MATLAB 7 (R14). The sorting technique being used is bubble sort.

From [13], a fast two-dimensional median filtering algorithm was proposed, and the algorithm was proven to perform better than Quicksort [14]. The two-dimensional algorithm can be adapted for one-dimensional computation.

VIII. Conclusion

From the experimental results, median filter can perform better than the weighted-average filter because consistent RTT and small RTO are obtained, which are desirable factors for high connection throughput.

As the bit rate of wireless networks is increasing, the time required for the median computation becomes critical. From the previous section, it is shown that a large window size is desirable. However, this also increases the amount of time in finding the median. The time required for the median computation is examined for different window sizes, and it is found that the computation time is directly proportional to the window size.

IX. Further Development

The performance of median filter will be modeled by simulation using Network Simulator, version 2 (NS2). Median filter will also be implemented in Linux. Data from simulation and implementation will be analyzed. The fast two-dimensional median filtering algorithm will be adapted for one-dimensional median filtering to improve the computational efficiency.

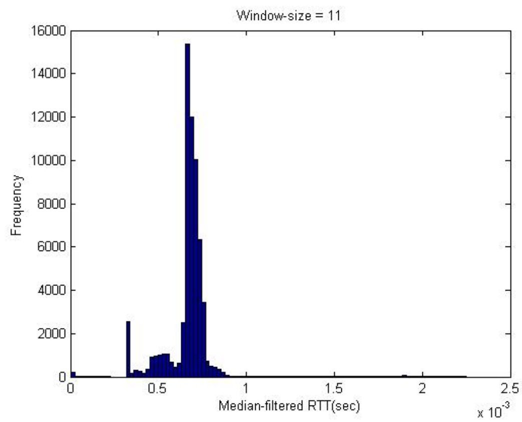

Fig. 19. Histogram of median-filtered RTT with window size equals to eleven.

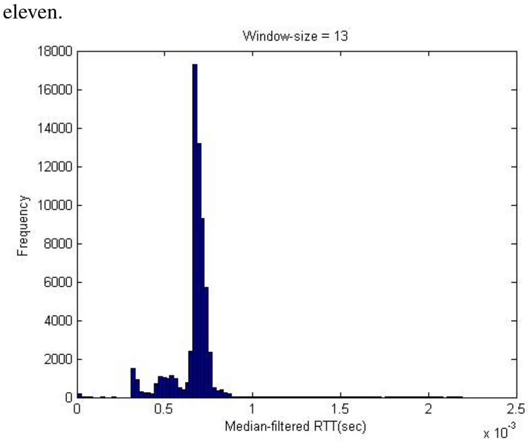

Fig. 20. Histogram of median-filtered RTT with window size equals to thirteen.

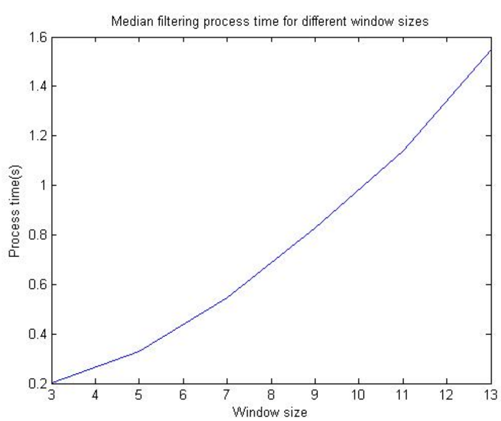

Fig. 21. Computational time against different window sizes.

References

[1] V. Jacobson, "Congestion avoidance and control," *ACM SIGCOMM Computer Communication Review*, vol. 18, no. 4, pp. 314 – 329, Aug 1988.

[2] P. Karn and C. Partridge, "Improving round-trip time estimates in reliable transport protocols," *ACM SIGCOMM Computer Communication Review*, vol. 17, no. 5, pp. 2 – 7, Oct 1987.

[3] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397 – 413, Aug 1993.

[4] University of California at Berkeley. The Berkeley Snoop Protocol. Viewed on 07 May 2008. [Online]. Available: http://nms.lcs.mit.edu/~hari/papers/snoop.html

[5] U. Black, *TCP/IP and Related Protocols*, 2nd ed. McGraw-Hill, New York, 1995, pp. 2–3.

[6] U. D. Black, *Data Communications and Distributed Networks*, 3rd ed. Prentice Hall, New Jersey, 1993, pp. 256–257.

[7] D. E. Comer, *Internetworking with TCP/IP Volume I*, 5th ed. Prentice Hall, New Jersey, 2006, pp. 204–206.

[8] Median filter. Wikipedia, the free encyclopedia. Viewed on 11 May 2008. [Online]. Available: http://en.wikipedia.org/wiki/median_filter

[9] P. M. Clarkson and H. Stark, *Signal processing Methods for Audio, Images and Telecommunications*. Academic Press Ltd., Great Britain, 1995.

[10] A. D. Poularikas, *The Handbook of Formulas and Tables for Signal Processing*. CRC Press, United States, 1999.

[11] RFC 793 (rfc793). Information Sciences Institute (ISI), University of Southern California. Viewed on 10 March 2008. [Online]. Available: http://www.faqs.org/rfcs/rfc793.html

[12] F. Halsall, *Computer Networking and the Internet*, 5th ed. Addision Wesley, United States, 2005, pp. 457–459.

[13] T. Huang, G. Yang, and G. Tang, "A fast two-dimensional median filtering algorithm," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 27, no. 1, pp. 13 – 18, Feb 1979.

[14] C. A. R. Hoare, "Quicksort," *The Computer Journal*, vol. 5, no. 1, pp. 10 – 15, 1962.

[15] A. Patil. (2002, 01 Jun) A Snoop for every node. IBM developerWorks. Viewed on 06 May 2008. [Online]. Available: http://www.ibm.com/developerworks/library/wi-snoop/?n-wi-6202

[16] S. G. Glisic, *Advanced Wireless Networks*. John Wiley and Sons Ltd., England, 2006, pp. 279–282.

[17] D. E. Comer, *Internetworking with TCP/IP Volume I*, 5th ed. Prentice Hall, New Jersey, 2006, p. 217.

[18] G. Arce and R. Stevenson, "On the synthesis of median filter systems," *IEEE Transactions on Circuits and Systems*, vol. 34, no. 4, pp. 420 – 429, Apr 1987.

[19] T. Nodes and N. Gallagher Jr., "Median filters: Some modifications and their properties," *IEEE Transactions on Acoustics, Speech, and Signal Processing [see also IEEE Transactions on Signal Processing]*, vol. 30, no. 5, pp. 739 – 746, Oct 1982.

[20] S. Perreault and P. Hebert, "Median filtering in constant time," *IEEE Transactions on Image Processing*, vol. 16, no. 9, pp. 2389 – 2394, Sep 2007.

[21] Q. Li and D. L. Mills, "Jitter-based delay-boundary prediction of wide-area networks," *IEEE/ACM Transactions on Networking*, vol. 9, no. 5, pp. 578 – 590, Oct 2001.

[22] M. Allman and V. Paxson, "On estimating end-to-end network path properties," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 2, pp. 124 – 151, Apr 2001.

[23] L. P. Ma, K. E. Barner, and G. R. Arce, "TCP retransmission timeout algorithm using weighted medians," *IEEE/ACM Transactions on Networking*, vol. 11, no. 6, pp. 569 – 572, Jun 2004.

[24] L. P. Ma, G. R. Arce, and K. E. Barner, "Statistical analysis of TCP's retransmission timeout algorithm," *IEEE/ACM Transactions on Networking*, vol. 14, no. 2, pp. 383 – 396, Apr 2006.

[25] D. Loguinov and H. Radha, "Retransmission schemes for streaming internet multimedia: evaluation model and performance analysis," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 2, pp. 70 – 83, Apr 2002.

[26] S. Mascolo, C. Casetti, M. Gerla, M. Y. Sanadidi, and R. Wang, "Tcp westwood: Bandwidth estimation for enhanced transport over wireless links," *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pp. 287 – 297, Jul 2001.

[27] DARPA Over the Years, August 1997. DARPA / ARPA –Defense / Advanced Research Project Agency. Viewed on 25 June 2008. [Online]. Available: http://www.livinginternet.com/i/ii_darpa.htm

[28] S. Vangala and M. A. Labrador. Performance of TCP over Wireless Networks with the Snoop Protocol. Department of Computer Science and Engineering, University of South Florida. Viewed on 25 June 2008. [Online]. Available: http://csdl2.computer.org/comp/proceedings/lcn/2002/1591/00/15910600.pdf