***I.J. Intelligent Systems and Applications*, 2018, 3, 59-66**
Published Online March 2018 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2018.03.07

New Delay-based Fast Retransmission Policy for CMT-SCTP

Lal Pratap Verma
Department of Computer Science and Engineering,
Jaypee University of Engineering and Technology, Guna, Madhya Pradesh, (INDIA)
E-mail: er.lpverma1986@gmail.com

Varun Kumar Sharma, Mahesh Kumar
Department of Computer Science and Engineering,
Jaypee University of Engineering and Technology, Guna, Madhya Pradesh, (INDIA)
E-mail: {varunksharma.102119.cse, mahesh.chahar}@gmail.com

Received: 20 April 2017; Accepted: 06 July 2017; Published: 08 March 2018

***Abstract*—**Concurrent Multipath Transfer (CMT) uses multi-homing feature of Stream Control Transmission Protocol (SCTP) to transfer data concurrently over the multiple paths. CMT provides bandwidth aggregation, fault tolerance, and reliability in multipath data transfer. In multipath data transmission, each path has different delay and bandwidth. Therefore, destination receives unordered data which causes receiver buffer blocking and unwanted congestion window (*cwnd*) reduction. Both the problem degrades the CMT performance significantly. Thus, this paper proposes a new delay-based fast retransmission policy to adjust the transmission rate of each path according to path delay. Simulation results show that the proposed approach achieves better throughput, reduces the number of the timeout and improves the *cwnd* growth. The proposed approach improved throughput up to 16% in variable packet loss and 18% in variable network delay environment.

***Index Terms*—**SCTP, CMT, Multipath, Multi-homing, Congestion window.

I. INTRODUCTION

A device is having more than one network interface, like Laptop or smartphones, is called multi-homed device. The multi-homing offers a pair of devices to establish the logical connection over the multiple interfaces [1]. The advantage of multi-homing devices is that they provide the backup path in case of network failure. The SCTP [1] is a transport layer protocol, provide message-oriented, full duplex, connection-oriented, and multi-homing services. SCTP offers elective reliability and ordering in a stream, multi-homing, multi-streaming and protection against SYN attacks. SCTP assumes one IP as a primary path while remaining treated as the secondary path. SCTP uses the transmission sequence number (TSN) to ensure the ordered data delivery. To take advantage of multi-homing, Iyengar et al. [2] proposed a CMT to transfer the data over the multiple paths concurrently. It provides bandwidth aggregation, robustness, and reliability in multipath data transfer. However, due to dissimilar path delay and bandwidth, multipath data transmission leads unordered data packet delivery at the receivers end. It causes unnecessary retransmissions, unwanted *cwnd* reduction, and receiver buffer blocking [3]. To minimize the receiver buffer blocking, Iyengar et al. [2] suggested the five retransmission path selection policies. However, these policies do not improve the buffer blocking in dissimilar bandwidth and delay network. To mitigate the unordered data chunk delivery, we suggested a new delay-based adaptive data chunk scheduling policy [24] to distribute data over the multipath according to the path delay and available bandwidth. The suggested policy improves the network utilization but still suffers from unwanted fast retransmission problem.

During fast retransmission [2], whenever CMT receives four duplicate SACKs, treats it as network congestion, reduces the *cwnd* and *ssthresh* to half of the current *cwnd*. However, destination also sent the SACKs when it receives unordered data. Therefore, the cause of unordered data delivery is dissimilar delay and bandwidth of each path. The maximum numbers of duplicate SACKs are generated due to unordered data delivery. Thus, the blind half reduction in *cwnd* is not appropriate for CMT and CMT-PF because it degrade the performance of CMT and CMT-PF significantly [3-4].

Therefore, this paper proposes a new delay-based fast retransmission policy to mitigate the receiver buffer blocking and *cwnd* growth problem. The new approach uses SRTT (Smooth Round Trip Time) as congestion window reduction factor. This factor reduces the *cwnd* in the small amount when RTT (round trip time) is small and reduces the *cwnd* in the significant amount when RTT is large.

The rest of the paper is organized as follows: Section 2 presents literature reviews of various CMT policies while section 3 presents a new delay-based fast retransmission technique for CMT. The performance evaluation of the proposed approach is presented in section 4 while section

5 concludes the overall performance of proposed method.

II. RELATED WORK

Iyengar et al. [2] identified the spurious retransmission problem of CMT and proposed a solution called Split Fast Retransmit (SFR) algorithm. It improves the performance of CMT concerning retransmission but suffers from unnecessary *cwnd* reduction when destination receives unordered data chunk due to dissimilar path delay and bandwidth. Iyengar et al. [2] suggested another algorithm, which maintains the separate congestion window for each destination to grow independently. It improves the *cwnd* growth but has the same problem of unnecessary *cwnd* reduction. The SCTP decrease the acknowledgment traffic by delaying acknowledgment until at least two can be sent collectively [2]. However, SCTP sends an immediate acknowledgment, when it receives unordered data chunk. Because of frequent unordered data chunk delivery, the reordering acknowledgment increase regularly. Delayed Ack for CMT (DAC) was included into SFR to minimize the acknowledgment traffic [2].

Ye et al. [4] proposed IPCC-SCTP to reduce the false retransmissions. It uses the unique path sequence number (PSN) for each path, which decides the ordered or unordered delivery of chunk for each destination. IPCC-SCTP improves the retransmission but suffers from buffer blocking problem. Dreibholz et al. [5] suggested a Sender Buffer Splitting approach which splits the sender buffer according to the number of paths. The author claims that proposed approach improved receiver buffer blocking but suffers from local blocking due to the dissimilarity of the path. Authors [3, 6] investigated the CMT and identified unnecessary fast retransmissions, crippled window growth, excessive network traffic; receive buffer blocking and naive scheduling problems.

Natarajan et al. [7] identified receiver buffer blocking problem due to path failure and suggested the solution (a new state for each destination) called Potentially-Failed (PF). This state indicates that the destination is not reachable due to congestion or link failure. Thus, all the new data transmitted over the available alternate path. It minimizes the packet loss due to link failure but suffers from receiver buffer blocking due to dissimilar bandwidth and delay of each path.

The CMT suffers from receiver buffer blocking due to dissimilar path delay and bandwidth causes unordered data chunk delivery. Yilmaz et al. [8] suggested a non-renegable selective acknowledgment (NR-SACKs) to free the receiver buffer. The NR-SACK simply removes the segment without bothering about reordering. Shailendra et al. [9] suggested an MPSCTP (Multipath SCTP) as a solution to unnecessary retransmission and window growth. The author claims better throughput and reduced retransmissions but suffers from buffer blocking problem. Shailendra et al. [10] proposed delay-based transmission adjustment policy to reduce the average packet delay of over the multiple paths. It minimized the buffer blocking problem but suffers from low bandwidth utilization. Shailendra et al. [11] suggested a Tx-CWND retransmission destination selection policy to improve the performance of MPSCTP in terms of receiver buffer blocking. Xu et al. [12] suggested a Quality-aware adaptive concurrent multipath data transfer in heterogeneous wireless networks (CMT-QA) to send data according to path quality. However, path quality estimation provides incorrect path quality value due to dissimilar path delay and bandwidth always have variable trends.

Authors also investigated soft computing based approaches [20-22, 29-30] to optimize network performance in wireless network. However, Thang and Tao [31] investigated the IPv6 routing protocol performance for Wireless Sensor Networks (WSN). Sharma and Kumar [23] suggested an adaptive congestion control scheme in mobile ad-hoc networks to improve the utilization of network.

MP-TCP [13] is another key connection-oriented protocol supports multi-homing. Likewise, SCTP does. MP-TCP works on the principal of distributing traffic over multiple paths. MP-TCP provides transparency in between top layer (application) to multiple connections. Moreover, MP-TCP works perfectly fine with the integrations of middle-boxes in today's Internet architecture [14-18]. MP-TCP offers better performance (comparing with conventional TCP) with data segments tearing middle-boxes in Internet's architecture. Consequently, MP-TCP offers better deployment capability with modern Internet architecture. In recent years, many of the un-coupled (independent congestion control between different sub-flows) strategies [25-26] were introduced. Nevertheless, the policy of controlling congestion independently (by sub-flows) leads to unfairness issue in the system. For this, MP-TCP introduces adaptive coupled congestion control policy by appropriately transforming congestion window growth policy concerning each sub- flow's network state [27-28].

Recently, various techniques have been presented to improve the MPTCP performance [14-18]. However, during the fast retransmission, all the suggested techniques reduce the *cwnd* to half of the current *cwnd* blindly which significantly degrade the performance of MPTCP.

III. PROPOSED WORK

In this section, we present a new delay-based fast retransmission policy to minimize the *cwnd* growth problem in multipath concurrent data transfer using CMT-SCTP. The multipath concurrent data transfer sends the data over the multiple paths while each path has different bandwidth and delay. Due to dissimilar path characteristics, data packet arrived out-of-order at the destination. When destination receives unordered data packet, it immediately sends gap information to the source. Four-time continuous reception of gap report concludes congestion on the path. Thus, source reduces the *cwnd* and *ssthresh* to half of current *cwnd*. However, network is not congested. Therefore, such reduction causes significant performance degradation while path is not congested.

When network congestion increases, RTT also increases, whereas unordered data chunk delivery may not increase the RTT. If, we include path delay as a factor of *cwnd* reduction, then it will control the reduction in *cwnd* and *ssthresh* appropriately instead of reducing to half blindly.

A. Effect of delay on cwnd reduction

The path delay plays a significant role in multipath data transfer because each path has different bandwidth and delay. Each path delay varies when path traffic intensity changes. If we reduce the *cwnd* according to path delay variation, then it may minimize the *cwnd* growth problem. The delay of the path is large if path is having high traffic intensity while delay is small when path has normal traffic intensity. If, reduction of the *cwnd* is made using the product of current path delay and *cwnd*, then it reduces the *cwnd* in a small amount when delay is small and reduces *cwnd* with large amount when delay is large.

Let, RTT and *cwnd* is the delay and congestion window of the path-1. According to proposed approach, *cwnd* can be reduced by the product of path delay and *cwnd*. Therefore, the formula of *cwnd* reduction when path-1 has normal traffic intensity is as:

$$cwnd_{i+1} = cwnd_i - (cwnd_i \times RTT_i) \tag{1}$$

Let, RTT of path-1 is RTT_1 when path is not congested then Eq. (1) is as

$$cwnd_{i+1} = cwnd_i - (cwnd_i \times RTT_1) \tag{2}$$

As congestion increases the RTT of the path also increases. Therefore, let the RTT of path-1 is RTT_2 when network is having high traffic intensity. Thus, the RTT_2 must be greater than RTT_1. Therefore, *cwnd* reduction to be done according to Eq. (3) is as:

$$cwnd_{i+1} = cwnd_i - (cwnd_i \times RTT_2) \tag{3}$$

If, $RTT_2 > RTT_1$.Then, the product of $cwnd_i$ and RTT_2 is also greater than the product of $cwnd_i$ and RTT_1. Hence, the reduction amount in *cwnd* is as

$$cwnd_i \times RTT_2 > cwnd_i \times RTT_1$$

This relation shows that when delay is large, reduction in *cwnd* is large while reduction is small when delay is small.

B. Path delay estimation

CMT uses round trip time (RTT) to estimate the delay of the each path. The estimation of RTT of each path includes queuing delay, transmission delay, processing delay, and propagation delay as:

$$RTT_{\min} = P_d + P_s + T_d + Q_{\min} \tag{4}$$

$$RTT_i = P_d + P_s + T_d + Q_i \tag{5}$$

where, RTT_i is a current RTT, RTT_{min} is a minimum RTT, P_d is propagation delay, Td is a transmission delay, Ps is the processing delay, Q_{min} is a minimum queuing delay, and Q_d is a current queuing delay of path. Our proposed method uses average path delay to reduce the error in RTT estimation. For average delay estimation, we use SRTT (smooth round trip time) can be estimated as:

$$SRTT_i = \begin{cases} RTT_i & first\ RTT \\ (I - RTO.Alpha) * SRTT_i + RTO.Alpha * RTT_i & current\ \ RTT \end{cases} \tag{6}$$

where, the recommended value of *RTO.Alpha* is 0.25 [1], RTT_{min} is the first RTT and RTT_i is the current RTT measured by source.

C. Congestion window reduction policy

Let $P_i = \{P_1, P_2, P_3 \ldots\ldots\ldots P_n\}$ be the paths used for multipath transmissions, and the round trip delay of each path is defined as $D_i = \{D_1, D_2, D_3 \ldots\ldots\ldots\ldots D_n\}$. If delay of i^{th} path changes, it means that traffic on the path also changes. If we include path delay as a factor of *cwnd* reduction, then it will control the reduction in *cwnd* and *ssthresh* appropriately instead of reducing to half blindly. Thus, the proposed method includes the current path's *cwnd* and SRTT as a *cwnd* reduction factor. This factor has been independently estimated for each path while receiving four duplicate SACKs. It reduces the *cwnd* of current path with large amount if congestion occurs while reduces with small amount in case of unordered data chunk delivery.

$$ssthresh_i = \max(cwnd_i - (cwnd_i \times SRTT_i), 4 \times MTU) \tag{7}$$

$$cwnd_i = ssthresh_i \tag{8}$$

where, MTU is the maximum transmission unit of SCTP. Eq. (7) and (8) show the formula for *cwnd* and *ssthresh* reduction. The algorithm of proposed fast retransmission policy is shown in Algorithm-1. The Algorithm-1 have two method, first is fast retransmission algorithm and second one is retransmission timeout (RTO) algorithm. Fast retransmission algorithm adjusts the transmission rate of path when source receives four duplicate SACKs. However, retransmission timeout algorithm adjusts the transmission rate of the path when retransmission timer expires. In multi-homing sinario, source calculates a separate RTO for each destination.

Algorithm-1: Fast retransmission and retransmission timeout algorithm

```
For every SACK received (at sender side for each destination):
Requirement: SRTT, MTU
Initialization: SRTT=RTT of current path, MTU=1500Byte
//Fast retransmission/recovery
If (four duplicate received)
    $ssthresh_i=max(cwnd_i-(cwnd_i*SRTT_i),4*MTU)$
    $cwnd_i=ssthresh_i$
End If
//Retransmission timeout
If (timeout occurred)
    $ssthresh_i=max(cwnd_i/2),4*MTU)$
    $cwnd_i=1*MTU$
End If
```

IV. PERFORMANCE EVALUATION

In this section, we compare the performance of proposed retransmission policy with well known CMT [2] fast retransmission policy. The whole simulation has been performed by using NS-2.35 [19]. Fig. 1 shows the network topology used for simulation. The topology has one SCTP source with two network interfaces S1&S2, and one SCTP destination with two network interfaces D1&D2. Initial bandwidth and delay of each link are shown in Fig. 1. The delay and bandwidth of each link may change according to simulation requirements. In this simulation setup, path-1 has fixed packet loss rate 1%, while path-2 has variable packet loss rate which varies from 1% to 10%. The SCTP source connected with FTP traffic generator and simulation time of this setup is 200sec.

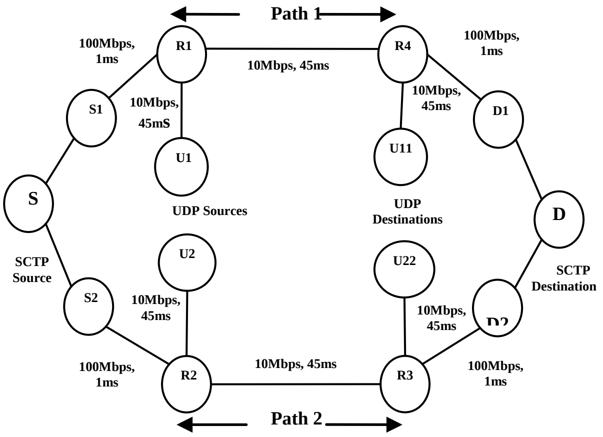


Fig.1. Simulation topology

The simulation topology also has two UDP sources U1, U2 and two UDP destinations U11, U22 respectively. The U1 and U11 are connected to router R1 and R4 while U2, U22 are connected to R2 and R3 respectively. This simulation setup is configured with drop tail queuing policy and default queue size is 50 packets. This simulation setup configured with recommended RTX-CWND retransmission path selection policy.

Fig. 2, 3 and 4 show the analysis of throughput, average throughput and retransmission timeout of proposed method, CMT and CMT-PF. In this simulation, receiver buffer size is 64KB; simulation time is 200 seconds and packet loss rate of path-2 varies from 1% to 10%. Rest of the network configuration is according to Fig. 1.

Fig. 2 demonstrates the throughput variation of CMT variants with variable packet loss rate. It shows that as packet loss rate increases the throughput of all CMT variants decreases. CMT and CMT-PF show the similar and linear throughput degradation because they use same cwnd and ssthresh reduction policy when congestion occurs, or unordered data chunk receives by destination. However, proposed method uses delay-based cwnd and ssthresh reduction policy which reduce the cwnd and ssthresh according to delay of the path. When path delay variation is large, it means that the traffic intensity is high, and if path delay variation is small, it means traffic is smooth. Thus, the proposed method use delay as factor of cwnd reduction which directly affect the cwnd reduction amount. Therefore, the proposed method shows the better cwnd growth and throughput for each packet loss rate.

Fig. 3 shows the average throughput of CMT variants with variable packet loss rate. It shows that CMT has least utilization as compared to CMT-PF and proposed method. However, the proposed method achieves improved throughput as compared to CMT and CMT-PF. The proposed method throughput improvement is 16% as compared to CMT and 15% as compared to CMT-PF.

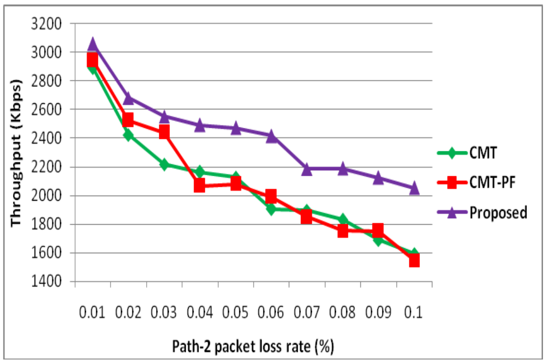


Fig.2. Packet loss rate Vs throughput

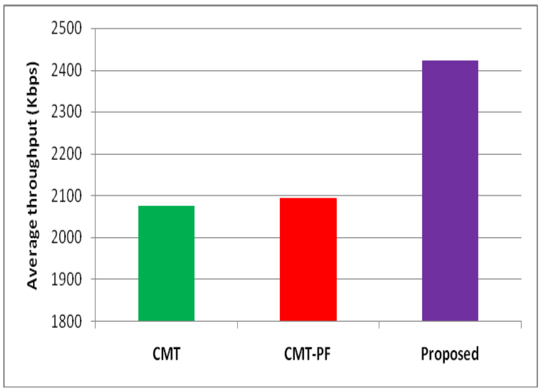


Fig.3. Average throughput of CMT variants

We also estimated the confidence interval for this simulation result. For 95% confidence level, the confidence interval of proposed method, CMT and CMT-PF are 2234.96-2611.63, 1837.27-2311.50 and 1830.40-2357.39 respectively. The confidence interval of all the CMT variants demonstrates that proposed method has better confidence interval as compared to CMT and CMT-PF.

Fig. 4 shows the average retransmission timeout of proposed method, CMT, and CMT-PF. It shows that CMT has highest number of timeout while proposed method shows the least number of timeout as compared to CMT and CMT-PF. It confirms that the delay-based *cwnd* and *ssthersh* reduction policy is a better approach as compared to halve the *cwnd* and *ssthresh* blindly. The proposed method average timeout improvement is 19% as compared to CMT and 6% as compared to CMT-PF. We also calculate the confidence interval for this simulation results. For 95% confidence level, the confidence interval of proposed method, CMT and CMT-PF are 7.74-16.45, 10.84-19.35 and 8.27-17.72. It is evident from confidence interval that proposed method has lower confidence interval concerning timeout as compared to CMT and CMT-PF.

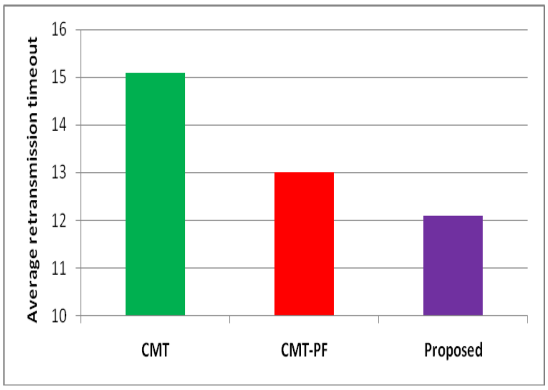
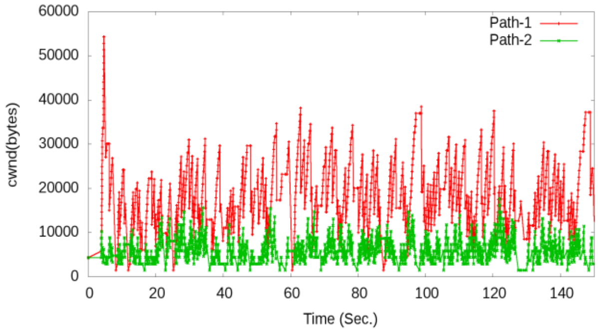
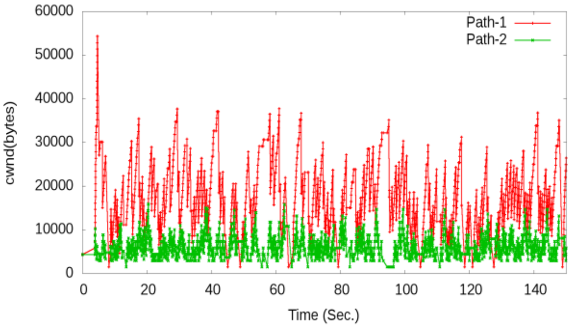


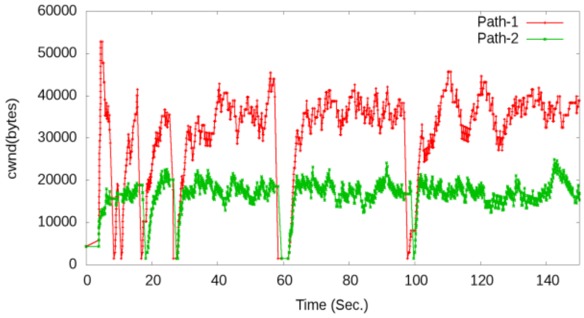
Fig.4. Average retransmission timeout of CMT variants



(a) CMT



(b) CMT-PF



(c) Proposed

Fig.5. Congestion window growths Vs time while packet loss rate of path-1 is 1% and path-2 is 10% (a) CMT (b) CMT-PF (c) Proposed

Fig. 5 (a)-(c) show the *cwnd* growth of proposed method, CMT, and CMT-PF while path-1 has 1% and path-2 has 10% packet loss rate. In this simulation setup, simulation time is 150 seconds and rests of the configuration parameters are same as given in Fig.1. This simulation study demonstrates the *cwnd* growth and reduction when packet loss or timeout occurs. The CMT and CMT-PF reduce the *cwnd* and *ssthresh* to half of current *cwnd* to adjust the transmission rate when source receives four duplicate SACKs. Therefore, CMT and CMT-PF suffers from *cwnd* growth problem. The proposed method uses the delay-based *cwnd* reduction approach to adjust the transmission rate. Therefore, proposed method reduces the *cwnd* is the small amount when the reason of *cwnd* reduction is unordered data chunk delivery. However, proposed method reduces the *cwnd* in the large amount when congestion occurs. Thus, the proposed method achieves better *cwnd* growth as compared to CMT and CMT-PF.

In next simulation, we analyze the effect of variable path delay on throughput and retransmission timeout. In this simulation setup, RTT of path-1 (100ms) remains constant while path-2 has variable RTT varies from 50-400ms. The packet loss rate of path-1 is 1% while path-2 has 5% packet loss rate. Rest of the simulation configuration remains same according to Fig. 1.

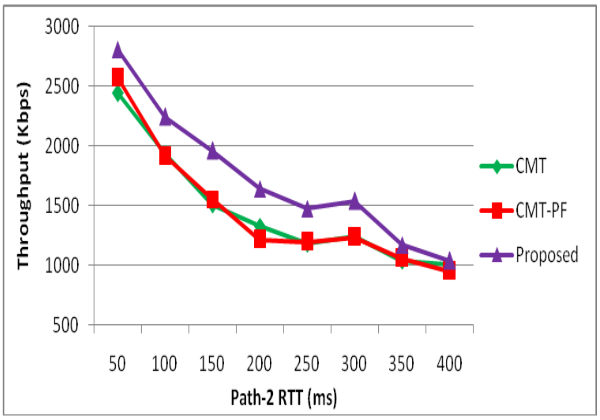


Fig.6. RTT Vs Throughput

Fig. 6 shows the throughput variation of CMT variants in variable RTT network environment. It demonstrates that as RTT increases, the throughput of the CMT variants decreases. The CMT and CMT-PF show the similar and linear trend in throughput drop. However, the proposed method demonstrates the higher throughput as compared to CMT and CMT-PF. The proposed method use path delay as *cwnd* reduction factor which reduces the *cwnd* according to path traffic conditions. On the other hand, CMT and CMT-PF reduce the *cwnd* to half of current *cwnd* blindly. As a result, proposed method achieves better throughput as compared to CMT and CMT-PF. The proposed method average throughput improvement is 18.83% as compared to CMT and 18.64% as compared to CMT-PF. Fig.7 shows the average retransmission timeout of CMT variants in variable RTT network. It demonstrates that the CMT-PF suffers from more timeout as compared to CMT. However, the proposed method has less timeout as compared to CMT and CMT-PF due to its delay-based *cwnd* reduction policy. The proposed method average timeout improvements are 23% and 34% as compared to CMT and CMT-PF respectively.

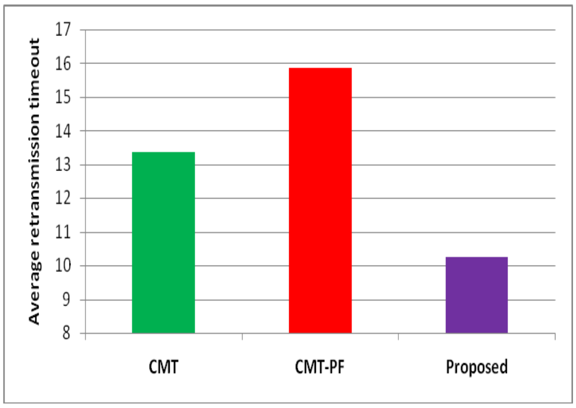
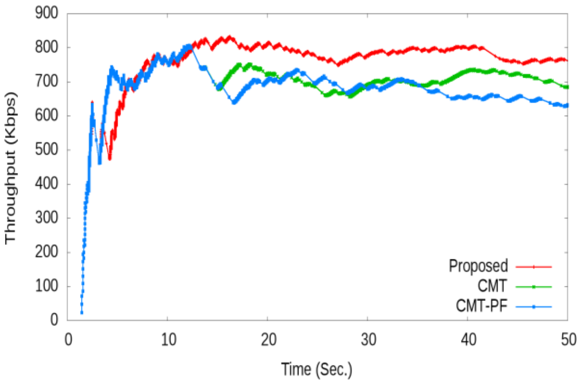
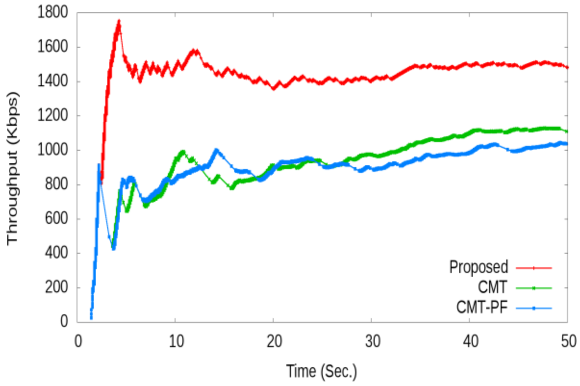


Fig.7. Average retransmission time of CMT variants

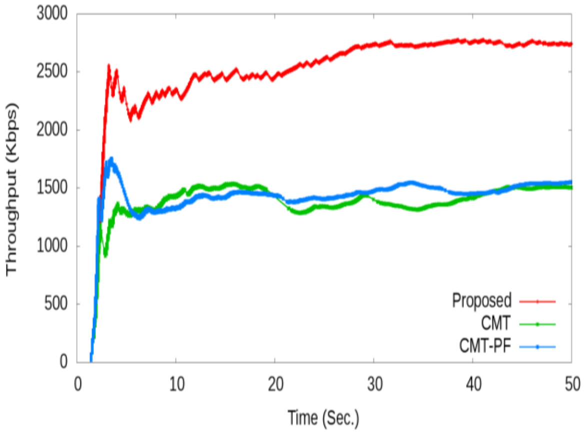
In another simulation, we analyze the effect of different receiver buffer (rbuf) on the performance of CMT variants. Fig.8(a)-(c) show the throughput of CMT, CMT-PF and proposed method with receiver buffer 32KB, 64KB, and 128KB. In this simulation setup, the packet loss rate of path-1 and path-2 are 1% and 5%. However, the propagation delay of path-1 and path-2 are 50ms and 150ms respectively. Rest of the simulation configurations remains same according to Fig.1.



(a) 32KB rbuf



(b) 64KB rbuf



(c) 128KB rbuf

Fig.8. Comparison of throughput using different receiver buffer sizes

It has been observed from Fig. 8(a)-(c), that the throughput of all CMT variants increases with the increase of receiver buffer size. At the start, the throughput of CMT variants increases rapidly because CMT variants probe the network capacity. After reaching network capacity, the throughput of CMT variants experiences variation due to packet loss detection (caused by congestion or unordered data delivery), then *cwnd* adjustment and fast retransmission. The proposed method differentiates the *cwnd* adjustment cause by either packet loss or unordered data delivery using delay-based *cwnd* adjustment policy. Therefore, the proposed method reduces the *cwnd* in a small amount when packet loss detected due to unordered data delivery while *cwnd* reduction is large when packet loss detected due to congestion. Such type of *cwnd* reduction improves the network utilization and reduces the timeout. As a result, the proposed method achieves better throughput as compared to CMT and CMT-PF for all receiver buffer size.

V. CONCLUSION

In this paper, we proposed a novel fast retransmission approach for CMT to adjust the *cwnd* and *ssthresh* based on path delay. The proposed approach uses the product of SRTT and *cwnd* as a *cwnd* reduction factor. This approach reduces the *cwnd* in the large amount when network is congested while it reduces the *cwnd* in the small amount when reduction caused by unordered data chunk delivery. The simulation results show that the proposed method achieves better throughput, reduces the retransmission timeout and has better *cwnd* growth as compared to CMT and CMT-PF. The proposed method average throughput improvement is 16% in variable packet loss rate and 18% in variable path delay environment.

REFERENCES

[1] R. Stewart, “Stream control transmission protocol”, RFC4960, *The Internet Engineering Task Force*, 2007. https://tools.ietf.org/html/rfc4960.

[2] J. Iyengar, P. Amer and R. Stewart “Concurrent Multipath Transfer Using SCTP Multihoming Over Independent End-to-End Paths,” *IEEE/ACM Transactions on Networking*, vol. 14, pp. 951–964, 2006.

[3] TD. Wallace and A. Shami “A Review of Multihoming Issues using Streaming Control Transmission Protocol,” *IEEE Communication Surveys & Tutorials*, vol. 14, pp 565-578, 2012.

[4] G. Ye, T. Saadawi and M. Lee “IPCC-SCTP: An enhancement to the standard SCTP to support multi-homing efficiently,” *IEEE International Conference on Performance, Computing, and Communications,* pp. 523–530, 2004.

[5] T. Dreibholz, M. Becke, J. Pulinthanath and EP. Rathgeb “On the use of concurrent multipath transfer over asymmetric paths,” *3rd IEEE Global Telecommunications Conference (GLOBECOM 2010)*, pp. 1-6, 2010.

[6] T. Dreibholz, EP. Rathgeb, I. Rungeler, R. Seggelmann,M. Tuxen and R. Stewartm “Stream Control Transmission Protocol: Past, Current and Future Standardization Activities,” *IEEE Communication Magazine*, vol. 49, pp. 82-88, 2011.

[7] P. Natarajan, N. Ekiz, P. Amer and R. Stewart “Concurrent multipath transfer during path failure,” *Computer communication,* vol. 32, pp. 1577–1587, 2009.

[8] E. Yilmaz, N. Ekiz, P. Natarajan, P. Amer, J. Leighton, F. Baker and R. Stewart “Throughput analysis of non-renegable selective acknowledgments (NR-SACKs) for SCTP,” *Computer Communication*, vol. 33, pp. 1982–1991, 2010.

[9] S. Shailendra, R. Bhattacharjee and SK. Bose “MPSCTP: A Simple and Efficient Multipath Algorithm for SCTP,” *IEEE Communication Letter*, vol. 15, pp.1139-1141, 2011.

[10] S. Shailendra, R. Bhattacharjee, SK. Bose “An implementation of Min–Max optimization for multipath SCTP through bandwidth estimation based resource pooling technique,” *International Journal of Electronics and Communication*, vol. 67, pp. 246-249, 2013.

[11] S. Shailendra, R. Bhattacharjee and SK. Bose “A multipath variant of SCTP with optimized flow division extension,” *Computer Communication*, vol. 67, pp. 56-65, 2015.

[12] C. Xu, T. Liu, J. Guan, H. Zhang and G. Muntean “CMT-QA: Quality-Aware Adaptive Concurrent Multipath Data Transfer in Heterogeneous Wireless Networks,” *IEEE Transaction on Mobile Computing*, vol. 12, pp. 2193-2205, 2013.

[13] A. Ford, C. Raiciu, M. Handley, S. Barre and J. Iyengar “Architectural guidelines for multipath TCP development,” *RFC6182, The Internet Engineering Task Force*, 2011. https://tools.ietf.org/html/rfc6182

[14] C. Paasch, R. Khalili and O. Bonaventure “On the benefits of applying experimental design to improve multipath TCP,” *In Proceedings of the Ninth ACM Conference on Emerging Networking Experiments and Technologies*, pp. 393-398, 2013.

[15] S. Barre, C. Paasch, and O. Bonaventure “Multipath TCP: from theory to practice,” *Networking 2011*, pp. 444–457, 2011.

[16] M. Li, A. Lukyanenko, S. Tarkoma, Y. Cu and A. Yla-Paaski “Tolerating path heterogeneity in multipath TCP with bounded receive buffers,” *In proceedings of the ACM SIGMETRICS/international conference on Measurement and modeling of computer systems*, pp. 375-376, 2014.

[17] D. Zhou, W. Song and M. Shi “Goodput improvement for multipath TCP by congestion window adaptation in multi-radio devices,” *In. Consumer Communications and Networking Conference (CCNC)*, pp. 508-514, 2013.

[18] N. Kuhn, E. Lochin, A. Mifdaoui, G. Sarwar, O. Mehani and R. Boreli “DAPS: Intelligent delay-aware packet scheduling for multipath transport,” *In: International Conference on Communications* (ICC), pp. 1222-1227, 2014.

[19] Home Page of Network Simulator-2 (NS-2), http://www.isi.edu/nsnam/ns, [Accessed 15 May 2017]

[20] M. Nirmala and R. V. Pujeri “Performance of TCP Vegas, Bic and Reno Congestion Control Algorithms on Iridium Satellite Constellations” *International Journal of Intelligent Systems and Applications (IJISA)*, vol. 12, pp 40-49, 2012.

[21] S. Sharma, S. Kumar, and B. Singh,"Hybrid Intelligent Routing in Wireless Mesh Networks: Soft Computing Based Approaches", *International Journal of Intelligent Systems and Applications (IJISA)*, vol.6, no.1, pp.45-57, 2014.

[22] V. K. Sharma, S. S. P. Shukla, and V. Singh, “A tailored Q-Learning for routing in wireless sensor networks,” *In: 2nd IEEE International Conference on Parallel, Distributed and Grid Computing*, pp. 663-668, Dec. 2012.

[23] V. K. Sharma, and M. Kumar, “Adaptive congestion control scheme in mobile ad-hoc networks,” *Peer-to-Peer Networking and Applications*, vol. 10, no. 3, pp. 633-657, May 2017.

[24] L. P. Verma, and M. Kumar, “An adaptive data chunk scheduling for concurrent multipath transfer,” *Computer Standards & Interfaces*, vol. 52, pp. 97-104, May 2017.

[25] H.-Y. Hsie and R. Sivakumar, “pTCP: an end-to-end transport layer protocol for striped connections,” *In: Proceedings of the 10th IEEE Conference on Network*

Protocols (ICNP), pp. 24-33, 2002.
[26] C.M. Huang and C.H. Tsai, "WiMP-SCTP: Multi-path transmission using stream control transmission protocol (SCTP) in wireless networks," *In: Proceedings of the 21st IEEE Conference on Advanced Information Networking and Applications Workshops (AINAW)*, pp. 209-214, 2007.
[27] C. Raiciu, S. Barre, C. Pluntke, A. Greenhalgh, D. Wischik, and M. Handley, "Improving datacenter performance and robustness with multipath TCP", *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4, pp. 266–277, 2011.
[28] C. Raiciu, M. Handly and D. Wischik, "Coupled congestion control for multipath transport protocols", RFC 6356, *Internet Engineering Task Force,* 2011. https://tools.ietf.org/html/rfc6356 .
[29] M. A. Mansor, M. S. M. Kasihmuddin and S. Sathasivam, "Enhanced Hopfield Network for Pattern Satisfiability Optimization", *International Journal of Intelligent Systems and Applications (IJISA)*, vol.8, no.11, pp.27-33, 2016.
[30] R. Janapati, C. Balaswamy, K. Soundararajan, "Enhancement of Indoor Localization in WSN using PSO tuned EKF", *International Journal of Intelligent Systems and Applications (IJISA)*, vol.9, no.2, pp.10-17, 2017.
[31] V. C. Thang and N. V. Tao, "A Performance Evaluation of Improved IPv6 Routing Protocol for Wireless Sensor Networks", *International Journal of Intelligent Systems and Applications (IJISA)*, vol.8, no.12, pp.18-25, 2016

Authors' Profiles

Lal Pratap Verma received his B.Tech. degree in Computer Science and Engineering from Dr. R.M.L. Avadh University, Faizabad, India in 2007 and M.Tech. degree in Information Technology from YMCA University of Science and Technology, Faridabad, India in 2011. He is currently working towards his Ph.D. degree in Computer Science and Engineering at Jaypee University of Engineering and Technology, Guna, India. His research interests include network systems, network security and Mobile Ad-hoc Network.

Varun Kumar Sharma received the Masters of Technology in Computer Science and Engineering from Jaypee University of Engineering and Technology, Guna, India and Bachelors of Engineering (Honors) in Computer Science and Engineering from Rajiv Gandhi Technical University, Bhopal, India. He is currently pursuing Ph.D. in Computer Science and Engineering from Jaypee University of Engineering and Technology, Guna, India. His research interests include performance evaluation of wireless network routing, cross-layer optimizations for ad-hoc and sensor networks, and energy aware routing.

Mahesh Kumar is working as Assistant Professor in the Department of Computer Science and Engineering at Jaypee University of Engineering and Technology, Guna, Madhya Pradesh, India. He earned his Ph. D. in Computer Science and Engineering, and M.Tech. in Information Technology. He has more than 11 years of experience in teaching and research. His current research interests include Wireless and Wired Networks, Internet Routing, Network Security, and IPv6.

How to cite this paper: Lal Pratap Verma, Varun Kumar Sharma, Mahesh Kumar "New Delay-based Fast Retransmission Policy for CMT-SCTP", International Journal of Intelligent Systems and Applications(IJISA), Vol.10, No.3, pp.59-66, 2018. DOI: 10.5815/ijisa.2018.03.07