A Rate Control Algorithm to Improve TCP over RFID Reader Network

R. Radha, Amit Kumar Tyagi and K. Kathiravan

Abstract In many emerging RFID applications such as location tracking, multiple readers are involved in collecting tag information and sending it to a computer. A large number of computers and readers are augmented into the network for efficient data processing and local decision-making. This type of ad hoc RFID reader network uses TCP as a transport layer protocol. But TCP cannot perform well in any wireless network due to its burst nature of data transmission. This work proposes a rate-based transmission algorithm which is implemented as a layer between TCP and network layer. The algorithm proposed in this work ensures that packets are sent one after another with less delay (i.e., between them). Here, exponential average of an end-to-end delay is used as a metric in determining the delay between the packets. This delay reflects the congestion status in the network and avoids contention between successive data packets. The evaluation of the performance of our algorithm against TCP New Reno using NS 2.35 simulator shows significant performance improvement of throughput, the end-to-end delay, the link layer contentions, and the route failure.

Keywords Location tracking · Rate-based transmission · RFID · TCP

1 Introduction

Radio Frequency Identification (RFID) is a technology for automatic data capture using electromagnetic transmission. It consists of two components: RFID readers and tags [1]. The reader can capture the IDs emitted by RFID tags. They use a

---

R. Radha (✉)
Department of Computer Science and Engineering, K L University, Hyderabad, Telangana, India
e-mail: radhupriya@yahoo.com

A. K. Tyagi
School of Computer Science and Engineering, Lingayas Vidyapeeth, Faridabad 121002, Haryana, India

K. Kathiravan
Department of Information Technology, Easwari Engineering College, Chennai, Tamilnadu, India

S. Choudhury et al. (eds.), *Intelligent Communication, Control and Devices*,
Advances in Intelligent Systems and Computing 989,
https://doi.org/10.1007/978-981-13-8618-3_3

specific frequency and protocol to transmit and receive data. RFID tags are of two types based on the fact whether they are powered by battery. Passive tags do not use battery power. They transmit IDs to the reader by backscattering the part of RF power emitted by reader [1, 2]. Active tags use battery power for transmission of data. RFID application includes factory automation, supply chain, production, smart home environment, health care, parking management, traffic control system, etc. Many applications need to track the physical location of multiple moving objects and make a local decision. The transmission range of tag to reader covers up to a few meters. Many organizations distribute multiple readers over space and connect them to a computer. They augment a number of computers and readers to cover their area of interest. Each computer is involved in huge data processing to eliminate redundancy and communicate with others for optimal local decision-making [3]. Note that this type of network can be formed in an ad hoc manner without the need for any infrastructure.

Basically, TCP (Transmission Control Protocol) is used as a transport layer protocol. It was designed for wired networks. It cannot perform well in wireless network due to its burst data transmission. It misinterprets any type of packet loss that occurs as congestion and reduces the rate of sending drastically. This work proposes a rate-based transmission algorithm, i.e., inserted as a small layer between TCP and network layer (for more details, see Sect. 3). It accepts data packets from TCP but does not send it out immediately. In our proposed approach, packets are delivered out one after the other with sufficient delay between them. An end-to-end delay is used as a metric for calculating this interval. End-to-end delay is measured as the time taken for a packet to reach the destination from source and vice versa. Sufficient interval between the packets helps to avoid contention between successive data packets, and it also reflects the congestion status in the network. This kind of periodic transmission is suitable to the RFID network because reader to tag (passive) communication also happens at a regular interval. Note that we do not make any modifications in the TCP source and destination and add a thin layer at the bottom of TCP and we do not rely on any cross-layer feedback. This algorithm can also be used to improve TCP performance for any application over multi-hop in mobile ad hoc networks.

Hence, remaining of this work is organized as follows: The explanation of the related work is presented in Sect. 2. In Sect. 3, the study of impact of the TCP on the link layer contentions and the system model and protocol architecture are presented in Sect. 4. Further, we discuss the implementation results in Sect. 5. Finally, this work is concluded by providing future work (in brief) in Sect. 6.

2 Related Work

This section discusses various research articles (published in the past decade by several researchers), in which the authors try to control the traffic and reduce the collision over multi-hop Mobile Ad hoc Network (MANET). Fu et al. [4] discussed the reason for having high packet losses, and they found that high competition among the nodes is the main reason for high packet drops (not buffer overflow, note that primary factor is contributing for maximum packet losses). Also in [4], authors maintain valid congestion Window based on the probability of link layer losses. In [5], Xin proposed an approach with cross-layer design that decrements congestion window based on the network's capability. Round trip delay is measured with queuing time gathered from the nodes present on the way. Further, in [6], Zhang proposed a rate-based transmission that controls congestion window by the measures of medium utilization and rate of contention. Further, in [7], ElRakabawy et al. proposed a method of rate control with out of interference period and variable rate RTT. Later in [8], Sundaresan discussed a method of rate-controlled transmission by using queuing delay. Further, in [9], Ehsan et al. proposed a control (contention) approach where TCP receiver tracks the contention delay and achieves goodput. In [9], the traffic rate is also computed and then updated back to the source to look after the transmission rate. Further, in [10], Jubari et al. determined and controlled the collision between data and acknowledgment packets.

Also in [10], the authors track the packet drop events by tracking the inter-arrival rate among acknowledgments and control the rate of transmission among the data packets. In [11], Jiwei Chen et al. proposed an adaptive ACK scheme (delayed) for ad hoc and hybrid networks. In [12], Tang Lun et al. discussed a Beaconing rate control approach (1-D Markov model). In [12], a beacon rate control scheme mitigates the congestion and maximizes the beacon's delivery efficiency. Further, in [13], Kaixiong Zhou et al. proposed a distributed channel allocation (also a rate control approach) to solve the cross-layer design problem. Further, in [14], Pham Thanh Giang and Kenji Nakagawa also established cooperation among TCP flows with channel access and rate adaptation using cross-layer design. Farzaneh et al. [15] proposed a dynamic TCP-MAC interaction approach to reduce the total number of induced ACKs based on the current channel condition. In last, Carlos De M et al. [16] suggest several mechanisms to enhance TCP performance such as for transmitting forward data and reverse data ACKs to minimize the capture likelihood.

Hence, this section discusses existing work related to our problem. Now, the next section will discuss the effect of TCP on link layer protocol in brief.

3 Effect of TCP on Link Layer Protocol

Transport Control Protocol (TCP) is a transport layer protocol which is designed for wired network, where the losses of packets occur due to congestion. TCP assumes any type of packet loss as congestion and reduces the rate of sending. TCP does not perform well in wireless network due to the inappropriate congestion control algorithm. TCP is burst in nature. It may deliver a group of two or three packets together without worrying about the time interval among them. It is clocked by ACKnowledgment (ACK).

Whenever it receives an acknowledgment, it executes congestion control algorithm and pumps more data into the network, which leads to severe contention at

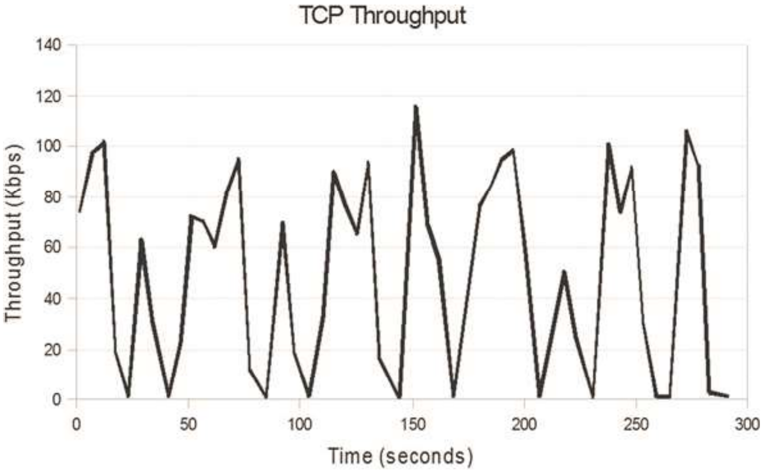


Fig. 1 Variation in TCP throughput

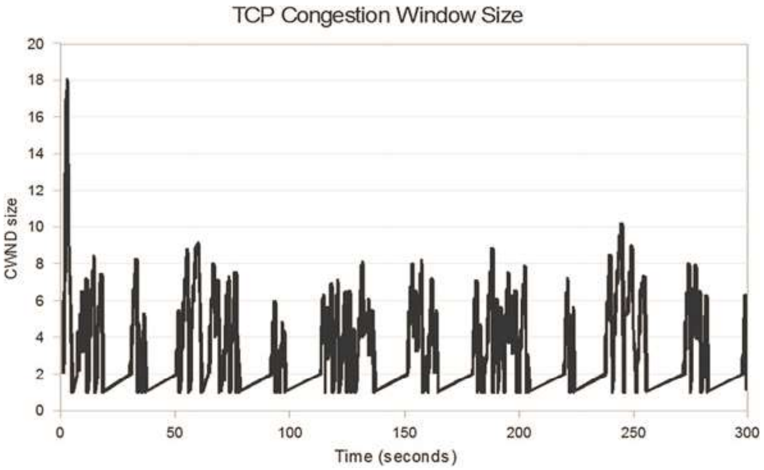


Fig. 2 Variation in TCP congestion window size

MAC layer. The contention results in co-channel interference [8]. This causes collision losses and thereby misinterpreted route failures. It increases the overhead and reduces the throughput finally. Figures 1 and 2 show the variation in TCP throughput and congestion window size for a static horizontal chain of eight nodes, which is shown in Fig. 3 with seven hop TCP connection. Hence, there is a drastic reduction in throughput at many points due to the intra-flow interference. The congestion window is reset to slow start phase at many points.

The contention which occurs at MAC layer can be divided into two types: intra-flow contention and inter-flow contention. The intra-flow contention occurs between

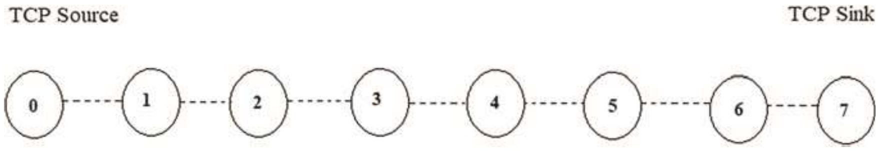


Fig. 3 Horizontal static network

the data packets or between data and acknowledgment packets of the same TCP connection to access the wireless medium whereas the inter-flow contention takes places between the packets of neighboring or parallel TCP connections to access the medium.

The intra-flow contention can be controlled using rate controlled transmission at TCP source. The packets can be sent one after the other with sufficient time intervals. The interval of time between the deliveries of successive packets should be carefully selected such that it does not cause contention and also it should not lead to increased delay which causes unnecessary timeouts for the TCP packets. We explain our rate control algorithm in Sect. 4.

4 System Model

This section discusses the system architecture in Fig. 4. The reader periodically emits the electromagnetic waves and collects tag information which is sent to a computer. Since each computer is connected with multiple readers, it generally receives duplicate tag information. It is the responsibility of the computer to eliminate duplications and communicate with other computers to facilitate local decision-making. The protocol architecture is explained in Fig. 5. Our protocol is implemented as a layer between TCP and network layer. It includes three functions: (a) data processing, (b) rate control, and (c) ACK processing.

Our proposed model can be discussed as follows: It accepts data from TCP layers but delivers them one by one at a specific time interval. This time interval is calculated dynamically by the rate control algorithm. The time interval should be carefully selected, it should be large enough which may otherwise lead to contention, and it should be small which may otherwise cause unnecessary timeouts for the queued TCP packets and reduce the overall throughput. We have selected exponential average of end-to-end delays as the values for calculating the inter-packet delivery period. The end-to-end delay in the forward and reverse path reflects the congestion in the network. Note that the end-to-end delay is smaller for less congested network and high for more congested network. We also prevent contention between the successive packets by setting the inter-packet delivery period approximately to the end delay. We assume that exponential average of end-to-end delay of recent n packets will reflect the end-to-end delay for the new packet to be transmitted. We find end-to-end

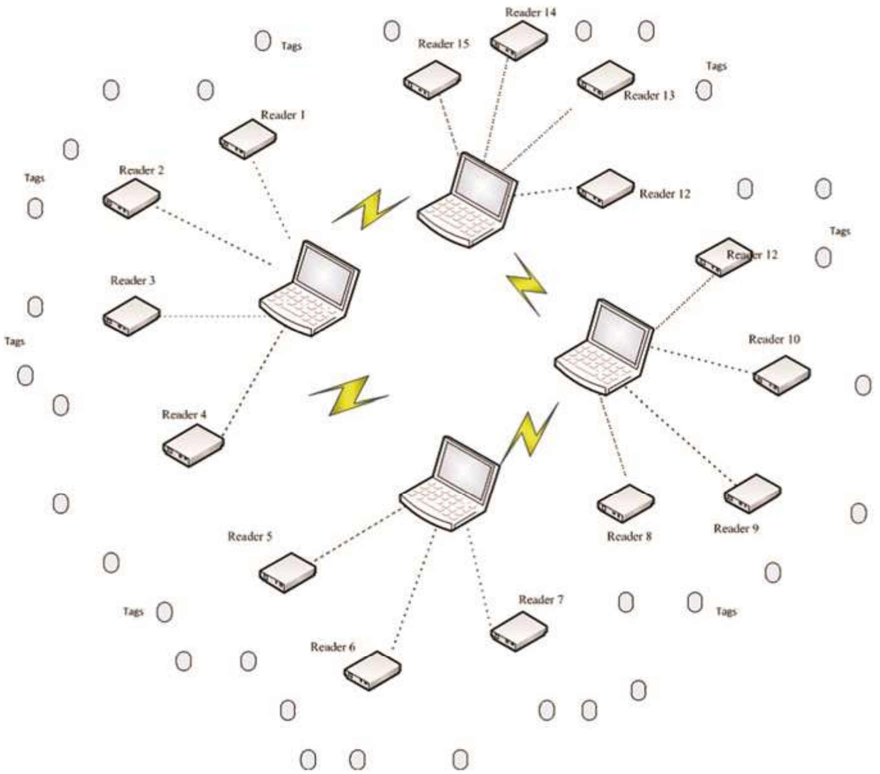


Fig. 4 System architecture

delay for every TCP (forward delay) and corresponding TCP ACK (reverse delay) packet. Here, forward delays are calculated as the interval between the time in which the packet is sent from the thin layer and the time when the packet reaches the destination TCP. And reverse delay is calculated in the reverse manner. The average of forward and reverse delay gives the delay for the recent packet. When the layer receives TCP ACK, it extracts the time stamps stored by the TCP sink and forwards it to the TCP layer immediately. The rate control algorithm updates the inter-packet delivery period using Fig. 5 process.

Here, α is the weightage factor which is given the value of 0.4 in our case, Ti1 is the time at which TCP Data (i) was sent out by source from the thin layer, and Ti2 is the time at which TCP Data (i) reaches TCP destination. Ti4 is the time at which TCP ACK reaches the thin layer in the source. Ti3 is the time at which TCP ACK was sent out from destination. These time stamps are extracted from the received acknowledgment. Eif is the forward delay, Eir is the reverse delay, and Delay (i) is the average of forward and reverse delay of *i*th transmission. Exp Delay (i + 1) is the exponential mean of recent i transmissions which will be updated as current

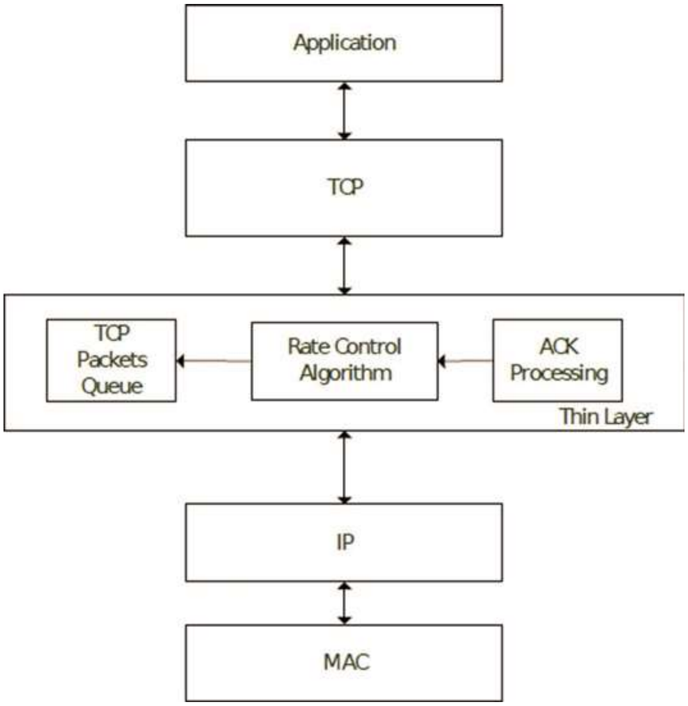


Fig. 5 Proposed protocol architecture

Inter-packet Delivery Period (IPD). This IPD will be updated during the arrival of every acknowledgment.

Whenever the duplicate acknowledgments are received, it does not execute the rate control algorithm, but they are immediately forwarded to the TCP layer. We have used a timer called transmit timer which is set for the duration of period of inter-packet delivery. On the expiry of this, a new packet is sent out.

5 Simulation Results

The performance of our solution is evaluated using rate TCP with New Reno using ns2 simulator. We focus a horizontal setup of eight nodes over static multi-hop ad hoc networks. Node 0 as the sender and Node 7 as the receiver. The configuration/parameters of our topology are listed in Table 1.

Table 1 Simulation parameters

Simulation time	300 s
Starting time	At 1.0 s
Number of nodes	8
Source, destination node	0, 7
Routing and MAC protocol	AODV and IEEE 802.11
Transport protocol	Rate TCP/TCP New Reno
Transmission limit	250 m
Interference limit	550 m
Distance among nodes	200 m

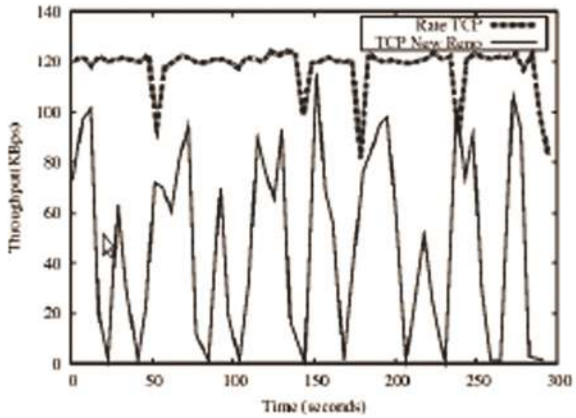


Fig. 6 Variation in TCP throughput

5.1 *Throughput*

In Fig. 6, Rate TCP is evaluated against TCP New Reno in terms of throughput which is defined as the amount of bytes received by destination per second. The throughput of TCP New Reno is highly variable throughout the simulation. It is due to the timeouts and mis interpreted route failures. Rate TCP performs better than TCP New Reno continuously, and it comes down in few positions because of the collision to the collisions between data and acknowledgments that leads to mis leaded route error followed by throughput reduction.

5.2 *End-to-End Delay*

Figure 7 shows the path delay for every packet emitted from the source. End-to-end delay is defined as the time taken for the packet to reach the destination after traveling through multiple hops. As packet generation is high for rate TCP, the plot is more

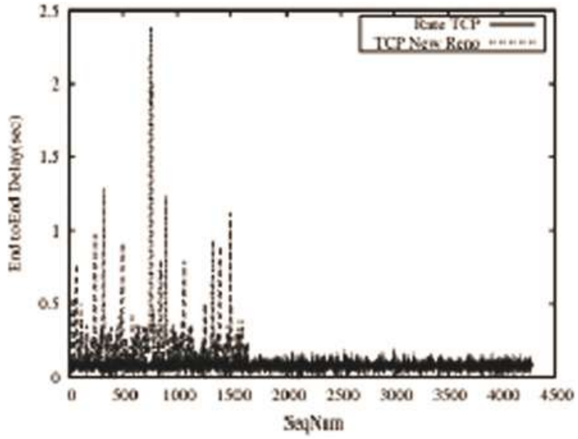


Fig. 7 End-to-end delay

for rate TCP. End-to-end delay is constantly less for Rate TCP than TCP. It shows that the local hop delay is also less for rate TCP.

5.3 *Route Failures and Timeouts*

Figure 8 shows the performance of rate TCP in a static topology free from mobility. Rate TCP has much less number of mistaken route errors, time exceeds, and retransmissions.

There is no mobility-related losses and note but route failure occurs due to the interference among the packets. Repeated route errors lead to much overhead and scarcity of bandwidth.

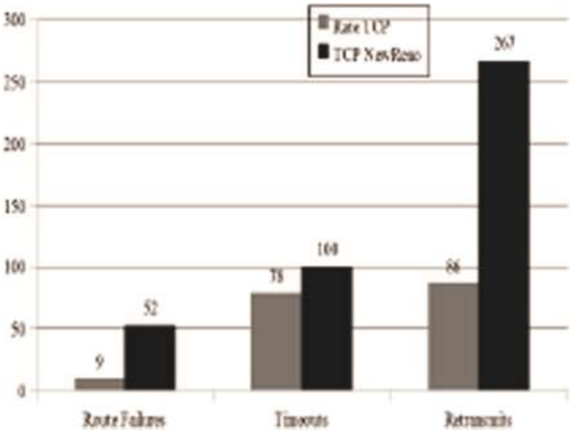
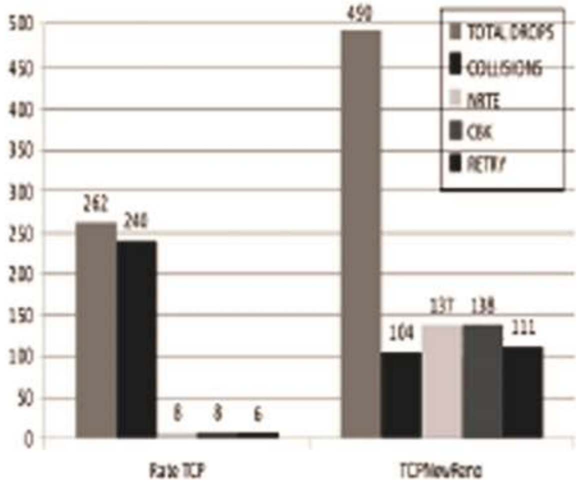


Fig. 8 Route failures, restransmits, and timeouts

Fig. 9 Packet drops



5.4 *Packet Drops*

Figure 9 describes all types of packet drops. The possible packet drops that may take place are interference losses, NRTE means No Route which is a loss due to unavailability of RTENTRY in routing table to transmit the packet. RETRY is the loss in MAC layer after the maximum attempt of retransmissions. Collision is high while comparing other losses. But the number of drops is very less than NewReno.

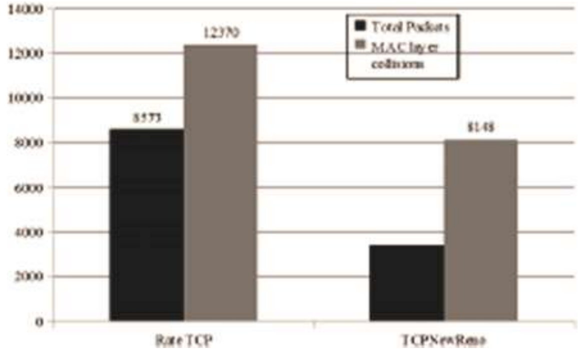
5.5 *MAC Layer Collisions*

Figure 10 represents the interference drops of MAC layer RTS/CTS packets. The collision also happens while transmitting the RTS. The collision is more in TCP than Rate TCP. Rate TCP had 12370 collisions for transmitting 8573 packets in the transport layer. Hence, in the last next section, we will conclude this work in brief.

6 Conclusion

This work proposed a rate control algorithm that is inserted as a middle ware between TCP and network layer. It accepts data from TCP but delivers them one after the other with sufficient interval among them. This interval is dynamically measured based on congestion that is occurred in the forward and reverse path. The exponential average of the end-to-end delay is considered as inter-packet delivery period. This kind of periodic transmission is suitable to the RFID network because reader to tag (passive)

Fig. 10 MAC layer collisions



communication also happens at a regular interval. The limitation of our algorithm is that it does not consider the spatial reuse property of the wireless medium. This issue/work can be considered as a future work.

References

1. Finkenzeller, K.: RFID Handbook. Wiley, New York (2003)
2. Sarma, S., Engels, D.W.: On the future of RFID tags and protocols. Technical report MIT-AUTOID-TR-018, Auto-ID Center (2003)
3. Vaidya, N., Das, S.R.: RFID-based networks: exploiting diversity and redundancy. SIGMOBILE Mob. Comput. Commun. **1**, 2–14 (2008)
4. Fu, Z., Zerfos, P., Luo, H., Lu, S., Zhang, L., Gerla, M.: The impact of multihop wireless channel on TCP throughput and loss. In: Conference of the IEEE Computer and Communication, vol. 3, pp. 1744–1753 (2003)
5. Zhang, X.M., Zhu, W.B., Li, N.N., Sung, D.K.: TCP Congestion Window Adaptation Through Contention Detection in Ad-Hoc Networks. Vehicular Technology, IEEE Transactions on **9**, 4578–4588 (2010)
6. Zhang, X., Li, N., Zhu, W., Sung, D.K.: CP transmission rate control mechanism based on channel utilization and contention ratio in Ad-hoc networks. IEEE Commun. Lett. **4**, 280–282 (2009)
7. ElRakabawy, S.M., Lindemann, C.: A practical adaptive pacing scheme for TCP in multihop wireless networks. IEEE/ACM Trans. Netw. **4**, 975–988 (2011)
8. Sundaresan, K., Anantharaman, H.-Y.H., Sivakumar, A.R.: ATP: a reliable transport protocol for ad hoc networks. IEEE Trans. Mob. Comput. **6**, 588–603 (2005)
9. Hamadani, E., Rakocevic, V.: TCP contention control: a cross layer approach to improve TCP performance in multihop AdHoc networks. In: Proceeding software 5th international conference on Wired/Wireless Internet Communications (WWIC'07), pp. 1–16. Springer, Berlin (2007)
10. Jubari, A., Othman, M.: An adaptive delayed acknowledgment strategy to improve TCP performance in multi-hop wireless networks, pp. 0929–6212 (2012)
11. Chen, J., MarioGerla, Y.L., Sanadidi, M.Y.: TCP with delayed ack for wireless networks. Ad Hoc Netw. **7**, 1098–1116 (2008)
12. Lun, T., Yifu, L., Qianbin, C.: Optimized beaconing rate control for vehicular Ad-Hoc networks. J. China Univ. Posts Telecommun. **22**(6), 10–17 (2015)

13. Zhou, K., Gong, C., Nan, W., Zhengyuan, X.: Distributed channel allocation and rate control for hybrid FSO/RF vehicular Ad Hoc networks. J. Opt. Commun. Netw. **9**(8), 669–681 (2017)
14. Giang, P.T., Nakagawa, K.: Cooperation between channel access control and TCP rate adaptation in multi-Hop Ad Hoc networks **98.B**, 79–87 (2015)
15. Armaghani, F.R., Jamuar, S.S., Khatun, S., Rasid, M.F.A.: Performance analysis of TCP with delayed acknowledgments in Multi-hop Ad-hoc networks. Wirel. Pers. Commun. **4**, 791–811 (2011)
16. De Cordeiro, M., Das, S.R., Agrawal, D.P.: COPAS: dynamic contention-balancing to enhance the performance of TCP over multi-hop wireless networks. In: Proceedings of the 10th International Conference on Computer Communications, and Networks, pp. 382–387 (2002)