electronics

MDPI

Article

TCP-LoRaD: A Loss Recovery and Differentiation Algorithm for Improving TCP Performance over MANETs in Noisy Channels

Nurul I. Sarkar [1,*], Ping-Huan Ho [1], Sonia Gul [2] and Salahuddin Muhammad Salim Zabir [3]

[1] Computer Science and Software Engineering, Auckland University of Technology,
Auckland 1010, New Zealand; holillian1@gmail.com
[2] Department of Computer Networks, College of Computer Science & IT, King Faisal University,
Al-Ahsa 31982, Saudi Arabia; sdgul@kfu.edu.sa
[3] National Institute of Technology, Tsuruoka College, Tsuruoka 997-8511, Japan; szabir@tsuruoka-nct.ac.jp
* Correspondence: nurul.sarkar@aut.ac.nz

Abstract: Mobile Ad hoc Networks (MANETs) are becoming popular technologies because they offer flexibility in setting up anytime and anywhere, and provide communication support on the go. This communication requires the use of Transmission Control Protocol (TCP) which is not originally designed for use in MANET environments; therefore, it raises serious performance issues. To overcome the deficiency of the original TCP, several modifications have been proposed and reported in the networking literature. TCP-WELCOME (Wireless Environment, Link losses, and Congestion packet loss ModEls) is one of the better TCP variants suitable for MANETs. However, it has been found that this protocol has problems with packet losses because of network congestion as it adopts the original congestion control mechanism of TCP New Reno. We also found that TCP-WELCOME does not perform well in noisy channel conditions in wireless environments. In this paper, we propose a novel loss recovery and differentiation algorithm (called TCP-LoRaD) to overcome the above-mentioned TCP problems. We validate the performance of TCP-LoRaD through an extensive simulation setup using Riverbed Modeler (formerly OPNET). Results obtained show that the proposed TCP-LoRaD offers up to 20% higher throughput and about 15% lower end-to-end delays than the TCP-WELCOME in a noisy channel under medium to high traffic loads.

Keywords: TCP; TCP-LoRaD; TCP-WELCOME; MANET; noisy channel

check for updates

Citation: Sarkar, N.I.; Ho, P.-H.; Gul, S.; Zabir, S.M.S. TCP-LoRaD: A Loss Recovery and Differentiation Algorithm for Improving TCP Performance over MANETs in Noisy Channels. *Electronics* **2022**, *11*, 1479. https://doi.org/10.3390/electronics11091479

Academic Editor: Qingqi Pei

Received: 16 March 2022
Accepted: 2 May 2022
Published: 5 May 2022

1. Introduction

Mobile Ad hoc Networks (MANETs) are becoming a popular networking option in various scenarios. The popularity of MANET is mainly due to its easy and quick setup which can be accomplished on the go. These features make MANETs one of the best suitable networks to be used in areas where people's movements are random and ad hoc e.g., online networked gaming, war fields, disaster relief operations, and even at hotel lobbies, airports, cafes, and student campuses.

MANETs are wireless networks where the mobile nodes configure themselves on their own, without any centralized infrastructure to control and link with. Another important characteristic of MANETs is that the network nodes are not expected to be stationary, they are mobile devices, and the network mobility keeps changing. It is a challenge to maintain a high level of performance of the MANETs due mainly to mobility, battery-powered nodes, and multihop routing structure [1]. The traditional Transmission Control Protocol (TCP) does not work well in a mobile ad hoc network (MANET) environment due to link failure, network congestion, and wireless channel errors [2]. To overcome the MANET performance issues, researchers have proposed various TCP variants suitable for MANETs.

TCP-WELCOME (WELCOME stands for Wireless Environment, Link losses, and Congestion packet loss modEls) is one of the popular TCP variants suitable for use in wireless network environments [3]. However, TCP-WELCOME inefficiently handles packet losses due to network congestion. In this paper, we propose a novel loss recovery and differentiation algorithm called TCP-LoRaD to overcome the performance issues of TCP-WELCOME. This paper proposes a method to improve the performance of traditional TCP so that it can be effectively and efficiently used for wireless mobile ad hoc networks. Riverbed Modeler-based simulations are performed to evaluate the system performance.

The main contributions of this paper are summarized as follows:

- We propose a novel loss recovery and differentiation algorithm (TCP-LoRaD) to improve the throughput and end-to-end delays over MANETs in noisy channels.
- We develop an analytical system model for TCP-LoRaD to derive Round-trip time (RTT) and TCP data transmission rate, analytically to estimate the system performance. We then perform an extensive simulation (about 12 simulation investigations) to validate the performance of the proposed TCP-LoRaD.
- We implement new nodes (in C++) and the corresponding process models in the Riverbed Modeler simulation environment to study the performance of TCP-LoRaD and to compare it with the existing TCP-WELCOME. This is a significant piece of work contributing toward the implementation of TCP in wireless and mobile networks.

This paper is organized as follows. Section 2 reviews relevant literature on TCP variants for wireless environments. The proposed TCP-LoRaD algorithm is described in Section 3. In Section 4, we describe how we evaluate the performance of TCP LoRaD. We present the simulation results in Section 5. Finally, we conclude this paper in Section 6.

2. Related Work

A detailed literature review is being conducted to explore the existing work on TCP variants. For a better understanding of the topic to the readers, we focus on TCP variants that are being proposed to overcome the performance degrading issues in MANETs.

Transport Control Protocol (TCP) is one of the most popular Transport Layer protocols used on the Internet [4]. It is a connection-oriented protocol that communicates over a virtual connection using port numbers between two machines. TCP also supports flow, error, and congestion control mechanisms to ensure reliable communications. This paper focuses on the algorithm that deals with the recovery of packets which is lost due to network congestion. Our discussion is limited to the error and congestion control of TCP [5]. In error control, both error detection and correction mechanisms are used. TCP uses three main parameters to detect and correct errors, such as acknowledgment, timeout, and checksum [5]. Similarly, for congestion control, TCP uses three phases which include congestion detection, congestion avoidance, and slow start [4]. Moreover, TCP faces even more challenges because of the frequent changes in network topology [6].

TCP is mainly designed to handle communications over wired networks. However, in MANETs, TCP becomes prone to some inherent features/problems of wireless channels [7]. These problems include loss of the wireless channel, loss of packets, node mobility, and high error rates. Many researchers have proposed solutions for improving the performance of TCP over the wireless network. One of the most popular variants of TCP is TCP- Wireless Link losses, and Congestion packet loss ModEls (TCP-WELCOME) [8]. TCP-WELCOME works as an end-to-end protocol and makes use of recovery and loss differentiation algorithms for lost packets. Moreover, estimating parameters at the sender side instead of getting them from the receiver, reduces the overhead by eliminating the need for sender-receiver synchronization. Moreover, loss recovery and differentiation algorithms are used to enhance end-to-end communications. The TCP-Adaptive Westwood (TCP-AW) is another TCP variant that is a congestion control algorithm based on the rate and packet delays. It used both TCP-Adaptive Bandwidth Share Estimation (TCPW-ABSE) [4] and TCP-Adaptive Reno [9]. TCP-AW increases network utilization by estimating the transmission rate and packet loss information.

TCP performance heavily depends on the detection of network congestion. In [10], authors have proposed an Enhanced Explicit Congestion Notification (EECN) mechanism to help TCP to make early detection of network congestion. It has been found that EECN improves the TCP average throughput, bandwidth, and round-trip-time (RTT).

Enhanced-TCP (E-TCP) is another TCP variant that works at multiple layers which prioritize the acknowledgments (ACKs) used in TCP and adjust the timer for retransmission, especially for real-time traffic (e.g., voice). It is found that E-TCP can utilize the communication channel more efficiently by increasing good-put and reducing packet delays [11]. In [12], researchers have introduced an approach to adjust Random Early Detection (RED); a TCP congestion control mechanism so the TCP performance over MANET is improved. In [13], researchers have proposed TCP-UB which combines the features of TCP-Vegas and TCP-Westwood. The TCP-UB provides a fair consumption of the bandwidth and consequently, it is more stable in congested networks.

In [14], researchers have reported that some congestions in the network occurred due to disaster events. To address such situations, the authors suggested using a multipath transport solution (MPTCP) which is routing traffic to less-congested parts of the network.

Another study conducted by Al-Zubi, R.T. et al. [15] suggests two improvements in existing TCP to enhance the network performance. First, using the concept of recycling the packets instead of dropping them to reduce the packet dropping ratio. Second, to use adaptive delay for acknowledgments allowing less network congestion. In [16], the authors have presented a congestion loss detection system for TCP. The idea is to focus on losses that occur in the system, not due to network congestion.

The review of literature on TCP variants is summarized in Table 1. The key researchers are listed in Column 1 and the corresponding main contributions are presented in Column 2. The limitations of the reviewed TCP variants are highlighted in Column 3. This summary table helps us to identify the research gaps and areas for contribution which is TCP-LoRaD.

Table 1. Summary of related work.

Researchers	Main Contribution	Limitation
Shahzad et al. (2020)	Developed congestion detection scheme to improve the TCP performance [10].	The scope of the investigation is limited to congestion detection.
Vevekananda et al. (2020)	CAM-SCTP [17]. Developed network traffic prioritization to achieve high performance.	The work is limited to streaming delay weights threshold.
Zhang et al. (2017)	Improved multipath hoping and optimized sub-paths for multipath TCP [18].	Limited investigation in terms of network topology.
Govindarajan et al. (2018)	Proposed an Enhanced TCP scheme to improve the end-to-end performance of MANETs [19].	Introduce high overheads due to protocol's reactiveness.
Dong et al. (2016)	Developed a mVeno algorithm with an adaptive transmission rate [20].	The power consumption aspect has not been addressed.
Reddy et al. (2016)	Developed TCP Friendly Rate Control scheme to reduce congestion [21].	The round-trip time for dynamically changing bandwidth has not been explored.
Manna et al. (2016)	ATCP (ad hoc TCP) [22] is proposed for listening and communicating network state information.	Mobility and interference management have not been considered in the study.
Sunitha et al. (2016)	SADCA [23] is developed to reduce packet transmission delays with acknowledgments.	The optimal delays of window size have not been explored.
Sharma and Patidar (2016)	Modified Hybrid-TCP [24] is proposed for rate increment.	Packet losses have not been studied.
Al-Zubi et al. (2014)	TCP-PR and TCP-ADW [15] are proposed for recycling a packet and using adaptive delay for packet acknowledgment.	The effect of the proposed solution on delays has not been explored.
Kuman and Singh (2014)	Proposed a congestion control mechanism in routing algorithm based on traffic rate and queue length [25].	Energy consumption has not been explored.

Table 1. *Cont.*

Researchers	Main Contribution	Limitation
Sharma et al. (2014)	Proposed a congestion-less, loss detection scheme for TCP [16].	End-to-End delay performance has not been explored.
Sangolli et al. (2014)	Proposed a cross-layer scheme on window size with energy-efficient routing protocol [26].	Performance parameters have not been explored.
Wazid et al. (2013)	E-TCP [27] is proposed under JellyFish delay variance attack.	The packet dropping aspect has not been studied well.
Elmannai et al. (2012)	TCP-UB [13]. Proposed a TCP variant by combining TCP-Vegas and TCP-Westwood.	Highly congested networks have not been investigated.
Könsgen et al. (2012)	MPTCP [14]. Investigated network congestion through a multipath transport solution.	Scheduling information for QoS is not addressed.
Bansal and Singh (2012)	Worked on TCP congestion and link instability problems [28].	The hidden node problem has not been studied.
Seddik-Ghaleb et al. (2009)	Proposed TCP-WELCOME [8] to address packet losses and recovery.	Performance for complex scenarios has not been explored.
Mbarushimana and Shahrabi. (2009)	Proposed E-TCP [11] for avoiding unnecessary retransmission and traffic starvation.	Low priority traffic provisioning has not been explored.
Wu et al. (2007)	Proposed TCP-HO [29] to minimize handoff delay by estimating link bandwidth.	High-density node performance has not been studied.
Kim et al. (2006)	Proposed TCP-Vegas and TCP-Reno for smooth integration and performance evaluation [30].	The effect of terrain has not been investigated.
Our work: TCP-LoRaD	TCP-LoRaD provides improved performance over MANETs in noisy channels because of its new loss recovery and differentiation algorithms.	

3. Description of the Proposed TCP-LoRaD

The TCP-WELCOME is one of the good variants of TCP suitable for use in MANET environments [8]. However, this TCP-WELCOME loses a lot of packets when the network becomes congested because its congestion control mechanism is based on TCP New Reno.

In this section, we describe our proposed TCP-LoRaD method by focusing on loss recovery and differentiation algorithms. Figure 1 shows the flowchart of TCP-LoRaD algorithms. By incorporating loss differentiation and loss recovery algorithms, TCP-LoRaD detects congestion and acts appropriately according to the identified packet loss model. It uses the features of TCP-AW [9] to improve the early congestion detection and handles network congestion-related packet losses more efficiently than the traditional congestion control mechanism of TCP-New Reno proposed in TCP-WELCOME.

3.1. Loss Differentiation Algorithm

TCP-WELCOME aims to implement the loss differentiation algorithm to identify the data packet losses in MANET environments. The idea is to correctly differentiate the various packet loss models, such as network congestion, link failure, and wireless channel error. The TCP-WELCOME loss differentiation algorithm helps TCP to identify the causes of packet losses at the sender side by observing the evolution of RTT samples history of sent packets over the connection, RTO, and three duplicated acknowledgments of packets received (i.e., the triggers of packet loss). Maintaining the RTT sample history of sent packets is not an efficient method because it takes a long time to build the results from RTT history. The congestion is not identified immediately and consequently creates significant delays in the system and the discarding of a lot of packets. We overcome the problem of delays and packet discarding in TCP-WELCOME by developing an efficient Loss Differentiation Algorithm presented in this section.

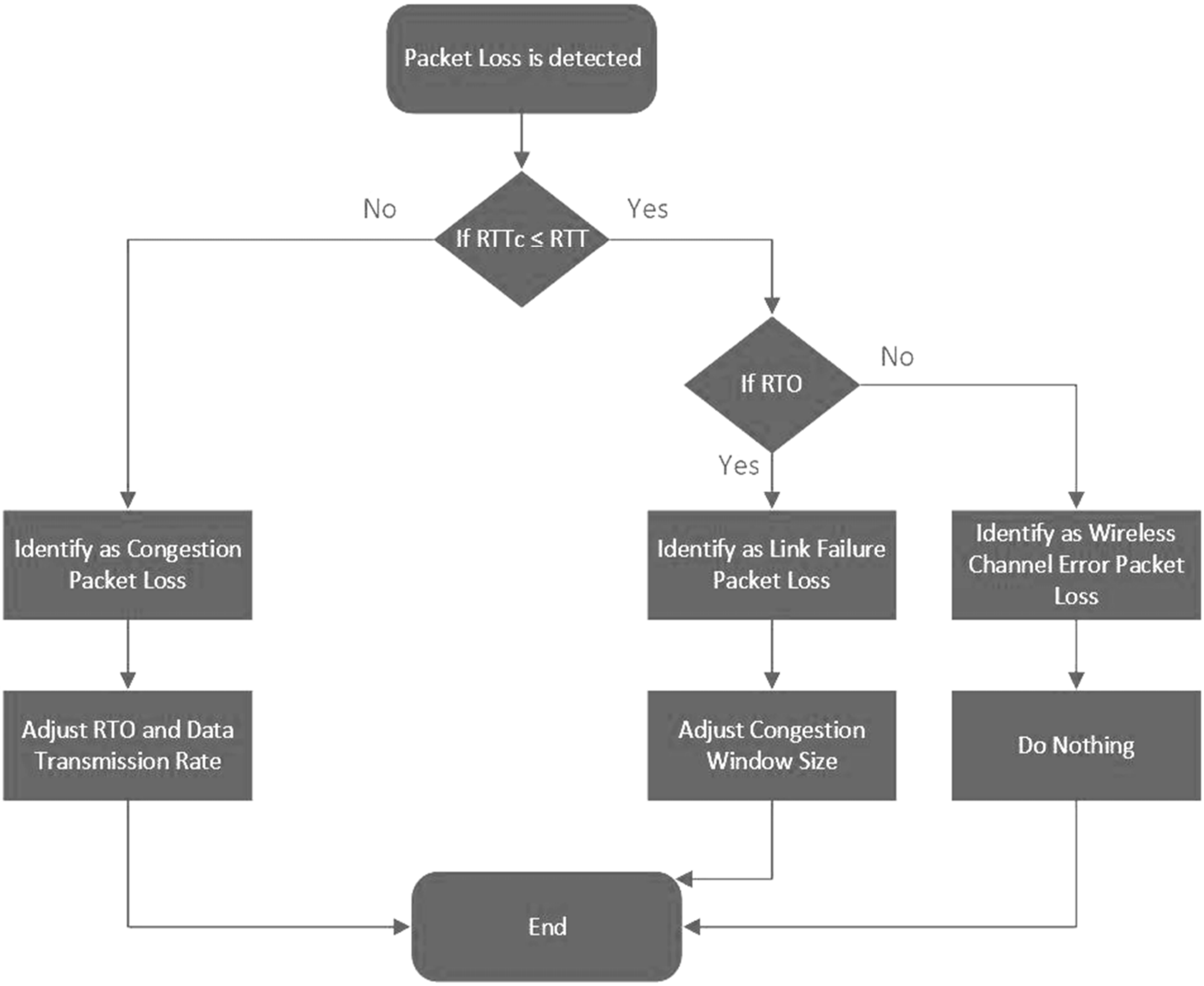


Figure 1. Flowchart of the proposed TCP-LoRaD Algorithm.

To identify the discarded packets more accurately during transmissions, TCP-LoRaD uses the loss differentiation algorithm to identify the packet loss model at the sender side. This is done by triggering Retransmission Timeout (RTO), three duplicated ACKs, and the Round-trip time (RTT) during transmissions.

By using the Incipient Congestion Detect mechanism introduced in TCP-AW, TCP-LoRaD detects network congestion before increasing the RTT value. This would take a long time as results need to be built upon RTT history, the congestion is not identified immediately and therefore the packets started to discard. The modified RTT value is introduced in incipient congestion detection and is defined as:

$$\mathrm{RTT}^{c} = \mathrm{RTT} + \max\left\{\left(\frac{\frac{\mathrm{CWND}}{\mathrm{RTT}}}{\mathrm{ERE}} - 1\right)\gamma, 0\right\} \tag{1}$$

The lightweight Eligible Rate Estimation (ERE) calculation algorithm proposed in TCP-LoRaD resembles TCP-AW but does not require ACK history.

$$\mathrm{S}_{\mathrm{rate}} = \frac{\mathrm{CWND}}{\mathrm{RTTmin}} \tag{2}$$

$$\mathrm{T_k} = \mathrm{RTT} \times \frac{\left(\mathrm{S_{rate}} - \mathrm{ERE_{prev}}\right)}{\mathrm{S_{rate}}} \quad (3)$$

$$\mathrm{ERE_{sample}} = \frac{\mathrm{Acknowledged_{bytes}}}{\mathrm{T_k}} \quad (4)$$

$$\delta = \frac{4\mathrm{CWND} - \mathrm{Acknowledged_{bytes}}}{4\mathrm{CWND} + \mathrm{Acknowledged_{bytes}}} \quad (5)$$

$$\mathrm{ERE_k} = \delta\ \mathrm{ERE_{k-1}} + (1-\delta)\mathrm{ERE_{sample}} \quad (6)$$

where, Srate is the instantaneous sending rate, and EREprev is the generated ERE value used in the previous transmission.

Tk is the time taken when the ERE sample is calculated using the arriving interval of ACK k, and Tj is the time taken when ERE is calculated using the arriving interval of ACK. δ is the time-vary coefficient at Tk.

The value of RTT is observed for the identification of packet losses. It can identify the packet losses due to both link failure and network congestion. If the RTT value is less or equal to the RTO value, it means that the packet loss is due to link failure otherwise the packet losses are due to network congestion. Moreover, to distinguish between link failure and wireless-related packet losses, the parameters related to the triggers of the retransmission are monitored. The RTO parameter is useful to identify whether the data transmission link between the sender and receiver nodes is down or not. However, the wireless-related packet loss is generally detected by three duplicated ACKs. Figure 2 illustrates the key concept of the loss differentiation algorithm of TCP-LoRaD.

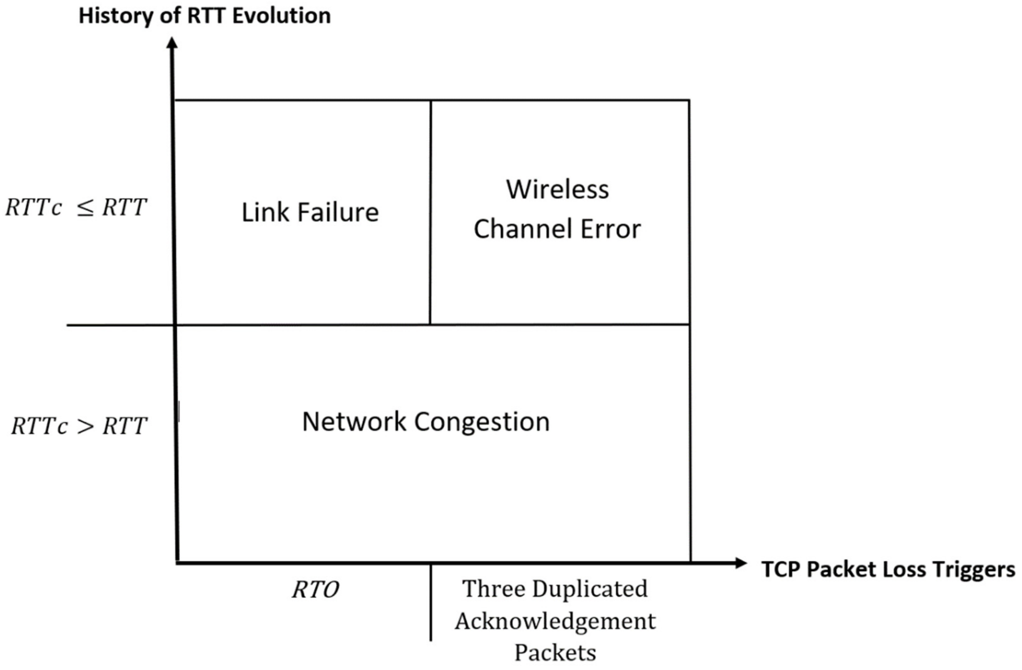


Figure 2. Illustrating the key concept of loss differentiation algorithm of TCP-LoRaD.

We observe that when RTT^c (new RTT value) is less than or equal to RTT and met the retransmission time out (RTO), the packet losses are due to link failure. However, when RTTc is greater than RTT, the packet losses are due to network congestion.

3.2. Loss Recovery Algorithm

For loss recovery, two parameters, namely RTO and duplicated ACKS are being used. These parameters are used to adjust and update the CWND and RTO values to come up with an improved congestion recovery process. The loss recovery algorithm of TCP-LoRaD is illustrated in Figure 3.

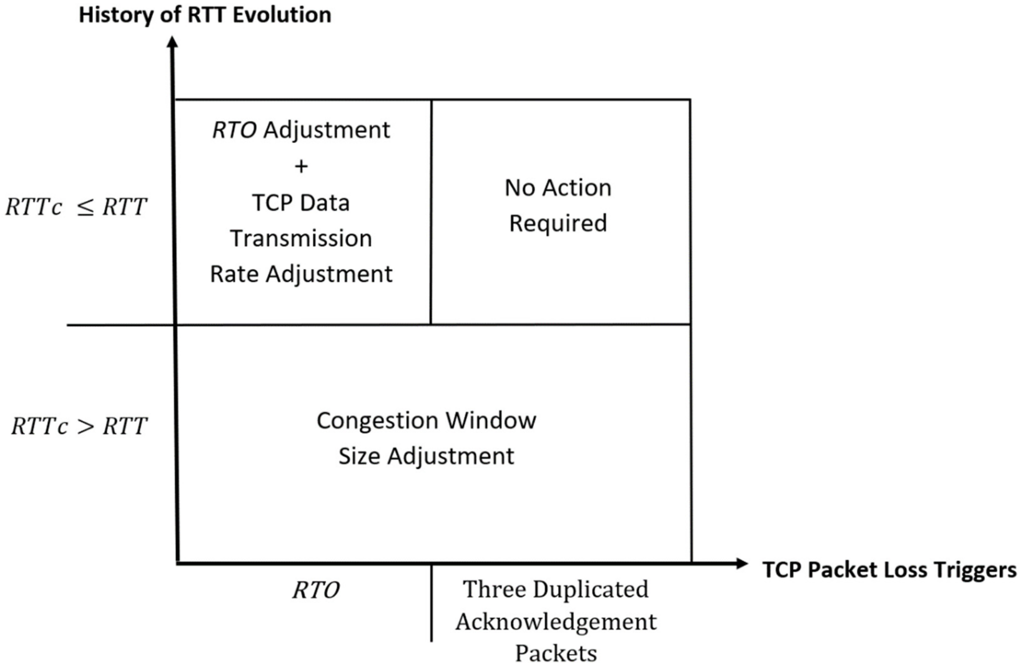


Figure 3. Illustrating the loss recovery algorithm of TCP-LoRaD.

TCP-LoRaD uses the congestion control algorithm introduced in TCP-AW for recovery if the packet loss model is identified as network congestion-related. To ensure RTT fairness, the CWND is increased based on the packet loss interval time. The adjustment of CWND can be written as:

$$\mathrm{CWND_{dif}} = \left(\alpha \mathrm{L} - \mathrm{CWND}\frac{\mathrm{RTT} - \mathrm{RTT_{min}}}{\mathrm{RTT}}\right)\frac{\mathrm{RTT_{min}}}{\mathrm{RTT_{ref}}} \tag{7}$$

$$\mathrm{CWND} = \mathrm{CWND} + \mathrm{CWND_{dif}}, \quad \text{if } \mathrm{CWND_{dif}} \leq 0 \tag{8}$$

$$\mathrm{CWND} = \mathrm{CWND} + \mathrm{CWND_{dif}} e^{-c}, \quad \text{if } \mathrm{CWND_{dif}} > 0 \tag{9}$$

where, α is the coefficient of congestion window increment, and the value is assigned to 0.2 based on the non-congestion time interval, which can increase friendliness and gain more network utilization. L is the period when two network congestion-related packet losses occur. c is the packet loss time interval, which can reflect the congestion level in the long term, and the value of c would be less than 0.4 so it is more accurate for congestion level estimation. $\mathrm{RTT_{ref}}$ is a gain factor for the effect of each error on RTT. There is no change proposed in the recovery process of packet losses due to link failure; it will remain the same as defined in TCP-WELCOME.

3.2.1. Adjustment of RTO Value

The Retransmission Timeout (RTO) value needs to be recalculated based on the queuing delays and propagation of the newly discovered path. Since both queuing delays and propagation influence the Round-Time-Trip (RTT) value and it can reflect the channel conditions over the connection directly, so the adjustment of the RTO value is based on the RTT values. The calculation of the new RTO value is given below.

$$\mathrm{RTO_{new}} = \left(\frac{\mathrm{RTT_{new}}}{\mathrm{RTT_{old}}}\right) \times \mathrm{RTO_{old}} \tag{10}$$

where, $\mathrm{RTT_{old}}$ is the RTT value over the old route before it was down, and $\mathrm{RTT_{new}}$ is the RTT value over the new route that is discovered. The new RTO value is updated after collecting a certain number of RTT samples ensuring accurate estimation.

3.2.2. Adjustment of TCP Transmission Rate

The conditions of packet buffering and queuing in the end nodes and the path capacity are the main parameters that need to be considered when updating the size of the congestion window. The rate of data transmission is estimated according to the characteristic of the newly discovered path, and it is calculated differently for each scenario. TCP can use the actual rate of data transmission and its congestion control mechanism to handle packet queuing and buffering before any packet losses are notified in the network. The RTT value is used here again because it can reflect the capabilities of the transmission links. The calculation of the new TCP transmission rate is given by

$$\mathrm{CWND_{new}} = \left(\frac{\mathrm{RTT_{old}}}{\mathrm{RTT_{new}}}\right) \times \mathrm{CWND_{old}} \tag{11}$$

where $\mathrm{CWND_{old}}$ is the size of the congestion window used for the previously failed route and $\mathrm{CWND_{new}}$ is the new congestion window size to be used over the newly discovered route.

4. Performance Evaluation

The performance of the proposed TCP-LoRaD is evaluated by quantitative discrete event simulation using Riverbed Modeler (formerly OPNET Modeler). The system is evaluated by measuring throughputs, packet delivery ratio, end-to-end delays, and retransmission attempts. These performance metrics are chosen because they are appropriate for TCP performance study.

The mean packet delay is defined as the average time (measured in slots) from the moment the packet is generated at a given station and joins its local queue until the packet is delivered to its destination. We use simulation to measure the mean packet delays at the nodes.

4.1. Modeling the Network

The simulation scenarios for TCP-LoRaD and TCP-WELCOME are implemented in Riverbed Modeler for system performance study. For MANETs, we used IEEE 802.11 standard networks with mobile nodes that only travel within the defined propagation range. These studies aim to evaluate the proposed TCP-LoRaD over MANETs in a noisy channel. The results for TCP-WELCOME are also presented for comparison purposes.

We investigate the effect of increasing the number of nodes (network size), node mobility, and packet lengths (Network traffic) on system performance in noisy channel conditions.

The general parameters used in the simulation are shown in Table 2. The AODV and the main TCP parameters used in the simulation are shown in Tables 3 and 4, respectively. To observe the impact of parameters on the performance of TCP-LoRaD over MANET in all possible scenarios, there would only be one control factor that is varied at a time in all defined scenarios. The control factors used to reflect the impact on TCP-LoRaD are thenumber of nodes, node speed, and packet length.

Table 2. General parameters used in the simulation.

Parameters	Value
Area	250 × 250 m^2
Number of nodes	5, 10, 15, and 20
Mobility model	Random waypoint (Auto Create)
IEEE 802.11 Data rate	11 Mbps
Transmission Power	0.005 W
Packet lengths	5000, 10,000, 15,000, 20,000, 25,000 bytes
Data type	FTP (File Transfer Protocol)
Mobility	3, 4, 5, 6, 7 m/s
Noise figure	5
Length of simulation	16 min simulated time

Table 3. AODV parameters used in the simulations.

Parameters	Value
Active Route Timeout	3
Hello Interval	Uniform (1,1.1) s
Allow Hello Loss	2
Net Diameter	35
Node Traversal Time	0.04
Route Error Rate Limit	10
Timeout Buffer	2

Table 4. Main TCP parameters used in the simulation.

Parameters	Value
Version	New Reno
Receive Buffer	8760 bytes
Receive Buffer Adjustment	None
Maximum ACK Delay	0.2 s
Maximum Segment Size (MSS)	1460 bytes
Maximum ACK Segments	2
Slow Start Initial Count	2 MSS
Fast Retransmit	Enable
Duplicate ACK Threshold	3
Fast Recovery	New Reno
Initial RTO	3 s
Minimum RTO	1 s
Maximum RTO	64 s
RTT Gain	0.125
Deviation Gain	0.25
RTT Deviation Coefficient	4.0

The moving area of the mobile nodes within the network boundary is defined as 250 m^2. The number of nodes used in the simulation represents network size. For example, we consider a small network with nodes up to 10; a small-to-medium network with nodes up to 15; and a medium-size network with nodes up to 20.

Noisy Channel Configuration: We study the performance of our proposed TCP-LoRaD over MANETs in noisy channels. We configured the noisy channel in Riverbed Simulator by setting up ‘noise figure’ to 5 (high level) and the noise model to WLAN noise. These configurations allow us to simulate the proposed TCP-LoRaD in noisy channel conditions. It should be noted that Riverbed Simulator is a credible simulation tool that offers various channel models including noisy channels for simulation tasks. Figure 4 shows a screenshot illustrating the noisy channel configuration in Riverbed Simulator.

To simplify the simulation model, the following assumptions are made throughout the simulation experiments.

A1. Battery life: The battery life of MANET nodes is not considered in the simulation.

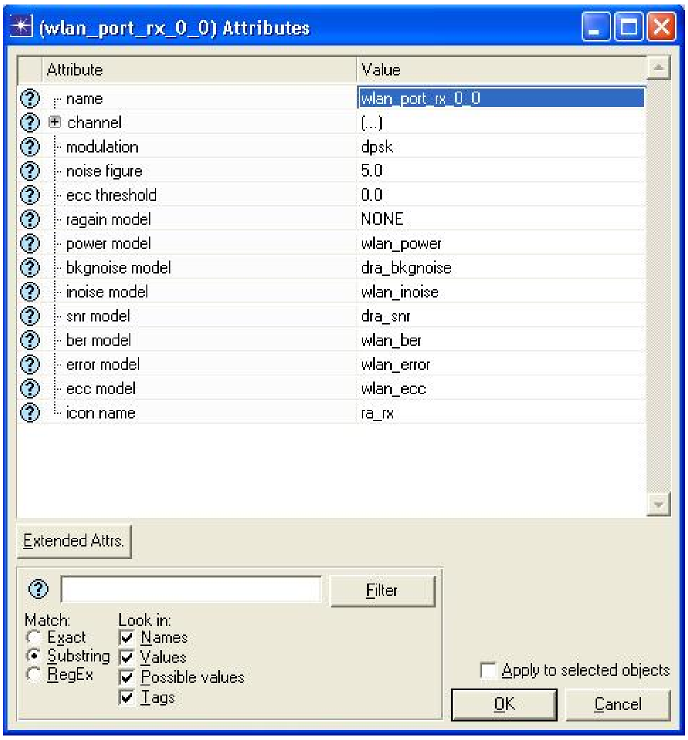
A2. Traffic: The nodes do not generate any traffic in the network.

A3. Node movement: All mobile nodes are moving within the network trajectory at predefined node speed (not all nodes are in movement at a given time).

A4. Analysis: We study the network performance in steady-state conditions.

4.2. Simulation Scenarios

Table 5 lists three simulation scenarios (12 investigations) that we considered in the study. The system performances are evaluated and reported in this paper for both the proposed TCP-LoRaD and the existing TCP-WELCOME. These scenarios are appropriate to study and compare the system performance under small-medium and medium-sized networks with varied traffic loads.



(wlan_port_rx_0_0) Attributes

Attribute	Value
name	wlan_port_rx_0_0
channel	(...)
modulation	dpsk
noise figure	5.0
ecc threshold	0.0
ragain model	NONE
power model	wlan_power
bkgnoise model	dra_bkgnoise
inoise model	wlan_inoise
snr model	dra_snr
ber model	wlan_ber
error model	wlan_error
ecc model	wlan_ecc
icon name	ra_rx

Match:
○ Exact
● Substring
○ RegEx

Look in:
☑ Names
☑ Values
☑ Possible values
☑ Tags

☐ Apply to selected objects

Figure 4. Noisy Channel Configuration in Riverbed Modeler Simulator.

Table 5. Network scenarios and the corresponding investigation and configurations.

Scenario	Investigation	Configuration
1. Effect of increasing the number of nodes (network size)	(1) Throughput versus number of nodes (2) End-to-end delay versus number of nodes (3) Packet delivery ratio versus number of nodes (4) Retransmission attempt versus number of nodes	Nodes = 5, 10, 15, 20 Speed: 5 m/s Traffic: FTP Packet size: 5000 bytes
2. Effect of increasing node speed (node mobility)	(5) TCP Throughput versus node speed (6) End-to-end delay versus node speed (7) Packet delivery ratio versus node speed (8) Retransmission attempt versus node speed	Speed: 3 to 7 m/s Node: 10 Traffic: FTP. Packet length: 5000 bytes
3. Effect of increasing the packet lengths (Traffic loads)	(9) TCP Throughput versus packet length (10) Delay versus packet length (11) Packet delivery ratio versus packet length (12) Retransmission attempt versus packet length	Packet length: 5000, 10,000, 15,000, 20,000, and 25,000 bytes Node: 10 Node speed: 5 m/s Traffic: File Transfer Protocol (FTP)

5. Results and Discussions

Based on the scenarios defined in Section 4.2 above, we conducted extensive simulations, and the results obtained are discussed next. All simulation results report the steady-state behavior of the network and were obtained with a relative statistical error $\leq$1%, at the 99% confidence level. Each simulation run lasted for 16 min simulated time where the first minute was the transient period. The simulation ended automatically when the results reached a 99% confidence interval. The observations collected during the transient period are not included in the final simulation results. We repeated each simulation experiment 10 times to obtain the mean results that are presented in this paper. The statistical errors of the simulation results are within $\leq$1% with a 90% confidence interval; therefore, we did not show the min, max, and standard deviation of all simulation results presented in this paper.

We measure throughputs, end-to-end delays, retransmission attempts, and packet delivery ratio.

5.1. Scenario 1: Effect of Increasing the Number of Nodes (Network Size)

Throughput versus the number of node: Figure 5 shows the throughput performance of TCP-LoRaD for increasing the number of nodes in the network (n = 5 to 20 nodes) representing a small, small-medium, and a medium-sized network for both good and noisy channel conditions. The performance of TCP-WELCOME is also shown for comparison purposes. We observe that the throughputs are decreasing with the increasing number of nodes in the network. This is expected because of more channel contention at a high number of nodes and consequently, the network-wide throughputs are decreasing. However, we found that TCP-LoRaD offers up to 20% higher throughputs than TCP-WELCOME. However, as the node size increases, the differences in throughputs are reduced between TCP-LoRaD and TCP-WELCOME.

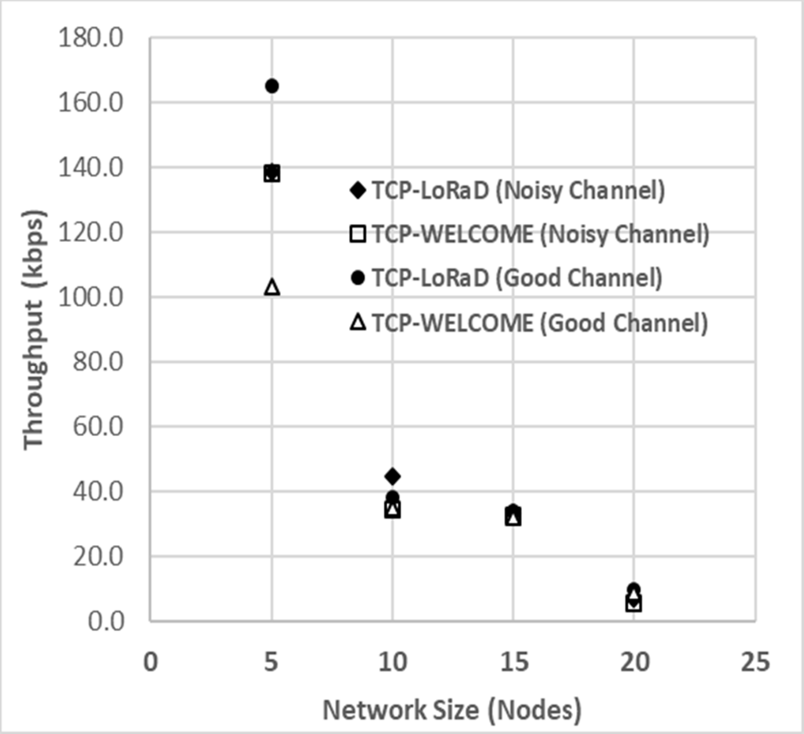


Figure 5. Throughput versus increasing the number of nodes in a noisy channel. Comparison of TCP-LoRaD and TCP-WELCOME. Results for good channel conditions are also presented for comparison purposes.

End-to-end delay versus number of node: Figure 6 shows the end-to-end delays for the proposed TCP-LoRAa in noisy channel conditions. Results for good channel conditions are also presented for comparison purposes. We observe that, as the number of nodes increases in the network, the network congestion occurs more often and hence TCP introduces longer end-to-end delays. However, TCP-LoRaD achieves up to 14% lower end-to-end delays than the existing TCP-WELCOME. The delay increases rapidly because of network congestion requiring longer queuing delays for the packet to be sent out at the buffer.

Packet delivery ratio versus number of nodes: In Figure 7, we plot the packet delivery ratios versus Network size (i.e., number of nodes) of the proposed TCP-LoRaD and the existing TCP-WELCOME in noisy channel conditions. Results for good channel conditions are also presented for comparison purposes. We observe that TCP-LoRaD can successfully receive more packets generated at the application layer in the small-medium and medium-sized networks. One can observe that TCP-LoRaD offers a 13% higher packet delivery ratio for a good channel while the performance of TCP LoRaD is better than TCP-WELCOME in noisy channels as well. As the node sizes increase, the packet delivery ratio drops due to congestion and end-to-end delays.

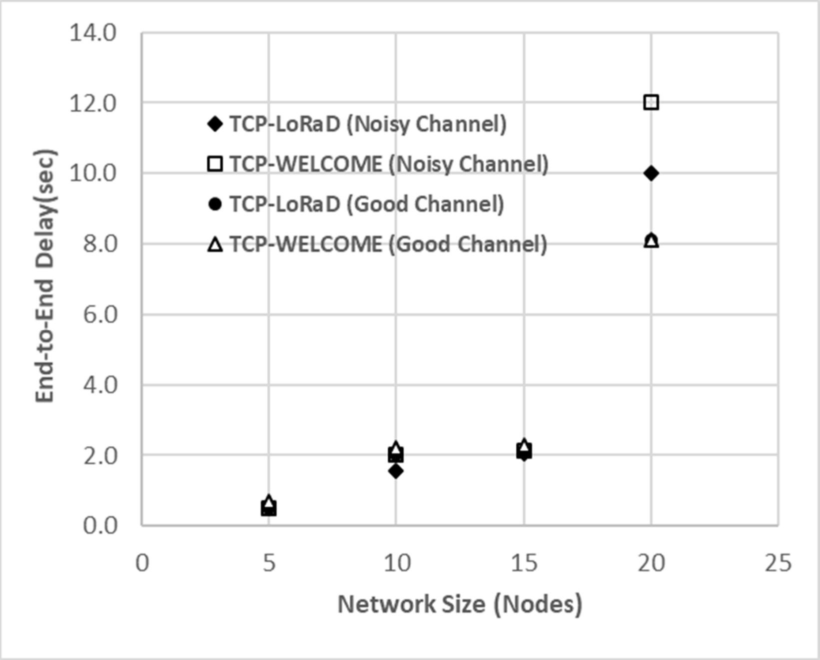


Figure 6. End-to-End delay versus increasing the number of nodes in a noisy channel. Comparison of TCP-LoRaD and TCP-WELCOME.

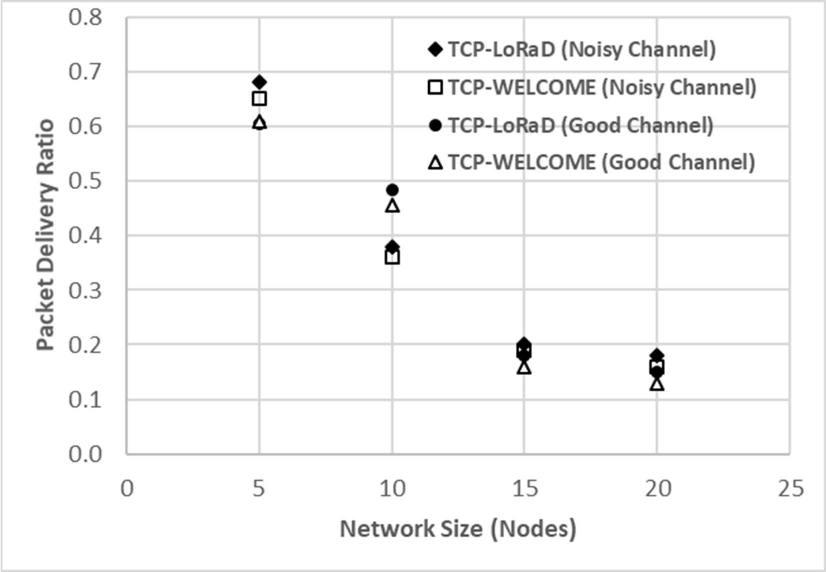


Figure 7. Packet Delivery Ratio versus increasing number of nodes in a noisy channel. Comparison of TCP-LoRaD and TCP-WELCOME.

Retransmission attempt versus the number of nodes: Figure 8 compares the retransmission attempts of both variants under the good and noisy channels in varying-sized networks. We observe that the heavy traffic load due to the noisy channel and bigger network size would result in more network congestion, and therefore the retransmission attempt increases rapidly as the network size enlarges. However, TCP-LoRaD sent out 6% fewer discarded packets than TCP-WELCOME. Therefore, TCP-LoRaD outperforms in data retransmission with varying network sizes.

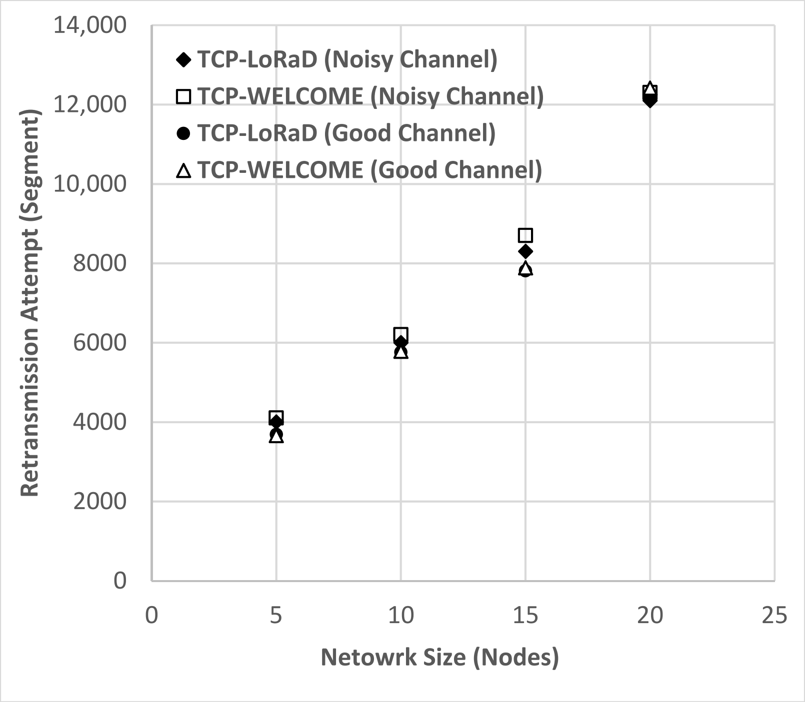


Figure 8. Retransmission attempt versus increasing number of nodes in a noisy channel. Comparison of TCP-LoRaD and TCP-WELCOME.

5.2. Scenario 2: Effect of Increasing Node Speed on System Performance in Noisy Channel

This section outlines the results acquired based on Scenario 2. This scenario represents the experimental results of TCP performance impact on varying node speeds under the noisy wireless channel and is presented with the different node speeds (3, 4, 5, 6, and 7 m/s), representing the people moving in the office.

TCP Throughput versus node speed: In Figure 9, we plot node mobility (3 to 7 m/s) against throughputs of TCP-LoRaD. The throughputs for TCP-WELCOME are also presented. We have also included throughputs for both TCP-LoRaD and TCP-WELCOME in good channel conditions for comparison purposes.

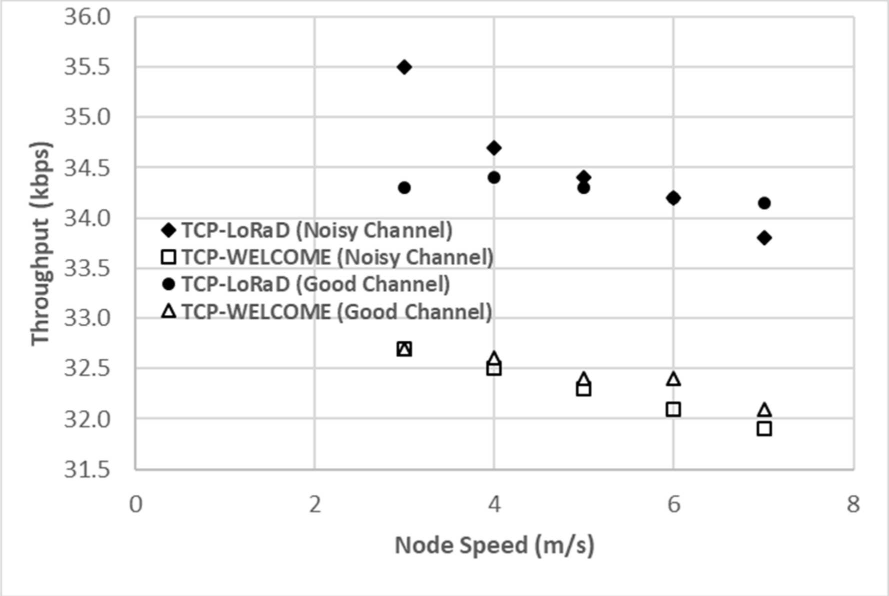


Figure 9. TCP Throughput versus node speed in a noisy channel. Comparison of TCP-LoRaD and TCP-WELCOME.

In noisy channels, TCP-LoRaD offers up to 13% higher throughputs than the existing TCP-WELCOME. The higher node mobility and the noisy wireless channel would cause the link failure and network congestion to occur more frequently during the communication, and therefore the generated throughputs decrease as the node speed increases. Moreover, under good channel conditions, TCP-LoRaD offers up to 6% higher throughputs than the existing TCP-WELCOME.

End-to-end delay versus node speed: Figure 10 compares the end-to-end delays of both variants in the noisy channel in the small-to-medium-sized network. We observe that TCP-LoRaD offers up to 6% lower end-to-end delays than TCP-WELCOME. The link failure and network congestion would occur more often because of the higher node speed and noisy channel. Consequently, the queuing delay is increased, and takes longer for the packet waiting to be sent out at the buffer; hence the end-to-end delay is affected and increases as the node speed increases.

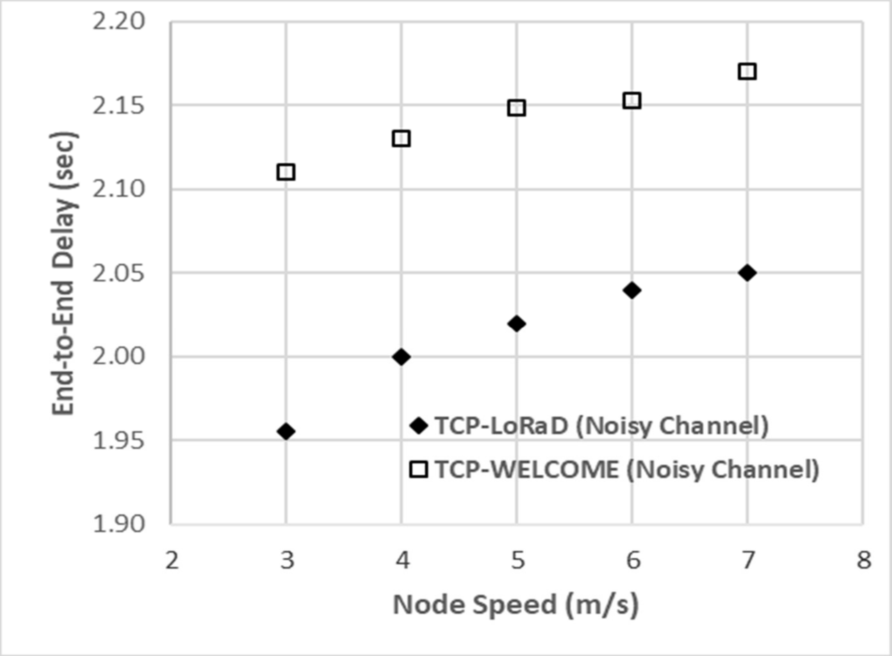


Figure 10. End-to-End delay versus node speed. Comparison of TCP-LoRaD and TCP-WELCOME.

Packet delivery ratio versus node speed: Figure 11 compares the packet delivery ratios of both variants in noisy channels. Results for a good channel are also presented for comparison purposes.

TCP-LoRaD can successfully receive larger amounts of data generated at the application layer without retransmission required than the existing TCP-WELCOME. In good channel conditions, TCP-LoRaD achieves up to 9% higher packet delivery rates than the TCP-WELCOME. The delivery rates decreased as the node speed increased. This is because the link failure due to the higher node mobility would cause more random packet losses hence the packet delivery ratio is influenced and reduced.

In a noisy channel, TCP-LoRaD achieves a 3% higher packet delivery ratio which allows the destination node to receive more data generated from the application at the destination nodes successfully without retransmission required. The link failure and network congestion caused by higher mobility and noisy communication link would result in more random packet losses, so the packet delivery ratios decrease with the increasing node speed.

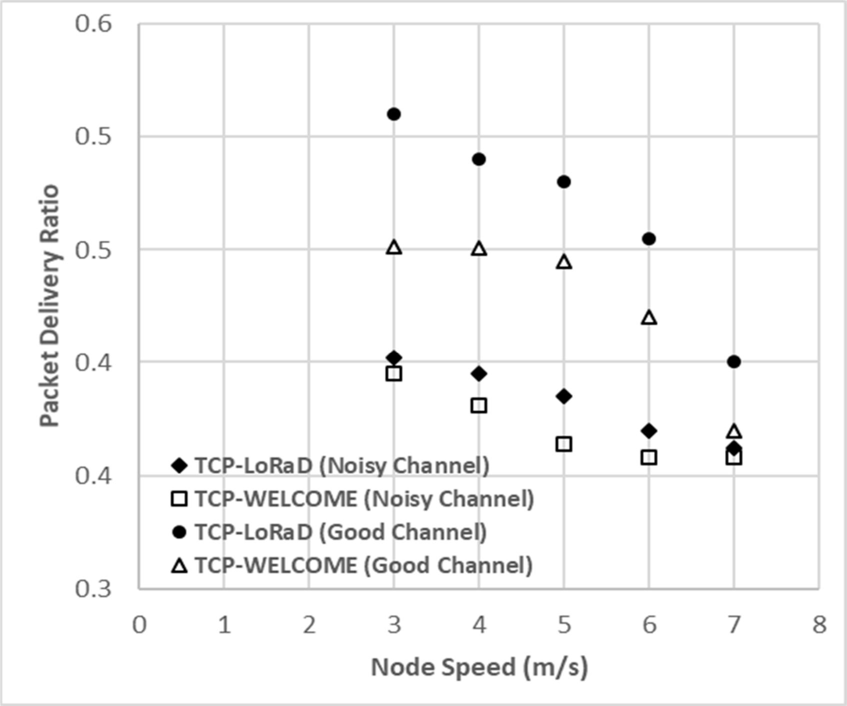


Figure 11. Packet Delivery Ratio versus node speed in a noisy channel. Comparison of TCP-LoRaD and TCP-WELCOME.

Retransmission attempt versus node speed: Figure 12 compares the retransmission attempts in noisy links with increasing node speeds. TCP-LoRaD achieves a 4% lower data retransmission attempt to recover from the packet loss. As shown in Figure 11, the significant outperformance of TCP-LoRaD proves that the proposed loss recovery algorithm takes more appropriate action to handle the packet loss due to link failure and network congestion and improve the performance of retransmission attempts.

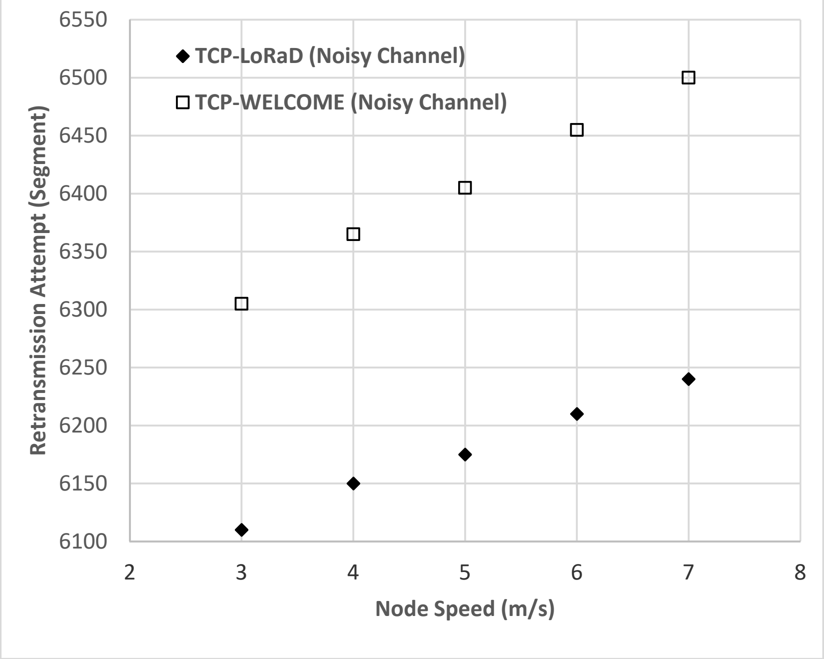


Figure 12. Retransmission Attempt versus node speed in a noisy channel. Comparison of TCP-LoRaD and TCP-WELCOME.

5.3. Scenario 3: Effect of Packet Lengths on System Performance in Noisy Channel

In this section, we investigate the impact of packet lengths (i.e., traffic loads) on TCP performance in the noisy wireless channel. We consider five different packet lengths (5000, 10,000, 15,000, 20,000, and 25,000 bytes) to represent medium to high traffic loads.

TCP Throughput versus packet length: In Figure 13, we plot throughput versus packet lengths of TCP-LoRaD in noisy channel conditions for a 10-node network. In this scenario, we investigate the impact of increasing packet lengths on system performance. Results for TCP-WELCOME are also presented for comparison purposes. In the simulation setup, the maximum segment size (MSS) was set to 1460 bytes which is a realistic figure given that IEEE 802.11 wireless Ethernet packet length is 1500 bytes including headers.

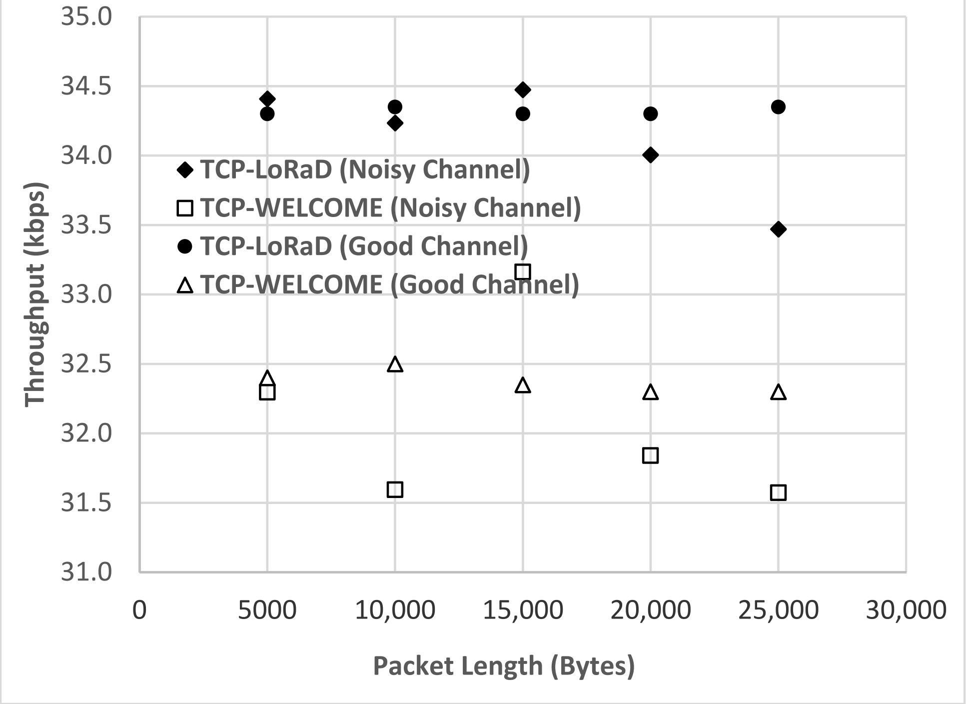


Figure 13. Throughput versus varying Packet Length in a noisy channel. Comparison of TCP-LoRaD and TCP-WELCOME.

We observe that the increasing packet lengths introduce more network traffic loads over the connection causing packet losses due to network congestion; hence the throughput decreases with the increase of packet lengths which is expected. However, we found that TCP-LoRaD achieves up to 7% higher throughputs than TCP-WELCOME in noisy channels.

End-to-end delay versus packet length: Figure 14 compares the end-to-end delays of both variants in a noisy channel with increasing packet lengths. We observe that the network congestion would occur more often due to heavy traffic loads generated by the increasing packet lengths. The network congestion introduces more queuing delay at the buffers during the data transmission, and therefore the end-to-end delay increases as the packet size increases. Overall, TCP-LoRaD achieves up to 6% lower end-to-end delays than TCP-WELCOME in a noisy channel.

Packet delivery ratio versus packet length: Figure 15 compares the packet delivery ratios of both variants under the noisy link with varying packet lengths. TCP-LoRaD achieves up to 18% higher packet delivery ratios than TCP-WELCOME allowing the receiver to successfully receive more data from the application layer without retransmissions. The noisy channel and increasing packet lengths would generate heavy traffic loads over the connection causing the network congestion to occur more frequently during the communication. The packet is discarded due to the network congestion, so the packet delivery ratio decreases as the packet length increases.

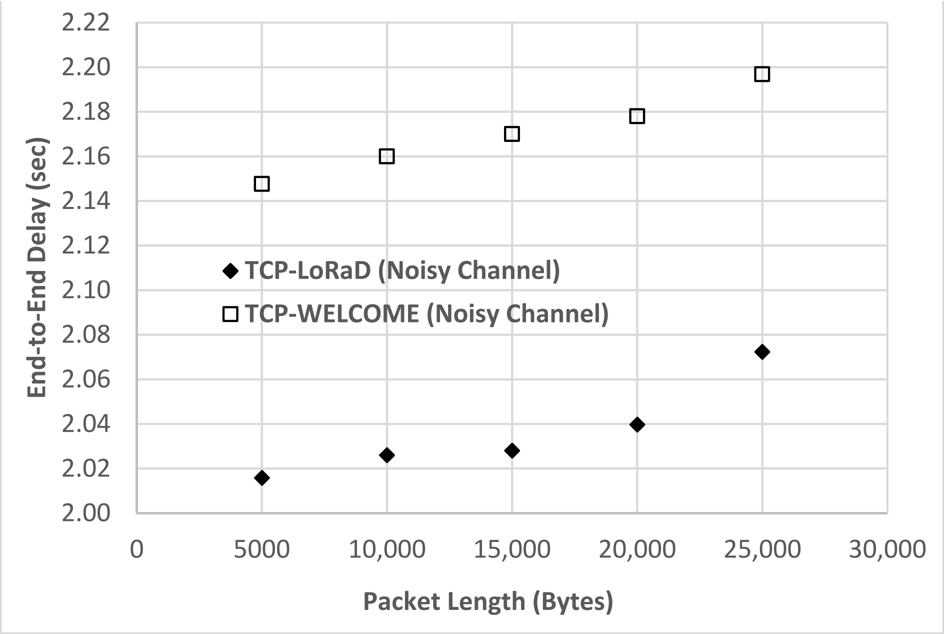


Figure 14. End-to-End delay versus Packet Length in a noisy channel. Comparison of TCP-LoRaD and TCP-WELCOME.

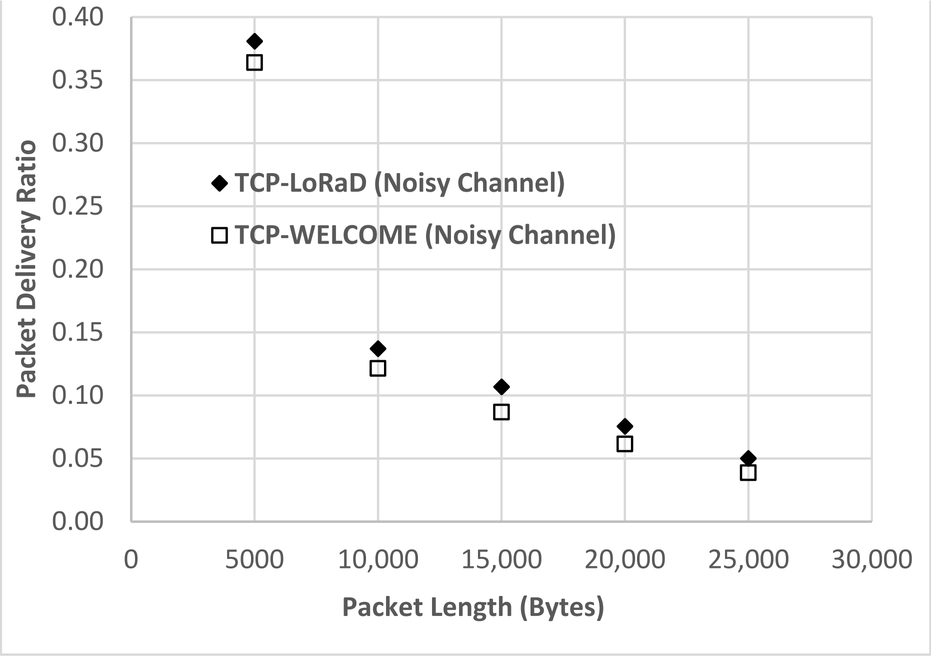


Figure 15. Packet delivery ratio versus Packet Length in a noisy channel. Comparison of TCP-LoRaD and TCP-WELCOME.

Retransmission attempt versus packet length: Figure 16 compares the retransmission attempts of both variants in noisy links with increasing packet lengths. TCP-LoRaD achieves a 3% lower data retransmission than the TCP-WELCOME to handle the discarded packets due to network congestion. This performance improvement confirms that the proposed TCP-LoRaD loss recovery mechanism of network congestion can take more suitable actions to recover the packet losses.

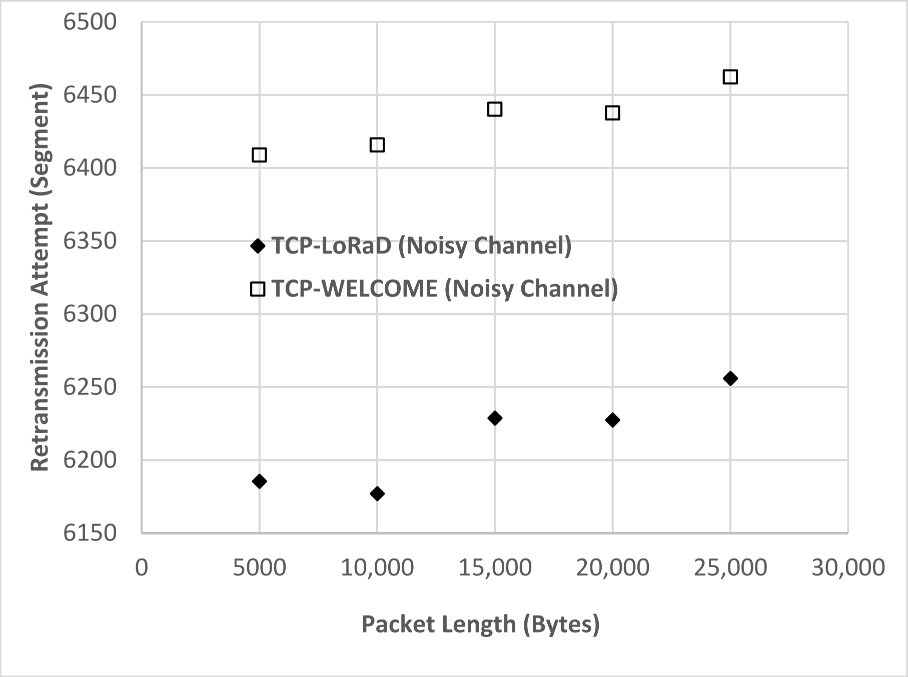


Figure 16. Retransmission attempt versus Packet Length in a noisy channel. Comparison of TCP-LoRaD and TCP-WELCOME.

5.4. Model Validation

Comparing results from the previous studies we validate the acquired results of simulation [31]. The simulation results presented in graphs are to ease the illustration and comparison of the proposed TCP-LoRaD and existing TCP-WELCOME. To ensure the accuracy of the simulation results, each simulation is replicated and run 10 times and the data collected from each run was recorded and compared. The graphical results of each replication can provide the similar or same illustration and therefore the analysis and interpretation of the simulation results could lead to the same conclusion.

As discussed in the literature [8], TCP-WELCOME outperformed some of the existing TCP variants, such as TCP-New Reno, TCP-SACK, TCP-Vegas, and TCP-Westwood. To compare the performance of TCP-LoRaD with that of TCP-WELCOME, we implemented both variants in the Riverbed Modeler simulator for simulation tasks.

The values of the general parameters implemented in the simulator were referred to as TCP-WELCOME. Moreover, the network size used for TCP-WELCOME evaluation in the literature [8] was 5 nodes and configured with a low node speed environment. Therefore, the performance evaluations of TCP-LoRaD impact on varying network size and node mobility were concentrated. The network size was relatively small with the node size of 5, 10, 15, and 20 nodes. The node mobility with the node speeds of 3, 4, 5, 6, and 7 m/s was considered to present the slow movement. Moreover, the performance impact of increasing packet lengths is explored in this paper. The packet lengths of 5000, 10,000, 15,000, 20,000, and 25,000 bytes were configured to represent medium to high traffic loads. We investigated the system performance in noisy channel conditions because it provides a real-world environment that is very challenging and hard to achieve good system performance.

6. Conclusions

In this paper, we proposed a novel transport layer protocol that we name TCP-LoRaD to improve the performance of TCP over MANETs in noisy channel conditions. TCP-LoRaD overcomes the packet loss problems due to TCP congestion control mechanisms in such environments. To verify the effectiveness of TCP-LoRaD, we conducted extensive simulations in realistic scenarios. We investigated the effect of increasing nodes, mobility, packet length, and conditions of the wireless channel (e.g., noisy channel) on system performance. It has been found that the effect of node speed and packet lengths have an impact on determining the throughput and packet delay performance of TCP-LoRaD in noisy channels. We compare the performance with that of TCP WELCOME, the existing high-performance TCP protocol for MANETs. Simulation results show that the proposed TCP-LoRaD yields up to 20% higher throughput, 15% lower packet delays, and 18% higher packet delivery ratios than TCP-WELCOME in a noisy channel under medium to high traffic loads. The findings reported in this paper provide some insights into TCP performance over MANETs in noisy channels that can help network researchers and engineers to contribute further towards developing transport protocols for the next-generation wireless networks. However, incorporating TCP-LoRaD in the design of the Internet of Things for saving human lives in an emergency application is suggested as future work.

Author Contributions: Conceptualization, N.I.S. and P.-H.H.; methodology, N.I.S. and P.-H.H.; software, N.I.S. and P.-H.H.; validation, N.I.S. and P.-H.H.; formal analysis, P.-H.H. and N.I.S.; investigation, P.-H.H. and N.I.S.; resources, N.I.S.; data curation, P.-H.H. and N.I.S.; writing—original draft preparation, N.I.S.; writing—review and editing, N.I.S., S.G. and S.M.S.Z.; visualization, N.I.S., P.-H.H. and S.G.; supervision, N.I.S.; project administration, N.I.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest. We also confirm that our paper has not been previously published and is not currently under consideration by any other journals. All authors involved have approved the contents of this paper and have agreed to the Journal of MDPI Electronics submission policies.

References

1. Ahmad, A. *Wireless and Mobile Data Networks*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2005.
2. Casetti, C.; Gerla, M.; Mascolo, S.; Sanadidi, M.Y.; Wang, R. TCP westwood: End-to-end congestion control for wired/wireless networks. *Wirel. Netw.* **2002**, *8*, 467–479. [CrossRef]
3. El-Sayed, H.M.; Bazan, O.; Qureshi, U.; Jaseemuddin, M. Performance evaluation of TCP in mobile ad hoc networks. In Proceedings of the Second International Conference on Innovations in Information Technology, Dubai, United Arab Emirates, 26–28 September 2005.
4. Wang, R.; Valla, M.; Sanadidi, M.Y.; Gerla, M. Using adaptive rate estimation to provide enhanced and robust transport over heterogeneous networks. In Proceedings of the 10th IEEE International Conference on Network Protocols, Washington, DC, USA, 12–15 November 2002; pp. 206–215.
5. Forouzan, B.A.; Fegan, S.C. *Data Communications and Networking*; McGraw-Hill: New York, NY, USA, 2007.
6. Henna, S. A Throughput Analysis of TCP Variants in Mobile Wireless Networks. In Proceedings of the Third International Conference on Next Generation Mobile Applications, Services and Technologies, Cardiff, UK, 16–19 September 2009.
7. Liu, J.; Yougu, Y.; Nicol, D.M.; Gray, R.S.; Newport, C.C.; Kotz, D.; Perrone, L.F. Simulation Validation using direct execution of wireless ad-hoc routing protocol. In Proceedings of the 18th Workshop on Parallel and Distributed Simulation, Kufstein, Austria, 16–19 May 2004.
8. Seddik-Ghaleb, A.; Ghamri-Doudane, Y.; Senouci, S.-M. TCP WELCOME TCP variant for Wireless Environment, Link losses, and COngestion packet loss ModEls. In Proceedings of the First International Communication Systems and Networks and Workshops, Bangalore, India, 5–10 January 2009.
9. Marcondes, C.; Sanadidi, M.Y.; Gerla, M.; Shimonishi, H. TCP Adaptive Westwood- Combining TCP Westwood and Adaptive Reno: A Safe Congestion Control Proposal. In Proceedings of the IEEE International Conference on Communications, Beijing, China, 19–23 May 2008; pp. 5569–5575.
10. Shahzad, S.; Jung, E.S.; Chung, J.; Kettimuthu, R. Enhanced Explicit Congestion Notification (EECN) in TCP with P4 Programming. In Proceedings of the International Conference on Green and Human Information Technology (ICGHIT), Hanoi, Vietnam, 5–7 February 2020; pp. 35–40.
11. Mbarushimana, C.; Shahrabi, A. E-TCP: Enhanced TCP for IEEE802.11e Mobile Ad Hoc Networks. In Proceedings of the 15th International Conference on Parallel and Distributed Systems, Shenzhen, China, 8–11 December 2009; pp. 632–639.
12. Yang, S.J.; Lin, Y.C. RED tuning for TCP performance on the mobile ad hoc networks. In Proceedings of the 11th International Conference on Parallel and Distributed Systems, Washington, DC, USA, 20–22 July 2005; pp. 412–418.
13. Elmannai, W.; Razaque, A.; Elleithy, K. TCP-UB: A New congestion aware transmission control protocolvariant. *arXiv* **2012**, arXiv:1210.2025. [CrossRef]

14. Könsgen, A.; Singh, A.; Jun, M.; Weerawardane, T.; Goerg, C. Responsiveness of future telecommunication networks under disaster situations. In Proceedings of the IEEE 4th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), St. Petersburg, Russia, 3–5 October 2012; pp. 892–899.
15. Al-Zubi, R.T.; Krunz, M.; Al-Sukkar, G.; Hawa, M.; Darabkh, K.A. Packet recycling and delayed ACK for improving the performance of TCP over MANETs. *Wirel. Pers. Commun.* **2014**, *75*, 943–963. [CrossRef]
16. Sharma, N.; Chakrawarti, R.K. Simulation for congestion-less losses control over MANET using TCP scheme. In Proceedings of the IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), Ghaziabad, India, 7–8 February 2014; pp. 410–415.
17. Vivekananda, G.N.; Ilknur, A. A congestion avoidance mechanism in multimedia transmission over MANET using SCTP multi-streaming. *Multimed. Tools Appl.* **2020**, *79*, 16823–16844. [CrossRef]
18. Zhang, T.; Zhao, S.; Ren, B.; Shi, Y.; Cheng, B.; Chen, J. Performance enhancement of multipath TCP in mobile Ad Hoc networks. In Proceedings of the IEEE 25th International Conference on Network Protocols (ICNP), Toronto, ON, Canada, 10–13 October 2017; pp. 1–2.
19. Govindarajan, J.; Vibhurani, N.; Kousalya, G. Enhanced TCP NCE: A modified non-congestion events detection, differentiation and reaction to improve the end-to-end performance over manet. In *Progress in Intelligent Computing Techniques: Theory, Practice, and Applications*; Springer: Singapore, 2018.
20. Dong, P.; Wang, J.; Huang, J.; Wang, H.; Min, G. Performance enhancement of multipath TCP for wireless communications with multiple radio interfaces. *IEEE Trans. Commun.* **2016**, *64*, 3456–3466. [CrossRef]
21. Reddy, N.R.; Pakanati, C.R.; Padmavathamma, M. Performance Enhancement of TCP Friendly Rate Control Protocol over Wired Networks. *Int. J. Electr. Comput. Eng.* **2016**, *6*, 2949–2954.
22. Manna, S.; Mondal, A.K.; Roy, P. Enhancement of TCP Performance over MANET. *Int. Res. J. Eng. Technol. (IRJET)* **2016**, *3*, 714–716.
23. Sunitha, D.; Nagaraju, A.; Narsimha, G. Cross-layer based Smart Acknowledgment Scheme for TCP in MANETs. *Int. J. Comput. Sci. Netw. Secur.* **2016**, *16*, 124–130.
24. Sharma, N.; Patidar, G. Improved Congestion Control Mechanism Using Modified Hybrid-TCP in Mobile Ad-Hoc Neworks. In Proceedings of the Second International Conference on Computational Intelligence & Communication Technology (CICT), Ghaziabad, India, 12–13 February 2016.
25. Kumar, H.; Singh, P. TCP congestion control with delay minimization in MANET. In Proceedings of the IEEE International Conference on Information Communication and Embedded Systems (ICICES2014), Chennai, India, 27–28 February 2014.
26. Sangolli, S.V.; Thyagarajan, J. An efficient congestion control scheme using cross-layered approach and comparison of TCP variants for mobile ad-hoc networks (MANETs). In Proceedings of the IEEE First International Conference on Networks & Soft Computing (ICNSC2014), Guntur, Andhra Pradesh, India, 19–20 August 2014; pp. 30–34.
27. Wazid, M.; Katal, A.; Sachan, R.S.; Goudar, R.H. E-TCP for efficient performance of MANET under JF delay variance attack. In Proceedings of the IEEE Conference on Information & Communication Technologies, Thuckalay, Tamil, India, 11–12 April 2013; pp. 145–150.
28. Bansal, A.; Singh, M. Enhancing MANET's performance: A cross-layer solution. In Proceedings of the 2nd IEEE International Conference on Parallel, Distributed and Grid Computing, Solan, Himachal Pradesh, India, 6–8 December 2012; pp. 407–412.
29. Wu, X.C.; Choon, M.; Ananda, A.L. TCP HandOff: A Practical TCP Enhancement for Heterogeneous Mobile Environments. In Proceedings of the IEEE International Conference on Communications, Glasgow, UK, 24–28 June 2007; pp. 6043–6048.
30. Kim, D.K.; Cano, J.C.; Manzoni PToh, C.K. A comparison of the performance of TCP-Reno and TCP-Vegas over MANETs. In Proceedings of the 3rd International Symposium on Wireless Communication Systems, Valencia, Spain, 6–8 September 2006; pp. 495–499.
31. Sargent, R.G. Verification and validation of simulation models. In Proceedings of the IEEE Proceedings of the 2010 Winter Simulation Conference, Baltimore, MD, USA, 5–8 December 2010.