World Journal of Engineering Research and Technology
WJERT
www.wjert.org
SJIF Impact Factor: 5.218

RPD: RELIABLE PACKETS DELIVERY CONGESTION CONTROL SCHEME IN WIRELESS SENSOR NETWORK

Dr. Mukhtiar Ahmed*[1,2], Fauzia Talpur[1,3], M.Ali Soomro[2]

[1]Department of Computer Science, University Technology Malaysia (UTM).
[2]Quaid-e-Awam University of Engineering, Science and Technology Nawabshah.
[3]University of Sindh, Jamshoro, Pakistan.

Article Received on 27/05/2018 Article Revised on 17/06/2018 Article Accepted on 08/07/2018

***Corresponding Author**
Dr. Mukhtiar Ahmed
Department of Computer Science, University Technology Malaysia (UTM).

ABSTRACT

In wireless communication network when sensor nodes transfers the data in the form of packets the loss of packets occurs due to upstream traffic, limited wireless bandwidth and network congestions. In this paper a new congestion control scheme called Reliable Packets Delivery (RPD) is introduced. RPD scheme controls the congestion and recovers the loss of packets. We have used the Multipath Packets forwarding mechanism to develop the link between sensor nodes. The famous two routing protocols DSR and 802.11 MAC are adapted to route the data packets. RPD scheme recovers the loss of packets, increases the throughput, and enhances the performance. Simulation results of RPD are compared with other famous schemes like Phase Divide TCP (PTCP) and FeWTCP.

KEYWORDS: PTCP, FeWTCP, RPD, congestion control, upstream traffic.

INTRODUCTION

In wireless communication networks the congestion problem is most common due to the multi-hop routing, many-to-one communication model, small node buffer and limited link bandwidth.[1] Due to the congestion problem the most of the sensor nodes will drop the packets and in resultant the overall network throughput will be reduced and the energy level of the sensor nodes also be reduced.[2] To prolong the network life; it is important to control the congestion.[3] The traditional TCP is not suitable for the wireless sensor network as defined in;[4-6] because traditional TCP is purely designed for wired network architecture and when there is no congestion the TCP will reduce the data transmission rate due to the trigger control mechanism.[7] The trigger control mechanism will reduce the network throughput and also increase the network delay. Traditional TCP uses the end-to-end mechanism for the congestion control and when congestion occurs it will take the longer response time; so this longer response time will drop the packets.[1] TCP is connection-oriented protocol and when three-way handshake connection build; the TCP can transmit the data.

There are majority of the congestion control schemes are introduced by different researchers and still the research in this area is needed. Some famous schemes like (i). CODA as in[8] and ESRT as in[9] schemes depends on the usage of the buffer occupy ratio to detect the congestion.[10] (ii). SenTCP as in[11] depends upon service time, buffer occupy ratio and local packet interval-arrival time to detect the congestion. It uses local packet service time to detect congestion. (iii). FeWTCP as in[12] uses Fractional Window increment scheme TCP performance, which allows the TCP congestion window to grow by the "growth rate of TCP Window" packets at every round trip time. (iv). PTCP uses the increase of the TCP congestion window by different means in TCP slow start. Out of these schemes our major focus is on traditional TCP, FeWTCP and PTCP.

FewTCP congestion scheme as defined in[12] is totally depends upon the congestion upon the fractional window algorithm and is also a multi-hop low bandwidth scheme. The FeWTCP keeps the small value for the window size (α) with the long time period; so it cannot uses the network resource efficiently and in resultant the end-to-end delay will be increases and overall network throughput will be decreases.

Phase-Dive TCP Scheme as defined in[1] has also used the same FeWTCP mechanism with little bit variations and in resultant faces the same issues like FeWTCP scheme.

The proposed Reliable Packets Delivery (RPD) congestion control scheme uses to vary the congestion window with growth rate of TCP (α) and time out factor (β). The growth rate and time out factor looks the condition of the traffic and balance the load on the route and resultant the congestion remains in fully controlled.

Related work

The different congestion control schemes are defined by the researchers on basis of congestion detection, congestion notification and rate adjustment mechanism.[7] The CODA (Congestion Detection and Avoidance) uses multi-source regulation for the rate adjustment and selective back pressure for the for congestion notification.[10] The CODA schemes also use two methods for the congestion detection; one is level of buffer load and other is present and past channel load. CODA specially used the AIMD algorithm to to control the flow rate of packets.

FeWTCP congestion control scheme is designed for 802.11 multi-hop networks of low bandwidth-delay product, which allows the TCP congestion window to add α packets at every RTT.[1] The FeWTCP is based on the variation of α and the window size W. The growth rate of FeWTCP can be calculated as in Equation-1:

$$\Delta W = W^{new} - W^{current} \tag{1}$$

In FeWTCP the researchers have adapted the size of the window's value on small level with the long time period; and the drawback of this small window level is that it increase the network delay and will reduce the overall network throughput.

The researchers of the Phase-Divide TCP scheme have used the same concept of the FeWTCP with some variations. The researches have used the growth rate (ΔW**)** with the adjusted rate of window (W) and added the α/W packets to the TCP window size on every RTT. The PTCP also care about the boundary (H) for the window (W). The Equation -2 focuses:

$$\text{At every RTT} = \frac{\alpha}{W} + \frac{1-\alpha}{2H-W} \tag{2}$$

After one RTT the TCP sender updates the

$$W^{new} = W^{Current} + \frac{\propto}{W^{Current}} + \frac{1-\propto}{2H+W^{Current}} \tag{3}$$

The PTCP scheme varies the TCP window size with the boundary factor and controls the congestion with the growth rate. The PTCP scheme tested the simulation results by using the chain and grid topology.

Reliable Packets Delivery (RPD) Congestion Control Scheme

i. Proposed architecture

The proposed scheme is based on the PTCP and FeWTCP. We have used the Multipath topology structure as described in Figure 1 for the transformation of the packets from sender node to receiver node. The Multipath structure can be varying up to the **N** number of sensor nodes. The proposed architecture uses the TCP friendly equation[12] with combination of Equation 1, Equation 2 and Equation 3 to control the congestion. In Figure 1; we have used the multipath structure with source and destination nodes; source node sends the number of packets from multiple paths to the destination node. The data sending refers the hop-by-hop mechanism.

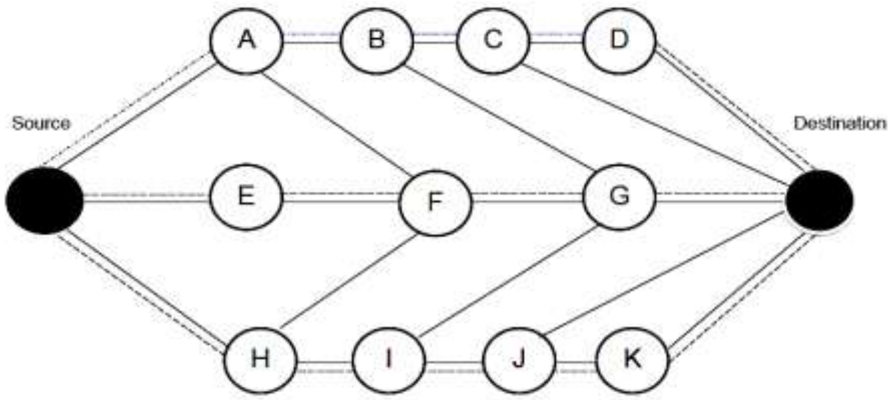


Figure 1: WSN Multipath Network Model.

ii. The RPD Routing

In RPD; when two or more than two sensor nodes are communicating between each other the two routing protocols are used, one is DSR and other is 802.11 MAC. In our simulation we consider these two routing protocols for to route the receiver to sender nodes.

a. Dynamic Source Routing Protocol (DSR)

This is specifically designed for wireless communication. This protocol allows the network to be completely self-organizing and self –configuring, without the need for any existing network infrastructure.

DSR consists on: (1) Route discovery, (2) Route maintenance, the route discovery allows discovering the route while the route maintenance will maintain the route.[13]

The DSR protocol allows multiple routes to any destination and allows each sender to select and control the routes used in routing its packets. The DSR includes easily guaranteed loop-free routing, it rapidly recovers when route in network changes.

b. 802.11 MAC (Media Access Control Protocol)

The 802.11 family uses a MAC layer known as CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance).[14]

In CSMA/CA a wireless node that wants to transfer data can perform the following tasks:

- Listen on desired channel.
- If channel is idle send the packets.
- If channel is busy node will wait until the transmission stops than a further CONTENTION period.
- If channel is still idle at the end of contention period the node transmits its packet otherwise it repeats the process as in above steps.

c. Contention Period

The contention period is a random period after every transmit on every node and statistically allows every node equal access to media.[12] To allow Tx to Rx turn around; the contention time is slotted 50 micro seconds for FH and 20 micro seconds for DS system.

iii. RPD Mathematical Model

First we have considered the TCP use friendly equation as given in Equation 4.

$$W = \frac{1}{\sqrt{\frac{2p}{3\alpha}} + Kmin\left\{1,3\sqrt{\frac{3p}{8\alpha}}\right\}p(1+32p^2)} \qquad (4)$$

Now after consideration of Equation 1, Equation 2 , Equation 3 and Equation 4; we introduce the mathematical model for our proposed scheme, that is in given in Equation 5.

$$T = \frac{W}{1.33 + \beta p(1+32p^2)} \qquad (5)$$

Where

$$W = \left(\frac{8\alpha}{3p}\right)^{\frac{1}{5}} \qquad (6)$$

T = Packets Sending Rate in bytes/sec

W = Average Congestion Window

β = Packets Gain Parameter

p = Packets Loss Rate

α = Growth rate of TCP Windows

Conditions

- When p is greater that T will be reduces.
- When α is greater than T will be greater.
- When β is greater than T will reduce.

We can control the loss of packets by varying the values of α and β, the Figure 2 and Figure 3 focuses the controllable loss of packets by sending the packets in byte/sec. The sending rate focuses the window size adjustment.

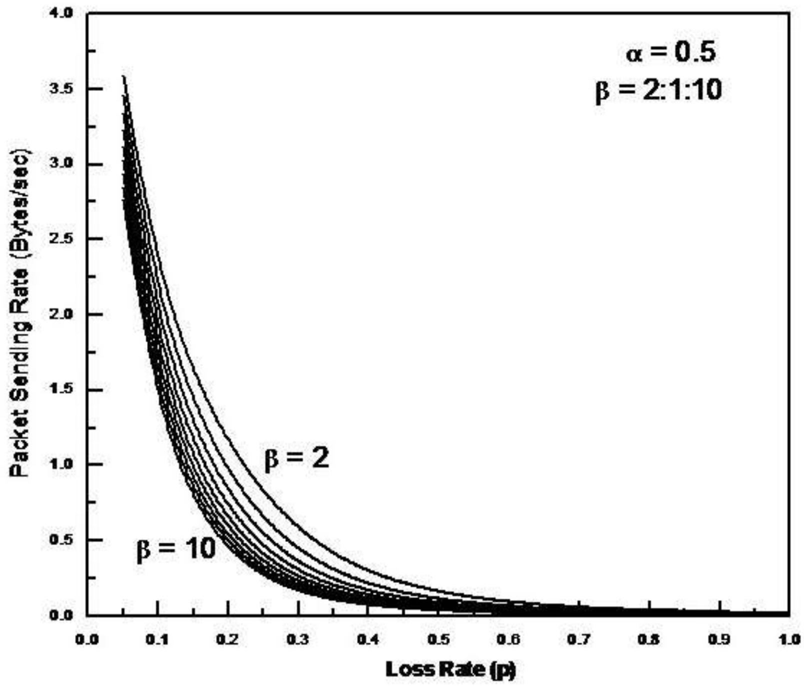


Figure 2: Loss Rate on fixed α=0.5 and β=2:1:10.

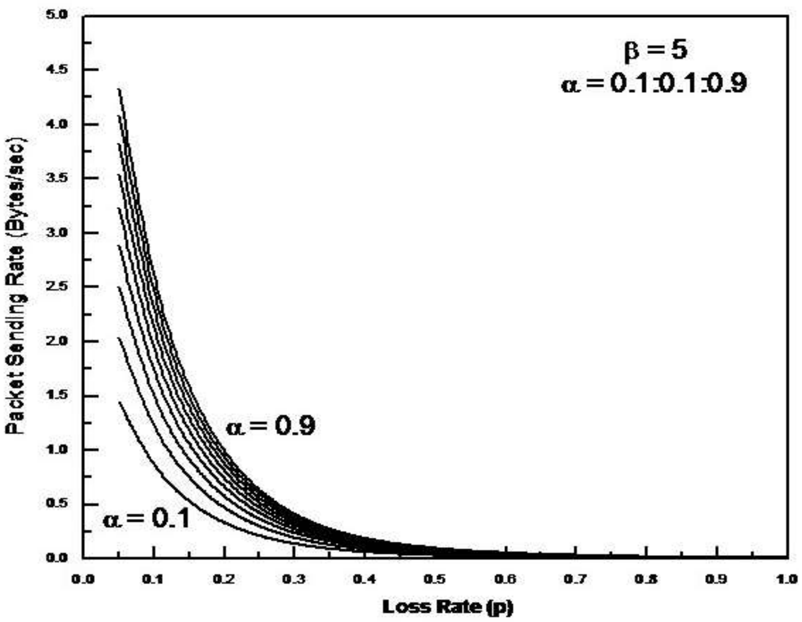


Figure 3: Loss Rate on fixed β=5 and α=0.1:0.1:0.9.

SIMULATION RESULTS AND COMPARISON

We have used the NS2 simulator for the RPD scheme and compare the results with traditional TCP, FeWTCP and PTCP. The simulation parameters are given the Table 1.

Table 1: Parameters used in NS2 for RPD (Multipath Architecture).

Parameters	Value
Data Rate	2 Mbps
Buffer Size	20 Packets
Number of Sensors	12
Radio Propagation	250 m
Carrier Sense Range	550 m
Interference Range	550 m
Initial Energy	2J
Data Packets Size	512 bits
Time Interval T	1 sec
Routing Protocols	DSR & 802.11MAC
No. of Nodes	250

i. Packets loss rate

When we vary the values of α and β the loss rate on different nodes is given in Figure 4. In Figure 4; the loss rate is measured in loss of packets per second; the loss rate of RPD is less than traditional TCP, FeWTCP and PTCP.

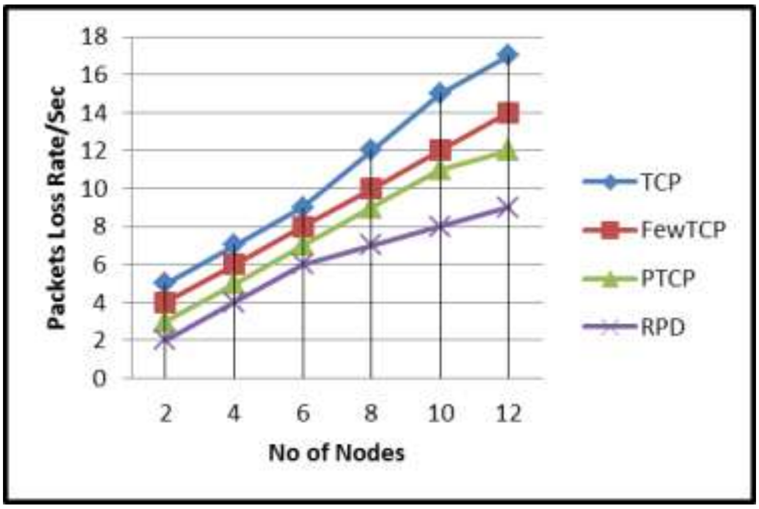


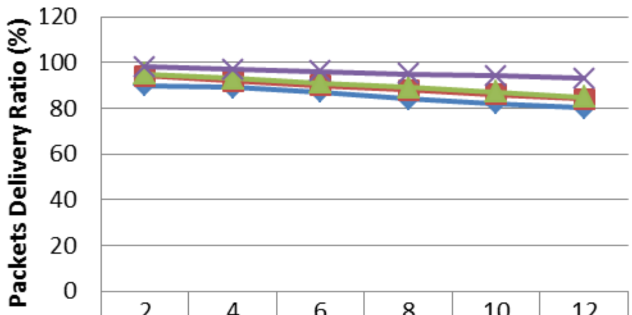
Figure 4: Packets Loss Rate in Multipath architecture.

ii. Packets Delivery Ratio

Packets delivery ratio can be measure as the ratio of total packet received by the destination node to the total of packet sent from the source node during the time of simulation. The higher the packet received the better the network performance.

$$P_{dr} = \frac{\sum P_r}{\sum P_s} \times 100 \qquad (6)$$

Equation 6; shows the Packets delivery is the ratio of packets sending over packets received. The packets delivery ratio can be measured in %. Figure 5 focuses the packets delivery ratio with respects to number of nodes. If we consider the two number of nodes are involved in packets sending than the RDP is on 98% and rest of the schemes as TCP is on 90%, FeWTCP is on 94% and PTCP is on 95% and in same way if 12 number are nodes are involved in packets sending than RDP is on 93% and rest of the schemes as TCP is on 80%, FeWTCP is on 84% and PTCP is on 85%. The results in Figure 5 shows that the RDP is more efficient in packets delivery ratio; which shows the RDP is also more efficient in congestion control.

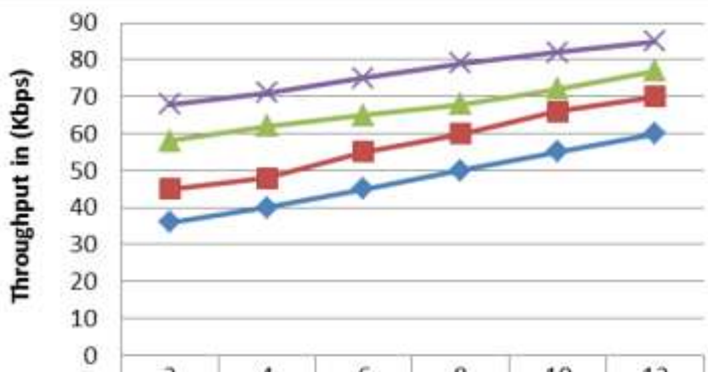


	2	4	6	8	10	12
TCP	90	89	87	84	82	80
FewTCP	94	92	90	88	86	84
PTCP	95	93	91	89	87	85
RPD	98	97	96	95	94	93

Figure 5: Packets Delivery Ratio for Multipath architecture.

iii. Network throughput

Network throughput can be measured the average of packets transferred within a time period over a communication link or channel. The higher the throughput the more efficient the network. Figure 6 shows on 2 number of nodes the throughput of RPD is 68 Kbps and rest of the schemes TCP has 36 Kbps, FeWTCP has 45 Kbps and PTCP has 58 Kbps. If we see on number of 12 nodes the RPD has 85 Kbps, TCP has 60 Kbps, FeWTCP has 70 Kbps and PTCP has 77 Kbps; it means that when the number of nodes increases the throughput of RPD is also increased in comparison of other schemes.



	2	4	6	8	10	12
TCP	36	40	45	50	55	60
FewTCP	45	48	55	60	66	70
PTCP	58	62	65	68	72	77
RPD	68	71	75	79	82	85

Figure 6: Network throughput for Multipath architecture.

CONCLUSION

The RPD Scheme gives us the good throughput than TCP, PTCP and FeWTCP Schemes, the clear difference we have seen in the simulation results.

The results are tested on Multipath Network architecture and we have considered the mathematical model as given in Equation 5; here we have varied the α and β parameters for controlling the congestion which can also controls the size of the packets sending TCP window. The NS2 based results focuses that the RPD congestion control scheme is more efficient than other schemes.

REFERENCES

1. L. Li, Y. Li, Q. Chen, and N. Nie, "PTCP: Phase-divided tcp congestion control scheme in wireless sensor networks," in *Mobile Ad-Hoc and Sensor Networks*, ed: Springer, 2007; 281-290.
2. H. Wu, Z. Feng, C. Guo, and Y. Zhang, "ICTCP: incast congestion control for TCP in data-center networks," *IEEE/ACM Transactions on Networking (TON)*, 2013; 21: 45-358.
3. C. Sergiou, P. Antoniou, and V. Vassiliou, "A comprehensive survey of congestion control protocols in wireless sensor networks," *Communications Surveys & Tutorials, IEEE*, 2014; 16: 1839-1859.
4. D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: Scalable coordination in sensor networks," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, 1999; 263-270.
5. C. Wang, K. Sohraby, B. Li, and W. Tang, "Issues of transport control protocols for wireless sensor networks," in *Proceedings of International Conference on Communications, Circuits and Systems (ICCCAS)*, 2005; 422-426.
6. C. Wang, K. Sohraby, B. Li, M. Daneshmand, and Y. Hu, "A survey of transport protocols for wireless sensor networks," *Ieee Network*, 2006; 20: 34-40.
7. A. A. Rezaee, M. H. Yaghmaee, A. M. Rahmani, and A. H. Mohajerzadeh, "HOCA: healthcare aware optimized congestion avoidance and control protocol for wireless sensor networks," *Journal of Network and Computer Applications*, 2014; 37: 216-228.
8. C.-Y. Wan, S. B. Eisenman, and A. T. Campbell, "CODA: congestion detection and avoidance in sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*, 2003; 266-279.

9. Y. Sankarasubramaniam, Ö. B. Akan, and I. F. Akyildiz, "ESRT: event-to-sink reliable transport in wireless sensor networks," in *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, 2003; 177-188.
10. M. A. Kafi, D. Djenouri, J. Ben-Othman, and N. Badache, "Congestion control protocols in wireless sensor networks: A survey," *Communications Surveys & Tutorials, IEEE,* 2014; 16: 1369-1390.
11. C. Wang, K. Sohraby, and B. Li, "SenTCP: A hop-by-hop congestion control protocol for wireless sensor networks," in *IEEE INFOCOM*, 2005; 107-114.
12. K. Nahm, A. Helmy, and C.-C. Jay Kuo, "TCP over multihop 802.11 networks: issues and performance enhancement," in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, 2005; 277-287.
13. J. Chen, M. Zhou, D. Li, and T. Sun, "A priority based dynamic adaptive routing protocol for wireless sensor networks," in *Intelligent Networks and Intelligent Systems, 2008. ICINIS'08. First International Conference on*, 2008; 160-164.
14. N. Dukkipati, T. Refice, Y. Cheng, J. Chu, T. Herbert, A. Agarwal, *et al.*, "An argument for increasing TCP's initial congestion window," *Computer Communication Review,* 2010; 40: 26-33.