A survey on efficient approach for congestion control commencing TCP pro multi hop wireless network

Dipti M. Jawalkar, Dr. M.V. Sarode
M.E(Final Year Comp Sci Engg), J.C.O.E. T. Yavatmal, diptifriends06@gmail.com, +91 7709025727
H.O.D (Comp Engg), J.C.O.E. T. Yavatmal, parthmilindsarode@rediffmail.com

Abstract: TCP performs weakly in multi hop wireless networks and even inferior if end-to-end connectivity is often broken such as in challenged networks. Lots of research has been carried out but this problem has not been solved completely. Recently, hop-by-hop congestion control originally proposed for wired networks has been applied for multi hop wireless networks to considerably improve performance. The work deals with moving congestion sort out down to lower layers is essential to overcome TCP problems in multi hop wireless networks and in this case, it is essential to further decouple congestion control from TCP. Such TCP only retains reliability control and is called semi-TCP henceforth. By way of hop-by-hop congestion control, the congestion manage efficiency of semi-TCP will not rely on the accessibility of end-to-end connectivity, that makes semi-TCP more appropriate than TCP for challenged networks. Besides performance improvement, semi-TCP may further decrease overall system complexity by removing unnecessary congestion control and using simple congestion control rather than TCP congestion window. This paper discusses such a semi-TCP using a hop-by-hop congestion control that only slightly modifies the RTS/CTS protocol used in the IEEE 802.11 DCF. In addition, an elucidation to a deadlock problem in the RTS/CTS- based hop-by-hop congestion manage is furthermore deliberated.
Keywords – semi-TCP, RTS/CTS

1. INTRODUCTION

With increase in belief on networks similar to the Internet, there is an augment in conflict for the network's resources. This argument has affected the performance of networks. While several network performs plausibly well below light load, problems surface while they are used expansively. The most distinguished and ordinary complexity that networks are faced with is failure of data. While data failure in network happens due to a multiplicity of reasons, congestion within the network is the chiefly common reason. Insecurely speaking, congestion refers to the failure of network performance while a network is intensely loaded [1]. This failure of performance can be data loss, bulky delays in data communication, this is frequently unacceptable. Owing to this, scheming and avoiding congestion is a decisive complexity in network management and plan.
The most accustomed category of computer network is a packet-switched network, where nodes send data contained in the form of packets to every other. The most common loom used to transmit data is store-and-forward. Every node waits till it has customary a whole packet prior to forwarding it at a later instance to the suitable output link. The Internet is an illustration of a network that is radically packet-switched. The data route commencing a source to destination is computed by dissimilar methods by routers. While discussion about congestion control it is obligatory to talk with reference to organize of data packets in these routers. Congestion occurs in router when the escalating bandwidth of arriving packets, predestined for a scrupulous output link, exceeds the link's bandwidth. A paradigm of congestion occurring at a router is given in Figure 1.

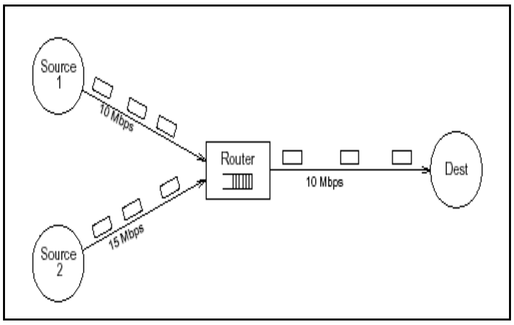

Fig 1: Small congested network.

1.1 Congestion organize in TCP/IP :

TCP/IP, the remote and wide used protocol suite in the Internet uses a window-based flow control mechanism. At this juncture both the sender and the receiver preserve a window for the purpose of session. The sender and receiver concur on an initial window size at the concern of a session.

There is an supplementary window that is used. It is the congestion window, cwnd, and the effectual window size swnd is the negligible of the window size advertised by the receiver (rwnd) and the congestion window.

swnd = min (cwnd , rwnd)

The TCP/IP congestion manage is principally done using the cwnd window. The effectual window that the sender uses to send data is the swnd. The source is tolerable to send the window size of data devoid for the future for an acknowledgement (ACK) from the receiver. The swnd shrinks by the quantity of bytes sent. Every packet that the source sends is exclusively acknowledged by a packet number. When the receiver obtain a packet it sends an ACK packet back to the source, thus symptomatic of the receipt of all the packets previous to and the next packet that is expected. Depending on the numeral of packets acknowledged, the sender window swnd is enlarged again. An ACK packet also obscure that the network can hold the figure of packets sent by the sender. This results in a linear boost in the cwnd windows size. As soon as a time-out occurs or a triplicate ACK is established, it is interpreted as a consequence of congestion. Owing to this, the congestion window shrink (to a fraction, usually ½). This is the Additive raise and Multiplicative shrink mechanism of congestion window, through which end-to-end congestion manage is implemented. This particular dissimilarity of TCP/IP implementation is recognized as TCP Reno [3].

1.2 TCP in Wireless Multi-hop Networks

In a multi-hop wireless network packets are delivered from a source to a destination using packet forwarding capabilities of transitional wireless nodes. An average user send emails, get news, do shopping, search for information and gaze at video or listen to music. All of the above activities use the TCP protocol in some form. Current measurements have also shown that around 90% of the Internet transfer is TCP traffic [4]. TCP recital in wireless multi-hop networks depends on the uniqueness of the network e.g. if the nodes are mobile or immobile

.

2. LITERATURE REVIEW

Extensive study has been done to understand the short coming and to improve the performance of TCP in wireless network. The work on improving TCP performance in wireless networks alert on distinguishing between packet loss due to wireless fraud from loss due to congestion, in the context of wireless wide-area networks. Another class of work concentrates on improving TCP's throughput by freezing TCP's congestion control algorithm during link crash induced losses, especially throughout route changes.

In [3] Balakrishnan et al. proposed the aim and implementation of a simple protocol called "Snoop" for the circumstances where a fixed host is communicating with a transportable host with the help of a base station. When the Snoop agent residing at the base station receives a replacement acknowledgement against a lost packet at the mobile host, it retransmits the absent packet locally to the mobile host and conceals the packet loss actions from the sender and hence prevents it from dipping its congestion window to maintain a good throughput.

TCP New Reno [4], distinct by RFC6582 improves retransmission throughout the fast-recovery point of TCP Reno. During fast revitalization for each replica ACK so as to is returned to TCP New Reno, a novel unsent packet commencing the conclusion of the congestion window is sent, in the direction of the broadcast window filled. For every ACK that makes imperfect progress in the sequence space, the sender presuppose that the ACK points to a fresh hole, and the next packet in advance of the acknowledged sequence number is sent. Since the timeout timer is reset every time there is expansion in the transmit buffer, this allows New Reno to fill up enormous holes, or numerous holes, in the sequence space. Since New Reno can initiate new packets at the end of the congestion window throughout fast recovery, high throughput is maintained all through the hole-filling process, even while there are numerous holes, of multiple packets each. When TCP enters speedy recovery it records the utmost exceptional unacknowledged packet sequence number. When this series number is acknowledged, TCP proceeds to the congestion avoidance state. A problem occurs with New Reno when close by no packet losses but instead, packets are reordered via additional than 3 packet series numbers. When this happens, New Reno erroneously enters quick revival, but as soon as the reordered packet is delivered, ACK sequence-number advancement occurs and commencing there until the end of fast recovery, every bit of sequence-number steps ahead producing a replica and unneeded retransmission that is right away acknowledged.

P. Senthil [5] focuses on avoiding packet loss previous to incast congestion, which is additional tempting than recovery after failure. An elucidation that modifies only the TCP receiver .Their effort deals to accomplish incast congestion avoidance at the receiver side by means of preventing incast congestion. The receiver side is a expected inclination since it knows the throughput of all TCP links and the available bandwidth. The receiver side can modify the receive window size of each TCP connection, so the collective burst of all the harmonized senders are kept below control. This completed to recommend Frame relay congestion control algorithm (FRCCA) congestion Control meant for frame relay transmission. The congestion control on the receiver side is performed as follows: It formulate use of the accessible bandwidth on the network interface as a allocation to harmonize the receive window boost of all inward bound connections, the per-flow congestion manage is performed independently of the slotted instance of the Round-Trip Time (RTT) of every connection and the receive window modification is based on the proportion of the difference stuck flanked by the precise and expected throughput in excess of the estimated one.

3. PROBLEM IN PRESENTED STRUCTURE

The vital crisis in hop-by-hop congestion manage is its obscurity in view of the fact that it requires the network to dwell in the congestion control process. To execute the hop-by-hop congestion manage scheme based on extensively used request-to-send/clear-to-send (RTS/CTS) protocol. The suggestion is straightforward to comprehend since barely media access control (MAC) layer is apprehensive and hence by taking benefit of the propagation nature of wireless media to relieve hop-by-hop congestion control. In the protocol plan for hop-by-hop clogging manage, the deadlock problem which prevents congestion situations is to be proficiently unrestricted. An algorithm is then planned to fully concord with the deadlock problem.

3.1 Routing failures

In steady wired networks, route failures ensue to be very rarely. Nevertheless, in multi-hop wireless networks they are the convention quite than the exception [4]. The node mobility is the foremost source of frequent topology changes and route failures in wireless networks. Moreover, the link failures owing to the argument on the wireless channel may guide to route failures in cooperation static and mobile ad hoc networks. When a route failure occurs, packets that are buffered at middle nodes down the route will be dropped. This large magnitude of packet drops may cause a sequence of time-outs at the TCP sender. As a consequence, the RTO price will be doubled for each succeeding time-out. Furthermore, TCP will misguidedly construe the loss as an indication of network congestion with set off the congestion control mechanisms to decrease the size of cwnd and ss thresh. In addition, TCP does not encompass every sign of the route re-establishment duration, because the route re-establishment occurrence following route failures depends on the essential routing protocol. These proceedings have three poor effects:

i. The small cwnd and ssthresh values decrease the original sending rate subsequent to the route is restored. Therefore, it takes an extensive time for the sending speed to catch-up to a high value subsequent to a new route is found.
ii. The bulky RTO value reduces the receptiveness of TCP; even if the route is restored, TCP will obtain an extensive time to converge to the right level of operation.
iii. The bulky idle time of TCP caused by route re-establishment will degrade the throughput.

3.2 Misjudgment on congestion status

In wireless networks especially multi-hop ad hoc networks, many novel issues take place for TCP. One difficulty is that TCP cannot differentiate between congestive losses and other losses caused via channel unreliability adding together to not healing mobility [6]. This difficulty reason the TCP source to needlessly decrease its congestion control window single time a retransmission timeout (RTO) occurs, results in stumpy network throughput. Besides, lost and delayed ACKs in the reverse route may well also cause the source not to obtain ACKs in time, which is as well regarded as congestive fatalities on the forward route by the source node. A further type of misjudgment is that yet if the initial result on congestion status in a route is accurate, this judgment may well happen to invalid due to immediate changes in the route. This change may possibly be caused by moreover a routing protocol that cannot toughen a path or terminal mobility which changes radio links intermittently. If the original route is indistinct, every one effort for congestion control along this route no longer makes universal sense in view of the actuality that the original congested node may not be part of the present route due to mobility, which may well also cause a congested link to turn hooked on component of an creative congestion-free route. The above scrape is mainly due to the fact that TCP does not encompass adequate information on the network condition for congestion manages. Thus, decoupling the congestion manage from TCP and moving along this task to lower layers can stay away from these problems since the lower layers can be acquainted with instantaneously what happens in the network.

3.3 Hop-by-hop congestion control in wireless networks

The hop-by-hop congestion control is even while more proficient than the end-to-end control and appropriate for multi-hop mobile ad hoc networks (MANETs), its implementation intricacy is high owing to per-node involvement in congestion control. On the other dispense with the MAC implemented in shared-media wireless networks, every node desires to detect activities of other nodes and still to interrelate with each other. For that reason, some mechanisms for information incarcerate and swap over between neighbors have been previously implemented in wireless networks. In this case, it is reasonably unproblematic to implement a hop by-hop congestion control with associated means with no big raise in implementation complexity. Basically, the RTS/CTS protocol necessitates a node to send an RTS first to the receiver and it will hurl back CTS if it is clear to receive. It is not complex to discover that this RTS/CTS swap over can be slightly customized by including congestion information for hop-by-hop congestion manage. Actually, this is currently the basic idea following the hop-by-hop congestion manage. There is also some extra hop-by-hop congestion manage schemes at the data link layer such as which changes MAC parameters such as CWmin and CWmax of IEEE 802.11 to bear congestion manage information. In [7, 8], an inherent hop-by-hop congestion manage is discussed, by which the information on congestion category and manage is obtained all the way throughout observing transmission activities of its neighboring nodes quite than explicit information swap.

4. PROPOSED STRUCTURE

To perk up the performance of TCP in multi-hop wireless networks it desires to regroup these approaches according to the policy used in order to improve TCP performance. However, a little approach can be harmonized to quite a lot of types of strategies but are just classified to their chief strategy.

4.1 A semi-TCP based on RTS/CTS protocol

This segment discusses the protocol plan of semi-TCP within IEEE 802.11 multi hop wireless networks. The two features of the structure are intra-node and inter-node congestion manages [8]. In intra node congestion manages; the upper layer in a wireless node precinct the delivery of data packets to the lower layer according to the congestion situation in the lower queue. In inter-node congestion manage, the neighboring nodes assist to release the congestion.

The subsequent notations are used in the discussion.

- L: buffer capability.
- ℵ: queue length, i.e., the tenancy of the buffer.
- Tc: common congestion threshold ($Tc \leq L$).
- m: buffer spaces reserved in favor of transient traffic.
- k: the number of packets the congested node transmits prior to it is considered not congested.
- g: buffer spaces kept to avoid the deadlock situation.

4.2 Intra-node congestion manage

The semi-TCP does not make use of the congestion window of TCP to sort the number of segments injected hooked on the network. Instead, the quantity of segments to be transmitted is rigid by the congestion status of the buffer at the lower layer, particularly the MAC sub-layer. In broad, the buffer is regarded open of congestion if the subsequent condition is pleased:

$$\aleph < Tc \quad (1)$$

At this point, the congestion is a logical condition comparatively than a physical congestion condition in which the intact buffer has been engaged. In a multi-hop MANET, a node can be communally a traffic source and a router simultaneously. Under heavy traffic load, the source traffic of a node may perhaps overlook its buffer, causing momentary traffic from other nodes to include smaller quantity or even no opportunity to exploit the buffer. Therefore, some buffer spaces require being aloof for transient traffic, which is admitted if

$$\aleph < Tc + m. \quad (2)$$

Another important utility of TCP is the end-to-end uniformity manage, through which each unacknowledged segment is retransmitted by the source node just the once the RTO is due, awaiting the segment is absolutely acknowledged. Only this part is set aside in semi-TCP. With TCP, replica ACKs is sent by the destination for the nippy retransmission of out-of-order segments and the fast recovery of congestion widow. In view of the fact that by way of semi-TCP, the congestion window is no longer used so that no duplicate ACKs are requisite to send in order to decline the traffic load on the reverse route sustaining to improve the performance.

4.3 Inter-node congestion manage based on RTS/CTS

By means of the inter-node congestion manage; the congestion state in the region will be implicitly fed reverse to the source node such that the sending rate of the source node will be throttled. In multi-hop wireless networks to apply the inter-node congestion manage scheme based on IEEE 802.11 RTS/CTS protocol here is need to establish two subtypes of RTS and CTS to hold the congestion information: request-to-send-congested (RTSC) specify that the sender of this RTSC is congested clear-to-send-congested (CTSC) specify to the sender of this CTSC is congested. These two sub type scrutinized and implemented by setting the idle bits in the unique RTS and CTS frames.

In the subsequent, the method of the hop-by-hop congestion manage algorithm is based on the extensively used RTS/CTS protocol. For the effortlessness of presentation the two nodes occupied are denoted as node A and node B. Suppose node A first intellects the inactive channel, and it has a packet for node B. According to the condition of node A and node B which are within, all plausible combinations of RTS(C) and CTS(C) used by this semi-TCP implementation are depicted in algorithm given in Figure 2.

Algorithm: Node A has a packet for node B
1: **if** Node B is congested **then**
2: Defer transmission
3: **else**
4: **if** Node A is congested **then**
5: Send RTSC
6: **else**
7: **if** there is no neighboring node congested **then**
8: Send RTS
9: **else**
10: Defer transmission

Fig 2. Algorithm for Inter-node congestion manage based on RTS/CTS

When node A have a frame to send to node B, node A needs to first aspirate the congestion status of its neighboring nodes. Node A perceives the congestion condition of the neighboring nodes (e.g., node B) by taking notice of the channel. Whenever an RTSC or a CTSC frame transmitted by node B is realistic, node B is regarded as congestion by node A, and a timer is started for node B. When node B has transmitted k packets or the timer epoch out, node A will rationalize that the congestion in node B has been released. For the nodes overhearing whichever an RTS or a CTS, they just trail the network allocation vector conceded by the RTS and CTS to settle on their behaviors, which are the alike as defined in the IEEE 802.11 DCF.

4.4 Deadlock situation

If A returns nCTS to the sender of an RTS B, it tells B that it is suffering from congestion and not capable to receive any data. Though, this will cause a deadlock situation as illustrated in Fig. 3(a). To liberate the congestion in A, A has to send out data in its buffer if packet dropping is adverse. But if the receiver of the data at the head of line (HOL) in A's buffer is now B, which however is also congested with its HOL packet just intended to A. In this case, if B also returns an nCTS to A ahead receiving an RTS from A, then the congestion in together A and B cannot be released unless dropping packets [9]. As illustrated in Fig. 3(b), this deadlock situation can be avoided by reserving n buffer spaces to receive packets from those congested nodes (e.g., A here). With this reluctance, if an RTS is sent by a node that just sent out an nCTS (i.e., A), one of these reticent buffer spaces can be used to obtain the packet from A. In this case, a CTS should be sent out by B.

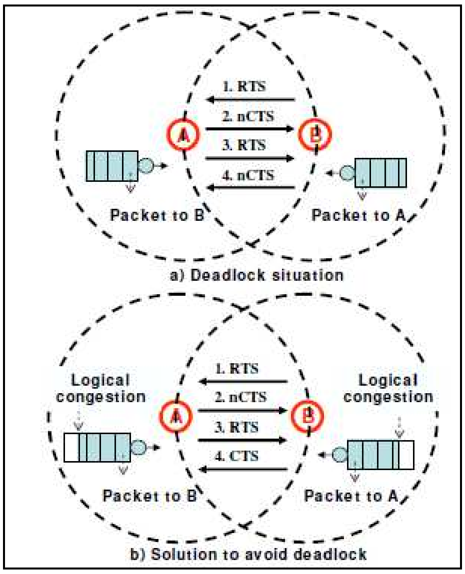

Fig 3.Deadlock situation and a solution to overcome from it.

5. BENEFITS OF PROPOSED STRUCTURE:

TCP is one of the finest protocols of the internet protocol set. owing to this we are used it hooked on multi hop wireless network for controlling non-congestion loss. throughout our proposed structure, if there is a congestion at that time there may be losses, but if there is non-congestion in network than our system will be very much competent for finding such losses and these losses are entirely dependent on following parameters :

- It is used for the consistent transmission of the data
- since the packets rescue of this protocol is secure.

- Transmission Manage Protocol is the vertebrae of the today's online effort since the TCP losses owing to non-congestion loss, it is un-useful mania for us
- because if any packet drop in any means the whole process is repeated for the packet retransmission, this
- state is very frantic at the time of link failure loss,
- time out & throughput.
- If some packet is drop or reordered in wrong way they are not in prearranged way so our research is constructive for this dilemma in both way in congestion loss or non-congestion loss.
- If we utilize our research result for TCP it decreased the dropped packets so it is favorable for us in both way such as pecuniary and social life it helps us to delivery quick data delivery in reliable way.
- If we hoard time and reduce the packets dropped ratio the recital of our system is increased and it helps us to diminish the cost of our system.
- If the packets do not send more than once time than it works fast and gives efficient results.

6.LIMITATION

This paper is based on our study as for every above assessment between presented & proposed structure we will simply boost the competence of our structure as match up to to existing or old structure. We have not implemented our innovative research, and we will be implementing this on network simulator i.e. NS2 for simulating our research. subsequent to appropriate simulation the authentic results will come.

7. CONCLUSION AND FUTURE SCOPE

Nowadays everyone are using internet for reduction of time to several of the actions & attain a reliable and efficient outcome for all such activity, with this scenario we are just trying to curtail the network load which is moreover data retransmission or data replication, due to which network resolve performed slow process and unnecessary data replication. Through our anticipated system network will be performed well & end to end reliable delivery of our packet.

This paper is study about analysis of congestion manage techniques. To implements one of the system appropriately with proper infrastructure then it helps to improve the performance of TCP. The idea is to propose a novel framework which is termed as semi-TCP. It considers the effectiveness of congestion manage and the utility of a transmission control protocol. This would turn into an efficient methodology in which semi-TCP, solve TCP's troubles in multi hop ad hoc networks. The study includes scrutiny of congestion control techniques in terms of the major performance indicators for comparison it includes throughput, delay, dropping ratio, and path length.

REFERENCES

[1] Gaifan LI, Yinlong XU, "A TCP-friendly Congestion Control Scheme for Multicast with Network Coding" Journal of Computational Information Systems 9: 21 (2013) 8541–8548

[2] Zhenghua Fu, HaiyunLuo, PetrosZerfos, Songwu Lu, Lixia Zhang, Mario Gerla "The Impact of Multi hop Wireless Channel on TCP Performance", IEEE Transactions on Mobile Computing, 2005

[3] Ahmed Khurshid Md. HumayunKabir, and Md. AnindyaTahsinProdhan "An Improved TCP Congestion Control Algorithm for Wireless Networks" IEEE ©2011 978-1-4577-0253-2/11

[4] Ammar Mohammed Al-Jubari, Mohamed Othman, BorhanuddinMohd Ali and Nor AsilahWati Abdul Hamid"TCP performance in multi-hop wireless ad hocNetworks: challenges and solution", EURASIP Journal on Wireless Communications and Networking 2011, 2011:198

[5] A. Mohan & Dr. P. Senthil Kumar," Congestion Avoidance using Frame Relay Architecture" The SIJ Transactions on Computer Networks & Communication Engineering (CNCE), Vol. 1, No. 1, March-April 2013

[6] SumitRangwala, Apoorva Jindal, Member, Ieee, Ki-Young Jang, KonstantinosPsounis, Senior Member, IEEE, And Ramesh Govindan "Neighborhood-centric congestion control for Multi-hop wireless mesh networks" IEEE/ACM Transactions On Networking.

[7] Marcos Talau and Emilio Carlos Gomes Wille, "Available Network Bandwidth Schema To Improve Performance In TCP Protocols" International Journal of Computer Networks & Communications (IJCNC) Vol.5, No.5, September 2013.

[8] YeguiCai, Shengming Jiang, Quansheng Guan and F Richard Yu "Decoupling congestion control from TCP (semi-TCP) for multi-hop wireless networks" Caiet al. EURASIP Journal onWireless Communications and Networking 2013, 2013:149.

[9] Shengming Jiang, Qin Zuo, Gang Wei "Decoupling Congestion Control from TCP for Multi-hop Wireless Networks: Semi-TCPffi" Shengming Jiang, Qin Zuo, Gang Wei CHANTS'09, September 25, 2009, Beijing, China.Copyright 2009 ACM 978-1-60558-741-7/09/09