A Novel Approach for Congestion Control in MANET
Robin Choudhary, Niraj Singhal
M.Tech Scholar, Shobhit University Meerut, Uttar Pradesh, India
M.Tech Scholar, Shobhit University Meerut, Uttar Pradesh, India

Abstract: In mobile ADHOC networks have limited bandwidth and are more prone to error than wired networks which further impose limits on the amount of data that can be sent. In order to conserve the limited resources, it is highly desirable that transmission should be as efficient as possible with minimal loss. The objective of congestion control is to limit the delay and buffer overflow caused by network congestion and provide better performance of the network. The traditional congestion control mechanism, applied by the transport control protocol is unable to catch up the network dynamics of ad hoc networks. Congestion control assumes all losses induced by congestion. In this paper, a novel approach of congestion control for supporting applications like multimedia streaming over MANET is being proposed.

Index term: Link failure, Velocity Change, Position Congestion Control, Non-congestion Loss.

1. INTRODUCTION
A mobile Adhoc network is composed of a group of mobile computing devices (nodes) that are equipped with wireless-LAN capability. In contrast to infrastructure wireless network, which uses base station to manage nodes in its area, MANET does not require any fixed infrastructure. Nodes in multi hop MANET help each other to forward packets from hop to hop such that two nodes that cannot hear each other can transmit data to each other. In this way, the connectivity of a MANET is greatly enhanced. In expensive deployment of MANET due to absence of fixed infrastructure as well as mobility feature for all nodes have considered MANET as a subject of research. In a MANET environment, communication links are unstable due to various reasons such as interference of radio signal, radio channel contention, mobility of the nodes and battery depletion. The wireless network have limited bandwidth and are more prone to error than wired networks which further impose limits on the amount of data that can be sent. Hence, in order to conserve the limited resources, it is highly desirable that transmission should be as efficient as possible (minimal loss and transmission). The main objective of congestion control is to limit the delay and buffer overflow caused by network congestion and provide better performance of the network [1]. In wire line networks, congestion control is implemented at the transport layer and is often designed separately from functions of other layers. Since wired links have fixed capacities and are independent, this methodology is well justified and has been extensively studied [18]. However, these results do not apply directly to Adhoc networks because the ad hoc networks result in large amount of packet loss, high delay, unfair scenarios and low throughputs. In Adhoc networks, each mobile node has limited transmission capacity and buffer and they mostly intercommunicate by multi-hop relay [1]. The random behavior of Adhoc networks cause the topology of wireless network to be changed rapidly and unpredictably. Moreover, node’s mobility puts an extra burden on TCP’s congestion control mechanism. As a result, traditional congestion control mechanism, applied by the Transport Control Protocol (TCP) [8], is unable to catch up the network dynamics of ad hoc networks. Congestion control is the most controversial part of TCP which degrades performance when encounters non-congestion loss in MANET. Congestion control assumes all losses induced by congestion. For example, link breakage lasts greater than Retransmission Time Out (RTO) is mis-interpreted as congestion loss. Thus regardless of kind of loss, it decreases sending rate to alleviate congestion and grows retransmission timeout exponentially to wait more for receiving acknowledgment. It is plausible in wired network since non-congestion loss occurs rarely and some application can tolerate some degrees of error. However, this unnecessary throughput drop that waste available resources such as bandwidth arises in MANET. Link failure needs TCP to explore how much new route is congested in comparison to the broken one. Traffic characteristics can affect queuing delay and processing delay of intermediate nodes that consequently influences Round Trip Time (RTT). If discovered route suffers heavier traffic than old one, retransmission timer must wait more to receive acknowledgment and RTO should be increased. Otherwise, when new route is approximately non-congested, data packets and acknowledgment transferred quicker than old route. Thus sender must wait less than before to receive acknowledgment and RTO should decrease. In this paper, a novel approach of congestion control is being proposed, for supporting applications like multimedia streaming over MANET. The following subsections give the brief idea about the problem, proposed solution and an outline of this paper respectively. The remainder of the paper is structured as follows: Section II introduces the related works and background in the area of mobile ad hoc networks, congestion control and classify the type of loses. Section III illustrates the motivation behind the
new proposal and proposed solution to classify the type of losses and alleviate the packet losses over mobile Ad hoc networks. Section IV concludes this paper with possible future research directions.

A) Problem Statement

- In this section, we have tried to point out the problems of using TCP congestion control mechanism for classifying the type of losses to control in mobile ad hoc networks.
- Standard congestion control cannot detect link failure losses which occur due to mobility and power Scarcity in multi-hop Ad-Hoc network (MANET).
- Congestion control is the most controversial parts of TCP which degrades performance when encounters non-congestion loss in MANET. Congestion control assumes all loss induced by congestion.

II. RELATED WORK

Sender should recognize state of MANET and wireless link to act accordingly. For example, specifying available buffer of intermediate nodes that assess congestion can greatly influence recovery operations. Measuring remained energy of nodes can assists sender to change route before link breakage. Calculating distance between nodes based on signal strength can help sender to predicate future link failure and switch into another route before breakage. All these information can be either measured explicitly with support from intermediate nodes or estimated implicitly from information in received acknowledgment. In First mechanism, TCP sender entirely does the job and estimate MANET situation implicitly without any support from intermediate node. It does not create processing overhead at intermediate routers. The main drawback of it is the lack of detailed information about state of wireless link at the sender [16]. For example, Fixed RTO interprets two successive timeout as route failure. Then retransmit unacknowledged packet while it keeps value of RTO unchanged [9]. However two successive timeout can be sign of congestion in congested MANET and is highly based on existing traffic pattern. That’s why it is not precise enough. In feedback (cross layers) approaches, sender get detailed information from network state by collaborating between TCP layers of intermediate nodes. For example, since congestion control is not aware from losses due to wireless medium contention over 802 MAC protocol, it must collaborate with MAC layer to address these losses. Although Feedback methods are more precise than end-to-end approach [13], modifications in intermediate nodes make implementation complicated for WAN. Moreover, extra overhead produced due to transmission notification packet. In addition, it reduces flexibility [5]. For example, TCP Muzha forces Intermediate nodes to fill special field in acknowledgment header to clarify sender how many empty rooms are available in their buffers [19]. TCP-F [12] and TCP-BUS [3] are feedback approaches that pursue the same mechanism. When intermediate node detects link breakage, route notification message informs source to stop sending further packets and freeze state variable such as RTO. When route rebuilt, route reconstruction notification packet informs source to resume transmission with old RTO. Westwood VT is an end-to-end approach, which classifies packet loss by estimating existing data packet in buffer of intermediate nodes. It is too resemble to TCP-Veno [2]. Actually both inherit policy of TCP Vegas to differentiate causes of packet loss [6]. After received acknowledgment, they measure the difference between expected rate and actual rate and assign it to Δ which is indication of amount of buffer in queue of middle nodes. Interpreting causes of loss is done based on two predefined threshold α and β and available buffer of intermediate nodes as Δ. If it becomes smaller than α, buffers of intermediate nodes still can accommodate incoming packets. So WestwoodVT relates any loss due to the wireless error. If Δ is larger than β, it shows that buffers are approximately full and any packet loss is due to congestion [6]. If estimated Δ becomes between two thresholds, decision is postponed to next losses. Main drawback of WestwoodVT that degrades performances (throughput and energy consumption) is revealed when Bit Error Rates (BER) increases [17]. In addition, WestwoodVT cannot address link failure. TCP-Feno introduces another challenge on TCP VEGAS proponents (WestwoodVT and TCP-Veno). It claims TCP VEGAS performance degrades in network when nodes use small buffer size [7]. MANET with nodes carrying small buffer size can quickly enter into congestion mode. However, since TCP VEGAS does not contribute maximum buffer size in estimation, it just compares Δ with threshold α and find out it is less than it. So TCP VEGAS declares loss as non-congestion loss while congestion exists. However, TCP-Feno still cannot cope with two first mentioned problems.

LDA_RQ is an implicit end-to-end approach which tries to estimate queue usage rate of intermediate nodes. It does not need any support or feedback from middle nodes. Available Information in transport layer is congestion window size (cwnd), round trip time (RTT) [14]. It defines two loss classification formulas, one for beginning and another for rest of transmission. It compares first classification metric with threshold until maximum EROTT exceeds three times greater than minimum EROTT. After this gap appeared, it uses second classification until the end. In addition, special ROTT called congestion ROTT calculated to show border between normal and congested MANET. When TCP recognizes loss through third duplicate acknowledgements, it verifies weather queue usage exceeds 50% or current ROTT becomes greater than
congestion ROTT. If either former or latter satisfied, detected loss is due to congestion. Otherwise, loss is induced by non-congestion factors [14]. However, it cannot detect link failure. In addition, gap between minimum and maximum of EROTT achieved experimentally that definitely varies based on experiment. Moreover, in situation which gap cannot reach to three queue usage remains less than 30% and congestion EROTT might not be initialized. TCP-WELCOME is an implicit end-to-end scheme, which differentiates causes of packet loss based on history of Round Trip Time. Ascending growth of RTT increment induced by congestion. However, If RTT didn’t fluctuate and remained around averaged value, the way packet loss recognized becomes important. Three duplicate acknowledgements are a consequence of wireless channel error while retransmission timeout is due to link failure [17]. However, TCP-WELCOME uses RTT, which includes both delays of forward and reverse path while only delay of forward path must be considered. In addition, it offers recovery method based on RTT comparison. TCP-Welcome claimed that RTO adjustment should be done based on the Capabilities of discovered route such as length, load and link quality. After link breakage, total delay for new route varies from broken route. Hence, RTT comparison seems to be suitable Parameter for tuning

\[
\frac{\text{RTO}_{\text{new}}}{\text{RTO}_{\text{old}}} = \frac{\text{RTT}_{\text{new}}}{\text{RTT}_{\text{old}}}
\]  

However, RTT is not enough for depicting capabilities of discovered route. In addition, it includes both delay of forward and backward path. ABRA does not offer method to classify packet losses. However, it uses smoothed Round Trip Time instead of RTT to set RTO after link breakage. When link failure lasts more time than RTO, timeout happens. Standard TCP grows RTO exponentially due to multiple successive back-offs. When route come back, TCP cannot retransmit last unacknowledged packet since it must wait until this long RTO expires. Thus, it is serious deficiency since route recovered but TCP remains idle unnecessarily. ABRA claims that new RTO is dependent on the smooth round trip time (SRTT) [4]. In end-to-end threshold-based algorithm, this enhances congestion control to address link failure loss in MANET. It consists of two parts. Threshold-based loss classification algorithm uses queue usage to classify network state periodically into congestion or non-congestion mode. Any retransmission timeout in period which MANET is non-congested mode is an indication of link failure loss. In addition, implementation showed that small percentage of three duplicate acknowledgments which emerge immediately after route recovery might be result of route changes. After detecting losses due to link failure, it should adjust RTO for reconstructed route by comparing its capabilities with broken route using available information in transport layer. This enhances congestion control by transmitting packet as soon as route recovered rather than being idle unnecessarily. when we send to destination the ack packet travels from source to destination through the intermediate nodes but the calculation is performed only at the source node. In case we want to find out the losses at intermediate nodes then the overhead of calculation will increase as then we have to send ack Packet at each intermediate node and thus calculation will be performed at each node respectively.

III. PROPOSED WORK

Existing solution in loss classification and control the congestion areas are based on the network Parameter such as RTT, RTO, Bandwidth and number of nodes are used for communication between sources to destination in MANET. These techniques are totally depending on the receiving ack and sending ack for each receiving packet. The most existing technique does not getting differentiate between the link of failure with other type of failure. This paper studies problem deeply by determining whether link failure loss occurrence or not. The link failure and network partitioning which mainly created by a failure such as mobility and battery depletion has negative on MANET. To solve this problem a novel approach, which uses velocity change and change angle in spherical coordination system to classify and control the congestion before packet lose in MANET is proposed.

A) Motivation

In the past, the Gauss Markov mobility model was proposed in which at each interval the next position of a node is calculated based on the concurrent position speed and direction of movement. In this model each node is assigned an initial current speed and position. After the fixed time T interval the value of the speed and direction of each node are calculated from speed and direction value at their (T-1) time interval. Find out the following value by using Eq.2,3,4 and 5

\[
S_t = S_{t-1} + (\alpha-1) \Delta S_t + \sqrt{1 + \alpha^2} S_{t-1}
\]  

\[
d_t = d_{t-1} + (\alpha-1) \Delta d_t + \sqrt{1 + \alpha^2} d_{t-1}
\]  

Where \(S_t\) and \(d_t\) are the new speed and direction of the node at interval \(t\), \(\alpha\) is the tuning parameter that is used to vary the randomness of movement \(s\) and \(d\) mean value. When \(\alpha = 1\) the node motion is completely random and the node motion is linear. At each time interval \(t\), a node's position is given by the equations:

\[
x_t = x_{t-1} + S_{t-1} \cos d_{t-1}
\]  

\[
y_t = y_{t-1} + S_{t-1} \sin d_{t-1}
\]  

Where \((x_t; y_t)\) and \((x_{t-1}; y_{t-1})\) are the x and y coordinates of the node's position at the \(t\)th and \((t-1)\)th time intervals, respectively, and \(S_{t-1}\) and \(d_{t-1}\) are the speed and direction of the node, respectively, at the \((t-1)\)th time interval. In this model, mobile device will be moves only in two direction but in space, device is free to move in any direction. To overcome this problem by using spherical coordinates system. It is a three dimensional space
Coordinate system where the position of a point is from a fixed origin its polar angle measured from a fixed origin its polar angle measured from a fixed Zenith director and the a azimuth angle of its orthogonal propagation on a reference plane that passes through the origin and is orthogonal to the zenith, measure from a fixed reference direction on that plane. Where mobile devices are free to move in space without any restriction. In spherical coordinate system, is easy to reduce the variable and conversion in Cartesian coordinate system or vice versa. It is dependent on the motions of mobile node. Proposed work is defined in this way. One mobile device is fixed as a considered as origin connected to the neighbor mobile node which device has mobility. There are two types of coordinate system used, one as spherical coordinate system and other Cartesian coordinate system. The mobile node displacement is shown in a figure 1.

![Fig 1. Displacement of Entity in Spherical Coordinate System](image)

Where θ is inclination angle from the normal axis and dθ

It is necessary to define a unique set of spherical coordinates for each point, one may restrict their ranges, and a common choice is

- \( r > 0 \)
- \( 0° ≤ θ ≤ 180° (\pi \text{ rad}) \)
- \( 0° ≤ ϕ < 360° (2\pi \text{ rad}) \)

The following equations assume that θ is inclination from the normal axis:

The line element for an infinitesimal displacement from \((r, \theta, \varphi)\) to \((r + dr, \theta + d\theta, \varphi + d\varphi)\) is

\[
\text{dr} = dr \hat{r} + r d\theta \hat{\theta} + r \sin \theta d\varphi \hat{\varphi}.
\]  

(6)

Where

\[
\hat{r} = \sin \theta \cos \phi \hat{i} + \sin \theta \sin \phi \hat{j} + \cos \theta \hat{k}
\]

\[
\hat{\theta} = \cos \theta \cos \phi \hat{i} + \cos \theta \sin \phi \hat{j} - \sin \theta \hat{k}
\]

\[
\hat{\varphi} = - \sin \phi \hat{i} + \cos \phi \hat{j}
\]

are the local orthogonal unit vectors in the directions of increasing \(r, \theta, \varphi\), respectively? The surface element spanning from \(\theta\) to \(\theta + d\theta\) and \(\varphi\) to \(\varphi + d\varphi\) on a spherical surface at (constant) radius \(r\) is

\[
dS_r = r^2 \sin \theta \, d\theta \, d\varphi.
\]  

(7)

Thus the differential solid angle is

\[
d\Omega = \frac{dS_r}{r^2} = \sin \theta \, d\theta \, d\varphi.
\]  

(8)

The surface element in a surface of polar angle \(\theta\) constant (a cone with vertex the origin) is

\[
dS_\theta = r \sin \theta \, dr \, d\varphi.
\]  

(9)

The surface element in a surface of azimuth \(\varphi\) constant (a vertical half-plane) is

\[
dS_\varphi = r \, dr \, d\theta.
\]  

(10)

The volume element spanning from \(r_0\) to \(r + dr\), \(\theta_0\) to \(\theta + d\theta\), and \(\varphi_0\) to \(\varphi + d\varphi\) is

\[
\text{dV} = r^2 \sin \theta \, dr \, d\theta \, d\varphi.
\]  

(11)

As the spherical coordinate system is only one of many three-dimensional coordinate systems, there exist equations for converting coordinates between the spherical coordinate system and others. There are three variant variables in spherical coordinates system or any one is constant and other two variables will be variant or vice versa. It depends on the behaviors of the mobile device. We are finding a velocity change and position of mobile device after the T interval by using a spherical coordinates system or coordinate system. The change of velocity is used to classify the type of losses and take a decision before losses the packet. The velocity change is representing by velocity change (CV). Each cluster has a stationary node. A stationary node is calculating velocity change and position of the moveable node as soon as node will reach near the boundary of the cluster to exit cluster, it will send response to the sending data to the moveable node. Find out displacement in figure 1. By using eqn (11).

\[
s_{\text{dr}} = \text{d} \hat{r} = r \, d\theta \hat{\theta} + r \sin \theta \, d\varphi \hat{\varphi}.
\]  

(11)

Acceleration is finding out by using a eqn (6) where acceleration was used to calculate the velocity by using given eqn below

\[
\dot{v} = \dot{r} \hat{r} + \dot{\theta} \hat{\theta} + (r \sin \theta \dot{\varphi} \hat{\varphi})
\]

\[
= \ddot{v}_r + \ddot{v}_\theta + \ddot{v}_\varphi.
\]  

(12)
If we have given position at the T time interval then find out velocity change from (T-1) time and T time positions by using below given equation.

\[ S = ut + \frac{1}{2} at^2 \]

\[ \nu = \nu \pm at \]

\[ \nu^2 - \nu^2 = 2as \]

If acceleration is constant then above following velocity equation are

\[ S = ut \]

\[ \nu = \nu \]

\[ \nu^2 = \nu^2 \]

If we have given Acceleration \( a = \text{Rate of change of velocity} = \frac{dv}{dt} \)

Velocity \( v = \text{rate of change of displacement} = \frac{dr}{dt} \)

and initial velocity ‘\( u \)’.

We can easily find out the new position of the mobile node and new velocity ‘\( v \)’ at time interval \( T \).

In figure 2, we have four clusters C1, C2, C3 and C4 respectively. Node L1 of cluster C1 is the source node as well as the stationary node of that cluster, this node controls other movable nodes of that cluster by tracing and measuring their distance from the stationary node repeatedly after a fixed interval of time.

![Fig 2. Source Mobile Node and Stationary Mobile Node Both are Same](image)

L1 node is sending data packets to D node i.e. the destination node in cluster C3. In C1 node L2 and L4 are movable nodes and L1 is sending data packets via L2, as soon as node L2 will reach near the boundary of the cluster to exit cluster C1, L1 will calculate velocity change and direction (position) and it will link up with L4 before getting disconnected from node L2, to transmit data packets to the destination without having any interruption in transmission of data. The stationary node selected by using a stationary selection algorithm. Every cluster has one stationary node to control and trace the movement of other movable nodes. The source node can be either a movable node or stationary node. In figure 3, we have L1 as the source node which is movable and L3 is the stationary node of cluster C1 respectively. L1 is sending data to destination D in C3 via L2.L2 and L4 are movable nodes and their movements are traced and controlled by L3. So as soon as L2 will reach near the boundary of cluster C1, node L3 will calculate velocity change and displacement and L3 will inform the source node L1 about the movement of L2 so that L1 can link up with L4 for transmission of data before the link between L1 and L2 breaks up due to the exit of L2.

![Fig 3. Source Node and Stationary Node are Different](image)

Algorithm for classify the type of lose

Algorithm (VC)

```
Begin
    If (VC > .9)
    {  
        They have more chance that link break  
        (find out new connection)  
    }
    else if (VC < .9 && VC > .5)
    {  
        Congestion occurred  
        (Adjust the data rate)  
    }
    else  
        Battery depletion (does not need to adjust the data rate)  
End
```

The node in a cluster which has no mobility or less mobility as compared to other available nodes in that cluster is selected as the stationary node of that cluster. The selection procedure of stationary is invocation after random amount of time. All calculation is done at the stationary node and it is responsible for Acknowledge to the mobile device, before lose a packet and a mobile device which are used as transmitter to control the date rate according receiving Ack. from stationary node. Stationary node does not need to send Ack continuous. Its sends Ack when it is predicts the possibility of packet lose. This procedure is use for reducing the packet loss rate and classifies the type of losses. We can save battery by making stationary node active at random amount of time and does not need to send Ack for each calculation.

IV. CONCLUSION

This paper motivated the need for loss classification in mobile ad-hoc networks and presented a novel approach.
which manages the mobility of node proactively, is easy to implement and imposes minimal computational burden on the resource constraint device. This approach will be required to less number of acknowledgement packets than Enhance congestion control techniques are used to address link failure and to control the congestion [11]. Result from this paper have shown that MANET performance can be improved by using novel approach as it reduces packet loss ratio (Thereby reducing the number of retransmission) and increase transmission efficiency. Moreover, as its computational burden is negligible, it is ideally suited for resource constrained environment such as MANETs. In future, our work will study the performance of novel approach in the presence of traffic in MANETs.

REFERENCES


AUTHOR’S PROFILE

Robin choudhary pursuing M. Tech. from Shobhit University Meerut in 2010-2012. His research area includes MANET (Mobile Ad-Hoc network).

Niraj Singhal received his M.Tech. in Computer Engineering (First class with honours), and currently pursuing Ph.D. in Computer Engineering from Shobhit University Meerut. He is the member of International Association of Computer Science and Information Technology.
HONG KONG, life member of Computer Society of India (CSI), and Indian Society for Technical Education (ISTE). Also he is the reviewer for World Applied Sciences Journal, The World Scientific and Engineering Academy and many other journals. He has the privilege of guiding undergraduate and postgraduate students for their research work. He has more than thirty research publications to his credit in International and National journals/conferences of repute. Presently he is working as Assistant Professor in the Faculty of Electronics, Informatics and Computer Engineering, Shobhit University, Meerut. His area of interest includes web information retrieval, system programming, and knowledge based systems.