

Review on Congestion Control Methods for Network Optimization in MANET

M.Tech. Scholar Vinita kashid, Asst Prof. Dr. Mukesh Patidar, HOD. Mr. K. K. Sharma

Department of Electronics & Communication Engineering,
Patel College of Science and Technology, Indore, MP, India
Kksharma18685@ gmail.com

Abstract- Ad hoc network is a multi-hop temporary self-organized network consisting of many mobile nodes without any infrastructure. When the packets arriving at a node cannot be forwarded, they are dropped. Congestion occurs in a network if the number of packets being sent through the network is greater than the capacity of the network. The main objective of congestion control is to minimize the delay and buffer overflow caused by network congestion and hence enable the network to perform better. In this paper, we analyze the major factors affecting performance in ad hoc networks and discuss several typical improved congestion control approaches. Network performance of these different approaches is discussed. The limitations of different approaches are also mentioned and the proposed method for congestion control using network coding is discussed.

Keywords- Congestion, MANET, Congestion control.

I. INTRODUCTION

Mobile Ad hoc network (MANET) is composed of mobile nodes communicating through wireless medium, without any fixed infrastructure. Every node can act both as a host and a router in a network, which forwards data packets to other nodes [1].

In most ad hoc networks, some areas of the network forward more data traffic than other areas. The nodes near the center of the network will tend to forward a higher load as the network routing protocol prefers shortest-path routes; this preference can make certain areas prone to congestion and can decrease the overall network throughput.

Congestion occurs in a network if the number of packets being sent through the network is greater than the capacity of the network. Consequently, an excessive number of packets arrive at a network which act as a bottleneck and leads too many packet drops. These dropped packets might already have traveled a long way in the network and thus consumes significant resources. Further, the lost packets often

Trigger re-transmissions which further increase the network traffic. Thus, network congestion severely deteriorates throughput of the network. Congestion collapse can occur, if no appropriate congestion control is performed.

As a result, data is not successfully delivered. The main objective of congestion control is to minimize the delay and packet loss caused by network congestion, resulting in improved performance of the network. Two approaches are used to handle congestion - congestion control (or recovery) and congestion avoidance. The former approach is reactive as congestion control is implemented after the congestion is detected. The latter is proactive as congestion avoidance comes into play when congestion is expected.

It has been shown that static solutions such as allocating more buffers, decreasing congestion window after successful transmission or doubling congestion window after a collision, are not effective for congestion control purposes as it causes bandwidth wastage and high variation in delay. In this survey paper, we provide a brief of existing attempts to solve the congestion problem in mobile

multihop ad-hoc net-works. Some typical improved approaches of congestion control are analyzed based on both congestion control mechanism and ad hoc network features.

II. TYPICAL IMPROVED CONGESTION CONTROL APPROACHES

Congestion control is a challenging subject in mobile ad hoc networks, mainly because of the frequent changes of the network topology. During the last years, several congestion control approaches designed to operate within ad hoc networks, have been presented. Congestion control schemes are usually composed of three components: congestion detection, congestion feedback, and sending rate control. Congestion detection can be processed in intermediate nodes or receivers.

The criteria for congestion detection vary with protocols. The basic idea behind Random early detection algorithm (RED) [2] queue management is to detect congestion early and convey congestion notification to the end-hosts. This allows them to reduce their transmission rates before queues in the network over flow and packets are dropped.

RED maintains an exponentially-weighted moving average of the queue length, which it uses to detect congestion. It compares the average queue length with a predefined threshold. Once the average queue length exceeds the predefined threshold, it starts dropping packets with given probability.

When the average queue size is less than the minimum threshold (minth), none of the packets are marked. Also, every arriving packet is marked, when the average queue size is greater than the maximum threshold (maxth). When the average queue size is between the minimum and the maximum threshold, each arriving packet is marked with probability P_a , where P_a is a function of the average queue size avg. A

s avg varies from minth to maxth, the packet marking probability P_b varies from 0 to maxp.

$$pb = \frac{\max p(\text{avg} - \text{min th})}{(\text{maxth} - \text{min th})} \quad (1)$$

The final packet-marking probability P_a is given below. It increases slowly as the count increases since the last marked packet.

$$P_a = \frac{P_b}{(1 - \text{count } pb)} \quad (2)$$

Network congestion is avoided in this algorithm and average queuing delay is reduced. One of the fundamental problems with RED and all other known active queue management techniques is that they depend on queue lengths as an estimator of congestion. The queue length gives very little information about the severity of congestion, implies, the amount of traffic sharing the link.

Explicit Congestion Notification (ECN) [3] is based on the capacity of the network to find out the probability of congestion. Contrary to the conventional approaches to avoid congestion based on packet dropping, implementation with ECN is based on the packet marking and reacts during incipient stages of congestion.

Packets are marked with probability depending on the average queue length, to avoid excessive congestion. When receivers accept such marked packets, the senders are informed about it, as a result of which the senders limit and reduce their rate of transmission in accordance with this received information.

Congestion adaptive routing protocol CRP is proposed for MANET [4]. A bypass concept is used in CRP design. A bypass is a sub-path connecting a node and the next non-congested node. When congestion is predicted, bypass path will be used. Bypass path is used for sending some of the incoming traffic, which results in the less amount of traffic arriving at the potentially congested node. The congestion may be avoided as a result.

CRP does not incur heavy overhead due to maintenance of bypass paths as bypass is removed when congestion is resolved. Since recovery of a link breakage is realized gracefully and quickly by making use of the existing bypass paths, the delay is reduced due to new-route establishment. CRP incurs fewer packet losses than existing routing protocols.

An early congestion detection and adaptive routing (EDAPR) [5] in MANET is proposed in which a NHN (non-congested neighbors) neighbors list is constructed and route is discovered to a destination through NHN nodes. All the primary path nodes periodically calculate its queue status and detect

congestion that is likely to happen. Then the nodes sends warning message to NHN nodes which are used in the primary path. The ancestor NHN node comes to know about the expected congestion and finds an alternate path to a destination by applying adaptive path mechanism. Thus, EDAPR improves performance by increasing packet delivery ratio without causing any significant additional cost.

Extra resources are needed to find alternative route using non-congestive nodes, thus it decreases node lifetime. Again, network overhead is increased as it is required to keep information about all the NHN nodes. Most congestion aware protocols for ad hoc networks use queue size as the main traffic load metric. This metric solely cannot reflect the impact of link contention due to neighbor nodes.

Contention and Queue-aware Routing (CQR) [6] is proposed that utilizes the contention information (contention window, CW) collected from the 802.11 Distributed Coordination Function (DCF) and queue size.

Two load metrics are used for route selection, including MAC layer channel contention information, and the number of packets in the queue of node. MAC layer contention information provides estimation of state of neighbor nodes, whereas queue length provides a measurement of traffic load at the mobile node itself.

Load can be effectively balanced by using this load-aware routing protocol and the performance of the ad hoc network can be improved. The calculation of the CW of a node is performed every T seconds. The well-known exponential weighted moving average method is applied to the CW old (average value) and CW sample (current value) to calculate CW,

$$CW = \alpha CW_{old} + (1 - \alpha) CW_{sample} \quad (3)$$

To better reflect the current condition of a node, α is set to 0.3, which grants a higher priority to the current sample. The calculation of the average queue size is updated every T seconds according to the following formula.

$$qlen = qlen_{old} + (1 - \alpha) qlen_{sample} \quad (4)$$

Where $qlen_{old}$ denotes the average queue length and $qlen_{sample}$ denotes the current queue length, α can be

any number selected from the range [0, 1]. The local load of node L_i can be calculated based on the CW and queue length as follow,

$$L_i = \gamma \frac{CW}{CW_{max}} + (1 - \gamma) \frac{qlen}{qlen_{max}} \quad (5)$$

Where γ is a constant. When a source node wants to send packet, a route request packet is initiated and flooded through the network. The route request packet will be processed and forwarded by each node receiving this request, until it reaches the final destination. The intermediate node adds its load value to the request message.

Once the request packet is received at the destination, the destination generates a route reply packet and sends it back to the source node. The path with smaller load is chosen. When an active route is broken, route discovery is reinitiated to find new least load route. The use of a fresh route reduces the probability of route breakdown but increases overhead.

An end-to-end rate-based flow control scheme called Explicit rate-based flow control (EXACT) is proposed, where allowed rate of a flow is explicitly conveyed from intermediate routers to the end-hosts through each data packet's special control header [7]. EXACT is rate based and is supported by the intermediate nodes.

The current bandwidth is determined by all nodes to their neighbors and local fair bandwidth shares for all flows are calculated. Explicit rate information is inserted into all passing packets by the intermediate nodes. Each node checks whether the rate it can supply for the flow of a packet being processed by it is lower than the rate currently specified in the packet header. Then the lower rate is written into the header before the packet is forwarded. Thus, the bottleneck rate is reported.

EXACT reacts precisely to rerouting and bandwidth variation, thus making it especially suitable for a dynamic MANET network. It is costly and needs much storage space and computational power if many flows coexist in the network. EXACT is targeted for networks where the flow number is relatively small. This limitation is avoided in [8], where enhancements are implemented in a router-assisted congestion control protocol termed Quick flow

control protocol (QFCP). The congestion information is appended in the special packet header and communicated to the sender.

In QFCP, each router directly uses its fair-share flow rate as the feedback and this rate remains same during the current control interval. Using the feedback information in any single ACK, the sender computes the target window size. Further, QFCP does not store per flow state on routers making it scalable for any number of flows. At the beginning of every control interval, the QFCP controller estimates the number of flows traversing this interface using the formula:

$$N(t) = \frac{y(t)}{R(t-T)} \quad (6)$$

Where $y(t)$ is the input traffic rate measured in the last interval T , and $R(t-T)$ is the old flow rate feedback given in the last control interval. The fair-share rate $R(t)$ is updated by the controller as follows:

$$R(t) = \frac{C - \beta \frac{q(t)}{T}}{N(t)} \quad (7)$$

Where C is the capacity of the output link and $q(t)$ is the minimum queue length observed in the last period d . At the arrival of a packet, the value in the rate-request field is compared with its own fair-share rate $R(t)$ and the smaller value is copied back into that field.

The rate-feedback field of the corresponding ACK packet will include this rate-request field and is sent back to the sender by the receiver. Upon receiving an ACK, the sender notes the feedback and adjusts its congestion window as follows:

$$CW_{nd} = \max(\text{feedback RTT}; MSS) \quad (8)$$

Where feedback is the router feedback on flow rate, RTT is the round-trip time measured by the sender and MSS is the maximum segment size. QFCP can help flows converge to the fair-share rate faster an efficient algorithm is proposed to support Quality of Service (QoS), by the use of load-distributing and congestion avoidance routing method [9]. The traffic flows are classified as real time traffic and normal traffic depending on their priority. The cost metric is calculated based on the link loads. Here multimedia

traffic is considered as high priority traffic and lightly loaded links are used for routing this type of traffic.

The lightly loaded path is used by normal traffic in the lack of multimedia traffic. The resources are shared among the high and low (normal traffic) priority traffic. On-demand multipath distance vector routing protocol (AOMDV) is a routing protocol that extends the AODV protocol to discover multiple paths between the source and the destination in every route discovery [10].

The computed multiple paths should be guaranteed to be loop-free and disjoint. AOMDV computes alternate paths with minimal additional overhead as compared to AODV. This is accomplished by exploiting already available alternate path routing information. The congestion can be avoided in hot spots by deviating traffic to unused network resources, thus, results in improved network utilization and load balancing.

As the number of paths increases, the complexity of a scheme that distributes traffic among multiple paths considerably increases. Therefore, it is desirable to use as few paths as possible while at the same time minimize the network congestion.

Hop-by-hop congestion control algorithms have been studied in the wire line context [11]. The feedback about the congestion state at a node is given to the preceding node. The preceding node then adapts its transmission rate based on this feedback. Such schemes react faster to congestion than end-to-end schemes. Hop-by-hop schemes require having per-flow state management in intermediate nodes that generates scalability problems.

The Rate-Based Congestion Control (RBCC) mechanism proposed in [12] adds a leaky bucket mechanism beyond TCP's window-based rate control.

In RBCC, a feedback field is appended to the header, which is used by all intermediate nodes to provide feedback on the allowed maximum rate of the flow. Each node on the route calculates its local channel busyness ratio, defined as the fraction of time the medium is locally non-idle, which is a good sign of network congestion and available bandwidth at the

MAC layer. It uses channel busyness ratio to calculate explicit and precise feedback at the transport layer.

The intermediate nodes aim to distribute the capacity fair between the flows. A RBCC sender sets its congestion window $cwnd$ to the receiver's advertised window size in packets. The author proposed Congestion Aware Adaptive (CAA) [13] routing protocol which relies on a metric that measures the degree of connectivity between a node and its neighbor. Thus, popular nodes (i.e., those having better connectivity) are favored as relay nodes to increase the delivery likelihood of a message.

In this approach, each node initially performs a self assessment of its connectivity to its neighbors and then calculates a safety margin for its buffer according to its popularity level. The buffer safety margin changes with the change in the node's popularity level. The CAA algorithm reduces queue waiting time by receiving messages destined for the more popular nodes, hence reducing the probability of buffer congestion.

At each node contact, the nodes exchange their storage availability information, i.e. buffer free space and threshold, as well as a list of nodes with the highest delivery predictability.

A sending node is allowed to forward a message to its contact node only if the message size satisfies the receiver's allowed buffer margin and the destination is in the receiver's list of nodes with high delivery probability.

A potential drawback of the algorithm is observed in large scale networks. The algorithm requires every node to maintain a node predictability table, so this table will grow linearly with the increasing number of nodes in the network, thus causing scalability issue in large networks.

The practical XOR coding scheme COPE proposed in [14] is shown to be able to improve the throughput of unicast traffic in MANETs. When a node wants to transmit, it can XOR multiple packets and transmit them as a single packet. The intended next-hop node should have enough information to decode the encoded packet. Packet reception information is exchanged to enable one node to learn which packets its each one-hop neighbor node has stored.

But, the exchange of reception information will induce extra overhead, and at times of severe congestion, the reception reports may get lost in collisions, while at times of light traffic, they may arrive too late. The table1 gives various congestion control methods along with their limitations.

III. PROPOSED METHODOLOGY FOR CONGESTION CONTROL

Recent years have seen the design of a large number of congestion control protocols for use on the Internet. Different congestion price metrics include packet loss, packet marks, packet delays, or other combinations. Congestion problem can be eliminated by realizing congestion control in the MAC or network layer. Since routers are the central places where congestion happens, they are in a better position to detect and react to such condition.

Network coding is proposed to increase the capacity of wireless networks towards the ever-increasing demand for wireless capacity. The fundamental idea in network coding is to reduce the number of wireless transmissions by encoding together different packets. In contrast with the traditional store and forward method, network coding uses a store, code and forward approach.

It allows intermediate nodes to encode packets and send/forward an encoded packet instead of the original packet. The modifications can be applied to network coding scheme by adopting adaptive network coding and forwarding scheme.

This copes with unnecessary traffic of packets in the network by allowing only few nodes to encode the packets depending on the traffic condition.

Adaptive forwarding dynamically adjusts forwarding rate locally due to channel condition. Thus nodes in the congested area forward more encoded packets than nodes in the other area.

In normal situation, traditional store and forward scheme is used for packet forwarding, but in the condition of congestion, network coding decreases a forwarding rate to save resources. A dropped or lost packet can be decoded and thus retransmission of packet is not required.

Thus, congestion can be further reduced. The length of the router's queue is made smaller by coding, thus reducing the probability that a congested downstream router will drop packets that have already consumed network resources.

The proposed Algorithm is given in brief below:

- The congestion status packets (CSP) are transmitted by a node to its neighboring nodes, in order to inform its congestion state.
- The frequency of CSP transmission by a node is inversely proportional to the traffic forwarded by the node. More is the congestion at the node; less is the frequency of the transmission of CSP packet by that node.
- The neighboring preceding node will come to know about the expected congestion at the next node depending on the frequency of arrival of CSP packet from that node. This preceding node will then encode together different packets and transmit encoded packet to congested node.
- Thus, the number of incoming packets will be less at congested node, which will resolve the congestion problem.
- As the congestion is reduced, the frequency of CSP packets transmitted by that node will again increase and the neighboring preceding node will stop encoding the packets. Thus, the congestion is avoided by adopting adaptive network coding scheme.

Table 1. Limitations of congestion control methods.

S.No	Methodology Used	Congestion Control Algorithm	Limitations
1	Load Based Queue Scheduling	Load-Balanced Adhoc Routing (LBAR) Adaptive Load Balancing (ABR) Load Aware Routing in Ad hoc (LARA)	Considers traffic size in its computation of load. It does not entirely reflect channel contention information.

2	Hop By Hop Congestion Control	Cooperative cross layer congestion control (CXCC)	Require to have per flow state management in intermediate nodes, which generates scalability problems.
3	Multipath Routing	On-demand multipath distance vector routing protocol (AOMDV)	Concurrent maintenance of multiple paths between source and destination results in overhead and aggressive use of route caching results. The complexity of a scheme that distributes traffic among multiple paths considerably increases with the number of paths.
4	Cross Layer Based Congestion Control	Cross layer-based congestion control technique (CBCCT) Cross layer congestion control for TCP(C3TCP)	Routing overhead is high
5	Receiver Assisted Congestion Control	Receiver assisted congestion control (ReACT)	This method incurs some routing overhead since it waits for multiple route replies to return to the source node.
6	Router Assisted Congestion Control Approach	Explicit Rate based congestion control (EXACT) Quick flow control protocol (QFCP)	Increases complexity and overhead at routers for computing rate allocation for flows

7	End To End Rate Based Flow Control	Rate based congestion control (RBCC) Explicit Rate based congestion control (EXACT) Explicit Rate based congestion control (XRCC)	It is costly and needs much storage space and computational power if many flows coexist in the network
8	Congestion Adaptive Routing	Dynamic congestion detection and control routing (DCDR) Early congestion detection and adaptive routing (EDAPR) Congestion adaptive routing protocol (CRP)	The queue at any single node cannot completely reflect the network congestion state. Unnecessary flooding of control packets in the network increases network traffic.

IV. CONCLUSION

It is clear from the algorithms available for congestion control, to have adaptive solution for congestion in the network. Current congestion control is seen as transport layer issue. It is desirable to tackle problem where it emerges, thus we can realize congestion control in network layer. Different approaches are discussed for congestion control and their limitations are also discussed.

The existing routing protocols proposed for MANETs has certain limitations. Router assisted congestion control approach and end to end rate-based flow control creates complexity at routers. Hop-by-Hop Congestion Control approaches induces scalability problem. Consequently, a well-balanced congestion control system is to be employed for the stability and optimized performance of the wireless network.

Adaptive Network coding depending on the channel conditions is proposed to resolve the problem of congestion in the network. The routing protocol will be modified to cope up with the dynamics of Mobile Adhoc network and provide better performance of the network.

REFERENCES

- [1] S. R. Murthy, B.S. Manoj, "Ad Hoc Wireless Networks - Architectures and Protocols", Pearson Edu. 2007.
- [2] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance", IEEE/ACM Transactions on Networking, Vol. 1, No.4, pp 397-413, 1993.
- [3] Jianxin Wang, Jie Chen, Shigeng, Weiping Wang, "An explicit congestion control protocol based on bandwidth estimation" 978-1-4244-9268-8/11/26:00; IEEE; 2011.
- [4] Duc A, Tran, Harish Raghavendra, "Congestion adaptive routing in mobile ad hoc networks," IEEE Transaction of Parallel Distributed System, Vol.17, No.11, pp16-28, 2006.
- [5] Kumaran, T.S. and V. Sankaranarayanan, "Early congestion detection and adaptive routing in MANET," Egyptian Information Journal, Elsevier, Vol.12, pp 165- 175, 2011.
- [6] Xuemei Gao, Xinming Zhang, Dong Shi, Fengfu Zou, Wenbo Zhu, "Contention and Queue-aware routing protocol for mobile ad hoc net-works," IEEE International Conference on. Wireless Communications, Networking and Mobile Computing, 2007.
- [7] K. Chen, K. Nahrstedt, and N. Vaidya, "The utility of explicit rate based flow control in mobile ad hoc networks," in Proc. IEEE Wireless Communications and Networking Conference (WCNC 2004) Atlanta, GA, 2004.
- [8] Pu, Jian, and Mounir Hamdi. "Enhancements on router assisted congestion control for wireless networks." IEEE Transactions on Wireless Communications, vol. 7, no.6 pp 2253-2260, 2008.
- [9] Sivakumar, P. and K. Duraiswamy, "A QoS routing protocol for mobile ad hoc networks based on the load distribution", Proceedings of the IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), pp: 1- 6, 2011.
- [10] K. M. Mahesh, and S.R. Das, "On-demand Multipath Distance Vector Routing in Ad Hoc Networks", Proceedings of IEEE ICNP, November 2001.
- [11] Y.C. Hu and D. B. Johnson, "Exploiting Congestion Information in Network and Higher Layer Protocols in Multihop Wireless Ad Hoc Net-works", 24th International Conference on Distributed Computing Systems (ICCS'04), IEEE, Hachioji, Tokyo, Japan, March 24-26, 2004.

- [12] H. Zhai, X. Chen, and Y. Fang, "Rate-Based Transport Control for Mobile Ad Hoc Networks", In WCNC '05: Proceedings of the IEEE Wire-less Communications and Networking Conference, volume 4, pages 2264- 2269, Mar.2005.
- [13] T. Kathiravelu, N. Ranasinghe, A. Pears, "Towards Designing a Routing Protocol for Opportunistic Network", Proc. Intl. Conf. on Advances in ICT for Emerging Regions, Colombo, Sri Lanka, Sep. 2010.
- [14] S. Katti, H. Rahul, W. J. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the Air: Practical Wireless Network Coding", In: Proc of ACM SIGCOMM 2006, Pisa, Italy, Sep 2006, pp. 243-254 .