International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 3, Issue 3, March 2014

Mobile Ad hoc Networks and Congestion Control

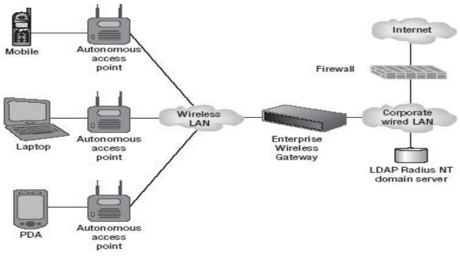
B. Subramani[#] **Dr. T. Karthikeyan**[*]
[#]Head, Dept of Information Technology, Dr.N.G.P Arts and Science College, Coimbatore, India.
[*]Associate Professor , Dept of Computer Science, P S G College of Arts and Science, Coimbatore, India.

Abstract

Mobile Ad-hoc networks have Congestion control as its main issue to be solved while transferring data. This considerable challenge is due to regular changes in network topology and sharing of wireless channel among all users. So many approaches are proposed to overcome this issue and the interrelations among those approaches, are going to be the key point of this paper.

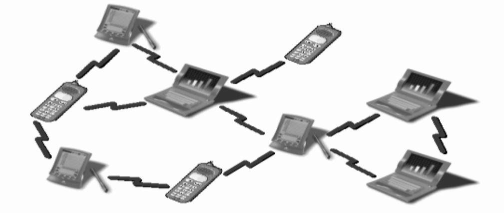
Introduction

In this paper we discuss the issue of congestion control in Mobile ad-hoc networks. Mostly the users communicate over wireless medium uses base stations and backbone networks as their infrastructure. Few others use direct communication with the help of their communication peers. In Mobile ad-hoc networks, every nodes act as switching elements and the packet data are transmitted via intermediate nodes to the defined destination.



(a). Infrastructure-based Wireless Network

(b). Mobile Ad-hoc Network



Recent researches on MANET (Mobile Ad-hoc Networks) area concentrates more on routing, along with the properties of transport layer's functions.

The key issue considered in MANETs is the appropriate *congestion control mechanisms* for the congestion-related problems. This paper discusses few existing attempts to overcome the difficulty in solving the congestion – related problems.

Core Idea

The core idea of this section is to provide the understanding on "What is Congestion Problem?" and "How to handle the congestion on Internet using TCP Congestion Control Mechanism?"

What is Congestion?

It is important to overcome the overload of a network that occurs when more number of users trying to share the common resource and compete for the link bandwidth. Few packets that cannot be transferred by the router are dropped, even after a long travel over the network medium. The retransmitted packets are sent into the network are more in number and causes congestion, which in turn affects network throughput.

So, appropriate congestion control mechanisms are important in these situations to accomplish the successful delivery of data to the destination. Occurrence of such situation in internet made the researchers to develop the TCP congestion control mechanism.

TCP Congestion Control

TCP always considers the losses of packets in the network to identify the occurrence of the congestion. By nature, all the missing packets are interpreted as a sign for network congestion.

TCP sends the consolidated acknowledgement after the transfer of the current segment is over. When segments are received with some missing packets, then the TCP sends the last acknowledgement once again. TCP uses timeout that depends on the measured round-trip time of the connection. If this RTO-Retransmission Time Out elapses without an acknowledgement TCP concludes severe congestion.

During the first phase of the connection and after a timeout a mechanism named slow start is employed. It allows for a faster convergence to the correct window size. While slow start is active, the window size is not increased by one segment size for every round-trip time, but instead for every received acknowledgement. This means that during this phase the window size grows exponentially.

During the option Delayed Acknowledgement, an acknowledgement is not sent for every single segment, but is sent only for every second segment, or after some timeout at the receiver has expired. This reduces the number of acknowledgement packets.

The selective acknowledgement option allows for a more fine-grained feedback on missing segments than the standard cumulative acknowledgement scheme. It is particularly useful if multiple segments are missing in the same round-trip period.

Mobile Ad-hoc Networks and Congestion:

TCP congestion control best suits for Internet, but not for MANETs. Because MANETs have some specific properties that are unique in nature, which affects the appropriate protocols design, stack and congestion control mechanisms.

Vital properties of MANETs are mobility of nodes and shared wireless channel. As nodes are in mobile, and the medium is non-reliable, packet delivery is delayed and some packets losses. These losses or delays should not be misinterpreted as congestion losses.

The connectivity range for one node can allow only one transmission of data at a time through the shared medium. Because of this nature, the network congestion manifests itself tremendously. Internet router typically is dedicated hosts connected by high bandwidth links. When congestion occurs on the internet, it is usually concentrated on one single router.

In contrast, congestion in MANETs affects a whole area because of the shared medium. Although this depends on the network type, packet losses which are not cost by network congestion can be much more frequent in wireless networks.

The heterogeneous application scenario of wireless networks allows appropriate congestion control solution for a specific network and application will largely depend on the properties and purpose of the respective network. There is no chance for the existence of a single general-purpose solution for all possible scenarios.

Route Failures

The route failures occur frequently in MANETs, because the amount of time required discovering a new route has a negative impact on the standard TCP congestion control mechanism. In some situations like, no packets delivered or no acknowledgements received, may not lead to real congestion situation, but the TCP sender reduces its size.

TCP-Feedback is an important approach to deal the congestion control problem in MANETs. The signal strength based link management is identified as the best approach to anticipate the route failures.

Wireless Losses

Wire-line connection has less chance for more or less packet losses when compared to wireless connection. Those losses are disadvantageous for a transport protocol performance if they are misinterpreted as congestion-induced packet drops.

Packets are dropped by the link layer only after a number of failed transmission attempts. This in turn usually happens when either a link is lost or when a lot of packet collisions occur.

Therefore although the packet loss has not been caused by a queue overflow in a router it can still be a valid congestion indicator.

The inter-packet delay difference at the receiver, defined as the time elapsing between two successive packet arrivals, increases when congestion occurs. Additionally, the short-term throughput, describing the throughput in a certain time interval in the immediate past, decreases in case of congestion. These two metrics are combined to gain a more robust congestion indicator. In a similar way, out-of-order packet arrivals and the packet loss ratio are used to detect route changes and channel errors.

In edge-based approach the TCP congestion control reaction is circumvented if a medium loss is detected. Also, route failures are detected when a timeout occurs and no packets at all have been received for a longer period of time.

Shared Medium Management

In a wireless network the medium is shared by all nodes in a certain area. Dealing with this property is a big challenge when one wants to perform congestion control in such networks: it makes congestion a spatial phenomenon, happening no longer in a node, but rather in an area. Several mechanisms paying attention to these special limitations have been proposed.

For a given topology and traffic pattern there exists an optimal TCP window size, but TCP is unable to find it. Instead it uses larger windows, leading to dropped packets caused by link-layer contention. This observation has influenced research in wireless congestion control significantly. Given this background the authors propose two mechanisms to improve TCP by earlier reaction to link overload—a distributed Link RED (LRED) and an adaptive pacing strategy.

COntention-based PAth Selection (COPAS), focus on a problem of TCP in MANETs called the capture problem. Nodes can capture the medium unfairly and gain an advantage in comparison to others. COPAS is an extension for reactive routing protocols.

During route discovery all routes between a source and destination node are gathered. Then two disjoint routes are used to forward upstream TCP traffic and downstream acknowledgments respectively, in order to avoid effects where one of the two directions captures the medium. The decision which routes are chosen is based on congestion measurements performed during the discovery process. The measurements are based upon the backoff times for which the node had to wait before the medium became free. They are updated continuously during operation, and a route that becomes too congested is substituted by a better one. Interestingly, the fact that two disjoint routes are used for forward and backward traffic is perfectly opposed to the Symmetric Route Pinning technique where special care is taken to use the same route.

ACK Traffic Handling

Because of the shared medium, packets using the same route—or spatially close routes—in opposite directions severely affect each other. A very prominent example for this situation is the end-to-end acknowledgment traffic generated by transport protocols, causing intra-flow contention between data packets and acknowledgment packets traveling in opposite directions. The question arises how the amount of ACK traffic or at least its negative impact on the performance of the forward channel can be minimized. This is closely related to the effects caused by the shared medium in general. Consequently there is some overlap with the previous section, and some of the approaches described there also consider the interplay between oppositely-directed data and ACK traffic. The work described in this section focuses solely on the acknowledgment traffic.

The Mechanism in intermediate nodes for combining oppositely-directed TCP data and ACK packets into one common frame is to avoid repeated collisions of ACKs with data packets from the same stream. Their main idea is to avoid using a whole FRN (Flexible Radio Network) time slot for a very short ACK frame and at the same time to reduce the probability of packet collisions by combining a data and an ACK packet into a common frame if the transmitting node has packets of both kinds in its queue. This frame has two destination addresses, one for each of the two parts. One of the two next hop nodes delays forwarding the packet for one time slot, in order to avoid a collision when both parts are forwarded further.

Protocol Designs

Development of new reliable transmission protocols that are specifically tailored to cope with the characteristics of MANETs leads to the protocol designs. Most approaches are also limited to “clean” environments where no other transport protocols are used.

MANETs can often be expected to be rather small, closed environments, such constraints can be perfectly reasonable. Additionally, it seems that it might even turn out to be absolutely necessary to rely on completely different queuing and congestion control paradigms than those used by TCP in networks with media properties like those of wireless networks.

Explicit rate information is inserted into all passing packets by the intermediate nodes to transmit the minimum bandwidth at the bottleneck to the receiver of the flow. Each node checks whether the rate it can supply for the flow of a packet it processes is lower than the rate currently specified in the packet header. In this case the lower rate is written into the header before the packet is forwarded. Thus the bottleneck rate is reported in the end.

This mechanism is used twice, i. e. on two different header fields. One field contains the current rate of the sender and another one the rate requested by the sending application. On the one hand with this procedure it is possible for the intermediate routers not to give a flow more bandwidth than it needs, and on the other hand the sender is notified when it is allowed to increase its rate above the current level. A safety window prevents the sender from overloading the network in case of a route failure. A sender is not allowed to have more unacknowledged packets underway than the size of the safety window.

Ad-hoc Transport Protocol does not use retransmission timeouts, strictly separates congestion control and reliability mechanisms and requires only limited feedback from the receiver. ATP does not require any flow-specific state variables in the intermediate nodes. All nodes calculate an exponential average of the delay of all packets passing through them. This delay consists of the time a packet had to wait in the node's local queue and of the time to wait for a free medium before it could be transmitted. These values are independent of the flows the packets belongs to.

The current delay value is piggybacked onto forwarded data packets if it is worse than the information currently in the packet's header. This way, the maximum delay over the packet's path is communicated to the receiver. The receiver aggregates this information and sends it back to the sender. Based on this information the sender can adapt its rate. To find a good rate at the start of a new connection a probe packet is sent along the route collecting information from the intermediate nodes about the current state of the network. For the acknowledgments of data packets a selective ACK scheme is employed in ATP.

Application controlled Transport Protocol is based on the observation that TCP's throughput in MANETs is very low, while UDP achieves reasonable throughput, but suffers from a high packet loss rate. ATP is meant to be somewhere in between TCP and UDP—UDP with optional packet delivery status feedback. The protocol supports packet acknowledgments, feedback is given to the application if an acknowledgment for a given packet has arrived or not. An application using ATP is expected to do retransmissions on its own, if they are necessary. In order to reduce the number of retransmission of packets, the decision on retransmission is left to the applications whether it is necessary or not. However, no other transport layer components are implemented; especially congestion control would have to be provided by the application itself. MANETs severe congestion problems can not be solved only with the ATP approach.

Wireless eXplicit Congestion control Protocol shares some fundamental concepts with TCP. WXCP uses explicit feedback from within the network and multiple congestion metrics. These are evaluated at the intermediate nodes, in order to avoid the necessity of probing for the highest available bandwidth.

A transport protocol for mobile ad-hoc networks is **TPA**. Its congestion control mechanism is inspired by TCP, but designed to minimize the number of required packet retransmissions. Packets are transmitted in blocks using a window-based scheme. A fixed number of packets is grouped into a block and transmitted reliably to the destination before any packet of the next block is transmitted. Packet retransmissions are not performed before every packet of a block has been transmitted once—thus a block is transmitted in several rounds: first every packet is transmitted once, then not yet acknowledged packets of this block are retransmitted until every packet of the block has been delivered and acknowledged. TPA enters a freeze state upon route failures, by decreasing the window size to one. For congestion control TPA uses a window mechanism with a tightly limited maximum window size. Two different values are used: a "large" window of 2 or 3 segments during normal operation and the minimum value of 1 when congestion is detected. TPA shows that even a quite simple end-to-end protocol without additional intelligence in the intermediate nodes has the potential to increase throughput in comparison to TCP. However, it is not yet clear if these benefits can be maintained in more complex, dynamic scenarios. Additionally, for time critical applications the higher latency introduced by the protocol might be a problem.

Conclusion

The proposed approaches to improve transport layers for MANETs have two aspects: First, the set of protocols that are trying to improve along with maintaining compatibility with established, wide-spread protocols, mainly with TCP. They also try to avoid any cross-layer feedback in order to stay independent from any specific network stack. The second one is that many approaches readily give up compatibility to gain more freedom in protocol design and hence to even better fit the specific needs of MANETs. Both attempts to address the problems have their respective application areas. The best solution to the problem will be a completely wireless network with few, well-known and potentially newly developed applications – a custom approach providing superior performance. In addition to this, the the overall system architecture design might depend largely on realizing the additional performance by choosing a non-standard protocol with higher degree of deployment and maintenance effort. Another striking fact is that a number of design patterns occur in multiple protocols. These common ideas and concepts have either been developed in multiple contexts or have been adopted from earlier approaches.

Future Scope

Currently congestion control, as considered in this paper, is often seen as a transport layer issue. A new perspective on this problem might be to realize congestion control in the network layer. After all, it might make sense to tackle the problem where it emerges. An exceedingly high network load is a problem closely associated with medium access and packet forwarding. Some approaches already follow this direction and separate congestion control strictly from reliability measures. Here a wide spectrum is open for more fundamental research.

References

1. V. Anantharaman, S.-J. Park, K. Sundaresan, and R. Sivakumar. TCP

performance over mobile ad-hoc networks: a quantitative study. Wireless Communications and Mobile Computing.

2. G. Anastasi, E. Ancillotti, M. Conti, and A. Passarella. TPA: A Transport Protocol for Ad Hoc Networks. InISCC ’05: Proceedings of the 10th IEEE International Symposium on Computers and Communication.

3. D. Berger, Z. Ye, P. Sinha, S. Krishnamurthy, M. Faloutsos, and S. K. Tripathi. TCP-Friendly Medium Access Control for Ad-Hoc Wireless Networks: Alleviating Self-Contention. In Proceedings of the 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems.

4. K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash. A Feedback Based Scheme for Improving TCP Performance in Ad-Hoc Wireless Networks. In ICDCS ’98: Proceedings of the 18th International Conference on Distributed Computing Systems.

5. R. de Oliveira and T. Braun. A Dynamic Adaptive Acknowledgment Strategy for TCP over Multihop Wireless Networks. In INFOCOM ’05: Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies, volume 3.

6. R. de Oliveira, T. Braun, and M. Heissenb¨uttel. An Edge-based Approach for Improving TCP in Wireless Mobile Ad Hoc Networks. In DADS ’03: Proceedings of the Conference on Design, Analysis and Simulation of Distributed Systems, Orlando, USA, Mar. 2003.

7. T. D. Dyer and R. V. Boppana. A Comparison of TCP Performance over Three Routing Protocols for Mobile Ad Hoc Networks. In MobiHoc ’01: Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing.

8. G. Holland and N. H. Vaidya. Analysis of TCP Performance Over Mobile Ad Hoc Networks. In MobiCom ’99: Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking.

About Authors:

B. Subramani received his M.C.A., from Madras University and M.Phil from Manonmaniam Sundaranar University. Currently he is a Ph. D. student at Bharathiar University under the guidance of

Dr.T. Karthikeyan. Associate Professor, Department of Computer science , P S G College of Arts and science, Coimbatore, INDIA. His current research interests, Image Processing, Networks, congestion control in mobile ad-hoc networks.