ISSN 1512-1232

NODE FEED-BACK BASED TCP SCHEME FOR MOBILE AD-HOC NETWORK

F. Alam

School of Computing and Mathematics, Liverpool John Moores University
Byrom Street, Liverpool L3 3AF, UK
cmpfalam@livjm.ac.uk,

Abstract

TCP is a reliable connection oriented transport layer protocol widely used for the Internet. But its performance decreases when deployed over mobile ad-hoc networks. TCP assumes all packets losses are due to congestion therefore invoking congestion control mechanism thus reducing network throughput. Many schemes have been reported to improve TCP performance in mobile ad-hoc network however the problem is still unresolved at a satisfactory level.

We have proposed a node feedback based TCP scheme for mobile ad-hoc network (NFBTCP), addressing with congestion control and slow start mechanism of TCP. In our approach route failure notification is introduced to inform TCP sender about path break or route failure. In NFBTCP TCP sender adjusts the size of congestion window (CWND) according to the link capacity of the established connection. In this way TCP doesn't need to invoke slow start mechanism. A new mechanism to start new TCP connections between two or more nodes in mobile ad-hoc network is also included as a part of NFBTCP. We believe NFBTCP could yield a solution which can give an impressive data delivery ratio in mobile ad-hoc network. NFBTCP has been implemented in Java and evaluated in SWANS. Results showed that NFBTCP performed well in different simulation environment.

I. INTRODUCTION

Mobile ad-hoc networks have become increasingly important because of their promise of ubiquitous connectivity beyond traditional fixed infrastructure network. Mobile ad-hoc network due to potentially high mobility have provided new challenges by introducing special consideration differentiating from the unique characteristics of the wireless medium and the dynamic nature of the network topology. Due to the unique structure of a mobile ad hoc network it can be deployed anywhere at any time where fixed networks cannot be deployed. The main applications of mobile ad hoc network are in emergency situations such as during earthquake, floods and disasters etc.

Routing is transfer of data packet from source to destination node through wireless medium in mobile ad hoc network. Because of its unique structure, routing is the main issue in mobile ad-hoc network. Many protocols with various techniques have been proposed as a solution to this problem such as, Destination Sequence Distance Vector (DSDV), Dynamic Source Routing (DSR), Ad-Hoc On-demand Distance Vector Routing (AODV), and Temporally Ordered Routing Algorithm (TORA) [1]. Routing in mobile ad-hoc network is an unresolved issue and still open area for research.

Transmission control protocol (TCP) is the most reliable transport layer protocol for the Internet [1]. TCP is responsible end-to-end connection, congestion control, flow control, in order delivery of packets and reliable transportation of data packets [5, 6]. Node feedback based TCP mechanism aims to improve TCP performance in a mobile ad-hoc network. NFBTCP addresses known TCP problems of mobile ad-hoc network. The proposed scheme introduced various measures to resolve these issues. NFBTCP uses failure notification to enable TCP sender differentiating between the real congestion and congestion assumed by the TCP due to link loss or route failure. Details of NFBTCP specification is covered in the later part of this paper. Rest of the paper has been organized as follows. In section 2 related work is covered. Section 3 explains Node feed-back based TCP mechanism (NFBTCP). Discussion and Conclusions is given in section 4.

II. RELATED WORK

A significant amount of research has been done to make TCP capable of supporting communication over mobile ad-hoc network [2, 3, 4, 5, and 6]. In mobile ad-hoc network packets are lost because of frequent path breaks due to mobility of destination node or mobility of the nodes working as routers between source node and destination node, high bit error rate (BER) in the wireless channel, collisions due to hidden terminals etc, when the data packet is lost and the sender dose not receive acknowledgement (ACK) from the receiver with in the retransmission timeout (RTO) period then TCP sender assumes this as congestion and invokes the congestion control mechanism [5]. When TCP sender assumes packet loss as congestion then it shrinks its congestion window (CWND) and reduces the packet transfer rate and thus degrades overall throughput of the network. To gain high throughput from the network TCP should differentiate between congestion and packet loss due to mobility or path breakage. Some of the known reported schemes of this area are TCP feedback (TCP-F), TCP explicit link failure notification (TCP-ELFN), Ad-hoc transmission control protocol (ATCP), Split-TCP and explicit congestion notification ECN.

In TCP-F [4] the RRN packet is generated when the intermediate node detects re-establishment of broken path and it depends on information from routing protocol. TCP-F has an additional state compared to the traditional TCP state machine, and hence its implementation requires modifications to the existing TCP libraries. Another disadvantage of TCP-F is that the congestion window used after a new route is obtained may not reflect the achievable transmission rate acceptable to the network and the TCP-F.

In TCP-ELFN [6] when the network is temporarily partitioned, the path failure may last longer; this can lead to the origination of periodic probe packets consuming bandwidth and power. Another disadvantage is that the congestion window used after a new route is obtained may not reflect the achievable transmission rate acceptable to the network and the TCP receiver.

ATCP [5] depends on the network layer protocol to detect the route changes and partitions, which not all routing protocols may implement. Addition of a thin ATCP layer to the TCP/IP protocol stack requires changes in the interface functions.

Split-TCP [3] requires modifications to TCP protocol. The end-to-end connection handling of traditional TCP is violated. The failure of proxy nodes or frequent path breaks, affects the performance of split-TCP. Comparing the two approaches, we find that end-to-end approaches are easier to implement and provide more flexibility, while feedback approaches are more accurate as the information is coming directly from the network. Furthermore, it is clear that each approach deals only with one or subset of the factors causing the bad performance of TCP in MANETs. However, most commonly, these solutions deal with route failures. Actually, this is reasonable because in such a dynamic environment the frequency of route failures is very high due to nod mobility. We also find that most of the presented approaches take reactive actions. In these approaches TCP takes different actions rather than invoking congestion control when a non-congestion loss occurs. Some approaches are preventive (e.g. Split TCP). The target of this kind of approaches is to reduce the probability of other losses that may lead to false notification and unnecessary congestion control reaction.

ECN [5] is appealing to be used in the Internet since it does not render any overhead regarding the current IP flows. Its drawback lies in the fact that to be effective, it requires changes to every network element. In the light of above this could be concluded that the problem of TCP within ad-hoc network are not resolved at a satisfactory level thus requires a solution which not only addresses identified issues but also gives an impressive throughput.

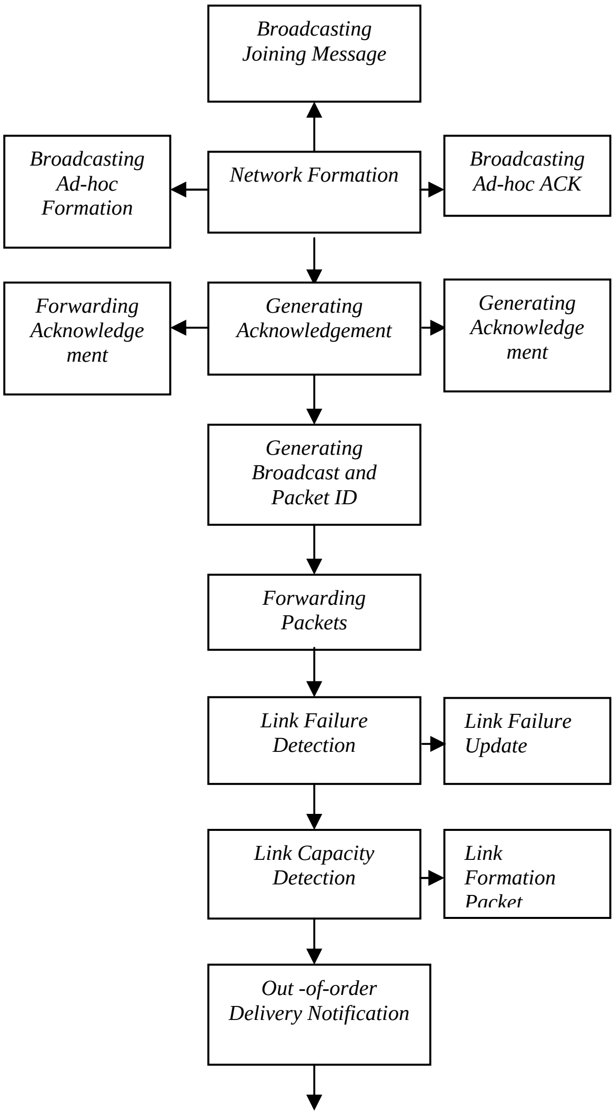
III. NODE FEED BACK BASED TCP MECHANISM FOR MOBILE AD-HOC NETWORK

NFBTCP addressed TCP slow start mechanism in the context of mobile ad-hoc network and introduce measures through whom TCP can differentiate between real congestion and congestion assumed by TCP due to packet lost or route failure in mobile ad-hoc network. In the light of background research, it is well understood that TCP poor performance over ad-hoc network is related with the typical nature of the ad-hoc network. Therefore it was necessary to define some of the interrelated operations alongside modification to TCP. All of these operations are made part of this specification.

A. NFBTCP Specification and Operational Details

This section present NFBTCP specification which could be viewed as operations required within the existing TCP structure and the remaining deal with the network operation. In this context, it is important to mention that the purpose of this section is to take into consideration of all those aspects which are involved in the routine network operations. In the following section details of various operations alongside some of the associated operations of the main operation are presented besides explanation of term which is used as a part of this specification.

Fig.1. NFBTCP OPERATIONAL STRUCTURE


Broadcasting Joining Message
Broadcasting Ad-hoc Formation
Network Formation
Broadcasting Ad-hoc ACK
Forwarding Acknowledge ment
Generating Acknowledgement
Generating Acknowledge ment
Generating Broadcast and Packet ID
Forwarding Packets
Link Failure Detection
Link Failure Update
Link Capacity Detection
Link Formation Packet
Out -of-order Delivery Notification

A1. Network Formation

NFBTCP defines ad-hoc network formation in between two or more mobile nodes. In order to establish a network, a packet named as ad-hoc formation packet (AFP) is broadcasted by any nodes Node A is broadcasting ad-hoc formation packet to the rest of the nodes to form the potential network.

Nodes wish to be part of the network on reception of AFP send Ad-hoc Acknowledge Packet back to the sender node where the nodes apart from node A sending ad-hoc acknowledge packet. This packet shows nodes are agreed to join the network. This automatically updates all the participating nodes with the relevant information about other nodes of the network.

It is important to note that this initial communication is taken as a starting point of communication between the participating nodes of a mobile ad-hoc network. These initial packets transmission is used to gather relevant information of other nodes of the network where nodes are updating their self with the information of other nodes. NFBTCP besides introducing a unique way of network formation also enables any other nodes want to join the established network via

B**roadcasting joining message**: Any node that was not part of the network at the time of network formation can broadcast join message containing Join Packet to introduce itself as a new participating node.

Below mentioned is the explanation of some the packets which are used within network formation of NFBTCP. Each of these packets are used to keep updated nodes with the current situation of the network as any node which receive any of these packet is required to forward it to the next hop neighbor. In this routes are formed and nodes in an ad-hoc network are linked with each other.

Ad-hoc Formation Packet (AFP): Ad-hoc formation packet is the packet which is broadcasted when two or more mobile nodes decided to form an ad-hoc network. This packet serves two purposes. Firstly it indicated to the other nodes within the proximity that an ad-hoc network is about to establish and lastly it gives other potential nodes a starting point of communication via some relevant information about the other participating nodes. This packet also helps nodes in determining the hop-count of one the other node.

Ad-hoc Acknowledge Packet (AACK): This packet is broadcasted by the potential nodes upon reception of ad-hoc formation packet. This serves two purposes. Firstly it shows willingness of a node or of nodes to become one of the nodes of the network being formed. Secondly, it also given option to the re-verify previous information of other nodes gathered via Ad-hoc Formation Packet.

Joining Ad-hoc Packet (JAP): Nodes which were not the part of network at the time of network formation are require to send a Joining Ad-hoc Packet (JAP) to join the network. This packet serves two purposes i.e. informing participating nodes about the new joining nodes and to update and re-verify previously stored links.

A2. Generating Acknowledgement

NFBTCP offer modification to the TCP scheme of generation acknowledgement for the sender nodes of a packet. Rather it stresses that any node which receive a packet over a link is responsible of sending acknowledgement packet for the sender node. This process is known as generating acknowledgement and this is achieve via acknowledge packet delivery from destination to the source node of a packet. It should be noted that such packets are only sends when packets with data packets are received at some destination. However, for control packets their individual acknowledgement depends on the type of the packet sends, some of such control packet and their acknowledgment type is defined in section 3.1.1. These packets are sending back to the source node using the same path developed during the delivery of packet from the source to the destination node.

NFBTCP modifies TCP approach to discovered link break in active path which is covered in the later section of this chapter. NFBTCP defines number of operations which are linked within the generating acknowledgement. The details of these operations are covered within this section and are as follows.

Verifying Broadcast and Packet ID: An intermediate node of acknowledged packet perform three operations in sequence, verifying broadcast and packet ID being the first function during which intermediate or acknowledge packet receiving node verify that it has not received the same packet before. If a packet with the same broadcast and received ID has received before it is discarded and no further action is taken.

Updating and Re-verifying Information: once it has confirmed that the packet with the same ID has not received before acknowledge packet receiver node updates relevant information about the sender of acknowledged packet or other nodes before this node with the fresh information receive contain in the acknowledge packet.

Forwarding Acknowledged Packet:

Intermediate nodes are responsible to forward packets to the other node if the route is known to them. In the case of an acknowledged packet, the same route through which packet received from the sender route is used. Therefore nodes which are used during the first route use the same route from their storage to send it back to the source node.

Acknowledge Packet (AP): This packet is send by the receiver of packet send by a node the node of the network.

A3. Generating Broadcast and Packet ID

This is an important aspect of NFBTCP as this is requiring avoiding loop problem within ad-hoc network. In general if a packet is broadcasted and if it is not received at the destined location, there are chances when this packet will loop around the network from one to the other node. In this case either packet is eventually dropped or expired after its expiring time. Therefore in NFBTCP all the participating nodes are required to generate fresh broadcast and packet ID for each individual transmission. This same procedure is followed in some of the earlier mentioned control packet for different purpose.

A4. Forwarding Data Packets

In NFBTCP whenever any intermediate node receives a packet destined for any other node of the network it first sees whether it has a route to the destination. If a route is found it uses the same route to transfer the received packet to the destination node. However, if no route is found for the destination node it forwards the data packet to the next hop neighbor.

If a route is found the following operations are performed in sequence before forwarding the data packet using the found route to the destination node. If a route is found nodes perform number of different action in sequence.

Verifying Broadcast and Packet ID: An intermediate node of acknowledged packet perform three operations in sequence, verifying broadcast and packet ID being the first function during which intermediate or acknowledge packet receiving node verify that it has not received the same packet before. If a packet with the same broadcast and received ID has received before it is discarded and no further action is taken.

Updating and Re-verifying Information*:* once it has confirmed that the packet with the same ID has not received before acknowledge packet receiver node updates relevant information about the sender of acknowledged packet or other nodes before this node with the fresh information receive contain in the acknowledge packet.

A5. Link Failure Detection (LFD)

Mobile ad-hoc network by nature suffers with frequent topology changes and link failure happens unpredictably, detecting such failure in mobile ad-hoc network is an important aspect to be seen. Such failure could be a means to degrading TCP performance over mobile ad-hoc network that makes this as an interrelated issue with the problem being investigated. It is assumed that the quicker we can detect such failure the better it could be for the network. Moreover, it could also add extra burden on the network via unnecessary data and control packets to the same route without knowing the route is broken. This could further leads to a point where network congestion could occur. If for a long time no packets are delivered and no acknowledgments are received, causing the TCP sender to reduce its window size dramatically, even though in fact no real congestion situation might exist.

NFBTCP introduces a new mechanism of updated notification to address this issue. The main aims of NFBTCP are the minimization of route failures, their prediction and a fast notification of the source in case of a route failure. In NFBTCP routing protocol which is used alongside TCP is made responsible of sending updated notification whenever a link failure is detected. This could stop sender to send any further packet using the broken route. Since, routing protocol is used alongside TCP which can also be made aware of the situation. In this case TCP will no longer be required to utilize its normal procedure of transporting packets via the same route. Specific action will be taken to communicate any updated link failure with TCP as effectively as possible in the implementation phase of this research project.

It could further be noted that Routing protocols for mobile ad-hoc network follows different strategies for route managements depend on the routing protocol used, this information can assists routing protocol in route management process and thus could make it easy to introduce such mechanisms in the overall communication structure of routing protocol used and TCP. It will be worthwhile to mention that AODV will be used alongside TCP to verify various concepts of the scheme alongside TCP in a simulation environment.

Link Failure Update Packet (LFUP): Link Failure Update Packet contains details of failed route or link. This packet is broadcasted by an intermediate route which is in between the sender and the receiver of an active communication. This node on finding any route failure can broadcast LFUP. This could also update the routing tables of all other nodes in an active path.

A6. Link Capacity Detection (LCD)

It is well known in the context of mobile ad-hoc network that link breakage happens frequently and unpredictably. This results in data loss and could also slow down the network speed. Protocols for mobile ad-hoc network deal this problem in various manners. However in the case of TCP, TCP suffers with two main problems, congestion and slow start mechanism. Whenever TCP recovers from congestion or after retransmission timeout (RTT) it invokes slow start mechanism. TCP shrinks its transmission rate to one segment (i.e. the size announced by the other end or the default, typically 512). Each time Acknowledgement (ACK) is received, the congestion window (CWND) is increased by one segment. The sender can transmit up to the maximum of the congestion window size.

In NFBTCP link capacity information is available at routing table of the routing protocol. Additional parameter could be added in the existing specification of the routing protocol mentioned above to store such information. Therefore, whenever a new link is established nodes involves in the active communication update their routing table with the link capacity information. In addition, all the nodes in between the active sender and receiver could also be updated.

TCP sender can get information from link formation packet stored in routing table and can adjust its congestion window size accordingly. When a node detects new link it broadcast link formation packet (LFP), containing link capacity information and is stored in routing table of routing protocol, therefore TCP don’t need to invoke slow start mechanism when new link is detected and communication is resumed. Before communication TCP sender get information about link capacity from routing table of next node involved in communication and adjusts its congestion window size.

It has been mentioned above that congestion could be avoided through the use of LCD operation of NFBTCP. Through LCD operation we determine the link capacity of the newly established link. In NFBTCP nodes are made responsible to inform TCP about the new link capacity of the established link. .

A7. Link Formation Packet (LFP)

Link formation packet contains information about the link capacity. This packet is broadcast in link capacity detection (LCD) operation of NFBTCP. To get information about link capacity in link capacity detection operation link formation packet is broadcast and is stored in routing table of routing protocol. Routing protocols with cache can add an extra parameter to link formation packet and can store in its cache. Every time when a new link is detected link formation packet is updated with link capacity information by broadcasting it in link capacity detection operation.

A8. Out-of-Order Delivery Notification

It has been mentioned before that mobile ad-hoc network suffers with frequent topology change. TCP is known for in order delivery to the receptionist; however no direct effective mechanism is known which can be used to deal with the lost or dropped packet. This is of particular interest in the context of mobile ad-hoc network, where packet could be dropped due to link or route failure. NFBTCP uses some of the known benefits to deliver solution of Out-of-order Delivery problem in mobile ad-hoc network environment.

In order to deal with out-of-order delivery of data packets a buffer is created in between TCP and the receiving node. Therefore rather than delivering packet as it arrives all the packets of a single transmission are stored in the buffer. Likewise, TCP will be modified so that it can send one acknowledgement for the complete delivery of all the data packets of a single transmission then single acknowledgement for a single packet. If a lost packet is detected a Buffer Update Packet (BUP) will be send to the sender pointing the missing packet. This lost packet can easily be identified either via sequence number or broadcast ID. Please note this information is normally included or assigned by the routing protocol of MANET. It is the responsibility of the sending node to re-broadcast the missing packet as identified by the TCP back to the receiver side using the same route as for the previous packet.

Buffer Update Packet (BUP): Buffer Update Packet is sent whenever an out-of-order delivery is received at TCP buffer side. This serves an additional purpose can also be used to update the intermediate node about the availability of the other nodes in between the source and the receiving node. Needless to mention such information is always fruitful in the context of mobile ad-hoc networking environment as such information could also be used for any other possible communication by the intermediate nodes.

IV. DISCUSSION AND CONCLUSION

NFBTCP offer modification to the TCP scheme of generation acknowledgement for the sender nodes of a packet. Rather it stresses that any node which receive a packet over a link is responsible of sending acknowledgement packet for the sender node. In NFBTCP routing protocol which is used alongside TCP is made responsible of sending updated notification whenever a link failure is detected. This could stop sender to send any further packet using the broken route. Link Failure Update Packet contains details of failed route or link. This packet is broadcasted by an intermediate route which is in between the sender and the receiver of an active communication. In NFBTCP link capacity information is available at routing table of the routing protocol. Additional parameter could be added in the existing specification of the routing protocol to store such information. Link formation packet contains information about the link capacity. This packet is broadcast in link capacity detection (LCD) operation of NFBTCP. In order to deal with out-of-order delivery of data packets a buffer is created in between TCP and the receiving node. Therefore rather than delivering packet as it arrives all the packets of a single transmission are stored in the buffer. Buffer Update Packet is sent whenever an out-of-order delivery is received at TCP buffer side.

In conclusions, NFBTCP focuses on some of the main problems of TCP in mobile ad-hoc network. We have utilized mixture of some of the existing new mechanisms to model our solution. Our next step is to further expand our proposed solution and its implementation in a practical environment. We shall also evaluate NFBTCP performance both on its own and against some of the earlier proposed solutions. We believe our proposed solution could yield an efficient TCP based solution for mobile ad-hoc network which can overcome weaknesses of the existing schemes of this area.

REFERENCES

1. T. D. Dyer and R. V. Boppana, "A comparison of TCP performance over three routing protocols for mobile ad hoc networks," in ACM International Symposium on Mobile Ad-Hoc Networking and Computing: MobiHoc 2001.
2. K. Sundaresan, V. Anantharaman, H.-Y. Hsieh, and R. Sivakumar, "ATP: A reliable transport protocol for ad-hoc networks," in ACM International Symposium on Mobile Ad-Hoc Networking and Computing 2003.
3. S. Kopparty, S. V. Krishnamurthy, M. Faloutsos, and S. K. Tripathi, "Split TCP for mobile ad hoc networks," in IEEE Global Telecommunications Conference, 2002.
4. K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash, "Feedback based scheme for improving TCP performance in ad-hoc wireless networks," in International Conference on Distributed Computing Systems, 1998.
5. J. Liu and S. Singh, "ATCP: TCP for mobile ad hoc networks," in IEEE Journal on Selected Areas in Communications, vol. 19, pp. 1300-1315, 2001.
6. G. Holland and N. Vaidya, "Analysis of TCP performance over mobile ad hoc networks," in Wireless Networks, vol. 8, pp. 275-288, 2002.
7. S. Ramanathan and M. Steenstrup, "A survey of routing techniques for mobile communication networks", in Mobile Networks and Applications (1996) 89-104.
8. V. T. Raisinghani and S. Iyer, "Cross-layer design optimization in wireless protocol stacks," in Computer Communications, vol. 27, pp. 720-724, 2004.
9. S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-layer design for wireless networks," in IEEE Communications Magazine, vol. 41, pp. 74-80, 2003.
10. G. Holland and N. Vaidya, "Analysis of TCP performance over mobile ad-hoc networks" – Part II: Simulation details and results, Technical report TR99-005, (1999).
11. L. Chen, S. H. Low, and J. C. Doyle, "Joint congestion control and media access control design for ad hoc wireless networks," in IEEE INFOCOM, 2005.
12. X. Yu, "Improving TCP performance over mobile ad hoc networks by exploiting cross-layer information awareness," in Mobile Computing and Networking, 2004.
13. Z. Fu, X. Meng, and S. Lu, “How bad TCP can perform in mobile ad hoc networks” in ISCC Computers and Communications, 2002.
14. R. Cáceres and L. Iftode, "Improving the performance of reliable transport protocols in mobile computing environments", IEEE Journal on Selected Areas in Communications 1995.
15. H. Balakrishnan and R. Katz, "Explicit loss notification and wireless web performance", in IEEE Globecom Internet Mini-Conference, 1998.
16. B.S. Bakshi, P. Krishna, D.K. Pradhan and N.H. Vaidya, "Improving performance of TCP over wireless networks", in International Conference on Distributed Computing Systems 1997.
17. T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," in Wireless Communications and Mobile Computing, vol. 2, pp. 483-502, 2002.

Article received: 2010-11-27