Avoid link Breakage in On-Demand Ad-hoc Network Using Packet's Received Time Prediction

Naif Alsharabi, Ya ping Lin, Waleed Rajeh
College of Computer & Communication
Hunan University – ChangSha 410082
Sharabi28@hotmail.com , yplin888@hotmail.com

KEYWORDS

Ad-hoc Network, AODV, PRT Algorithm, Link breakage.

ABSTRACT

Most existing on-demand mobile ad hoc network routing protocols continue using a route until a link breaks. During the route reconstruction, packets can be dropped, which will cause significant throughput degradation. In this paper we introduce the Packets Received Time (PRT) to predict the link state, that is, if a particular node is soon to go out of transmission range. The Prediction Algorithm together with our PRT approach enhances the performance of the existing Ad-hoc On-Demand Distance Vector (AODV) protocol. The new approach is compared with original AODV in CBR and TCP traffics using various scenarios. The simulation results showed that our scheme is more efficient, reliable and improves throughput of the Ad-hoc network.

INTRODUCTION

In most of the current ad-hoc routing protocols like (DSDV, DSR, TORA, DSR etc.) a node will keep using the route until the link is broken. It then has to discover a new route to the destination. During this discovery time the packets are lost and it will cause significant throughput degradation. When the network traffic requires real time delivery (such as voice, or video), dropping data packets at the intermediate nodes can be costly.

In this paper, we propose an algorithm that utilizes PRT (Packet's Received Time) to predict the signal power of the link state and find out if the route is going to break. Our scheme aims at modifying mobile ad-hoc network (MANET) reactive routing protocol (AODV), to give it a proactive behavior to improve its performance. Under our proposed scheme, route maintenance decisions are based on predicted values of 'link-breakage times' (when the next-hop node will move out of transmission range). If a link is about to break, proactive discovery of new routes to all destinations using the next hop node depends on the history of traffic to that destination. In recent years few algorithms and protocols tried to improve performance by using link state information. Associatively-Based Routing (ABR) proposed by Toh [1] favors routes with longer-lived links according to the associatively of the incident nodes. He proposed a link state prediction model based on the knowledge of mobile nodes' position [1].

Flow Oriented Routing Protocol (FORP) [2] also uses a mobile node's position information that is provided by Global Position System (GPS) to predict link state. The destination node can determine the route expiration time based on the link prediction in the route. It will inform the source node when a link is predicted to be broken, then the source can select the most reliable route to send the rest of the packets.

The more recent preemptive routing protocol proposed by Goof [3] uses signal power strength. It initiates early route discovery by detecting that a link is likely to be broken and builds an alternative route before the link fails. However, their simulation results show that the increase in overhead (The preemptive ratio is used for defining a preemptive zone that is adjacent to the signal strength threshold. Because of is a constant, it implies that for mobile nodes with different relative speeds, the size of the preemptive zone is the same.) could be as high as three times the overhead of the original DSR protocol. The link availability prediction requires that two nodes maintain their movement patterns during the prediction time.

Several papers have proposed the Probability model for the link availability. But the GPS and signal strength methods presented in [4] [5] [6] use physically measured parameters to predict the link status. The node with GPS can know its position directly, but the GPS system currently is not a standard component of mobile devices and the signal is too weak to be received in the metropolitan area and indoors.

This paper concentrates on the PRT prediction approach in ad hoc networks to reduce the data packets that would have been dropped because of link failures. As seen above, in most existing protocols, a mobile host will keep using the route until the link is broken. Our proposed scheme will use power measurement of received packets to predict the topological change in order to rebuild a route prior to the link breakage, thus avoiding the data packets being dropping. Generally, a link failure happens when two mobile nodes A and B

Proceedings 19th European Conference on Modelling and Simulation
Yuri Merkuryev, Richard Zobel, Eugène Kerckhoffs © ECMS, 2005
ISBN 1-84233-112-4 (Set) / ISBN 1-84233-113-2 (CD)

move out of their radio transmission ranges. Node B monitors the packets coming from A, predicts the link breakage time of link A-B, and then sends a warning message to the source node of this active route. The source node can rebuild a new route before the link breaks. The simulation results show that our PRT algorithm can increases the packets delivery ratio and reduces the number of drop's packets due to link failure.

The rest of the paper is organized as follows:
Section 2 discusses the AODV concepts; section three explains our PRT approach, whereas section four gives the simulation method and presents the simulation results. The paper ends with the conclusion and future work to be done in section five.

2. AODV CONCEPT

In this section, we present the conceptual details of the AODV protocol. AODV nodes use four types of messages to communicate among each other. Route Request (RREQ) and Route Reply (RREP) messages are used for route discovery. Route Error (RERR) messages and HELLO messages are used for route maintenance. The following sections describe route determination and route maintenance in greater detail.

2.1 PRT Route Construction

Our algorithm does not require any modification to the AODV's *RREQ* (route request) propagation process. When a source needs to initiate a data session to a destination but does not have any route information, it searches a route by flooding a *RREQ* packet. Each *RREQ* packet has a unique identifier so that nodes can detect and drop duplicate packets.
An intermediate node upon receiving a non-duplicate *RREQ* records the previous hop and the source node information in its route table. It then broadcasts the packet or sends back a ROUTE REPLY (*RREP*) packet to the source if it has a route to the destination. The destination node sends a *RREP* via the selected route when it receives the first *RREQ* or subsequent *RREQ*s that traversed a better route (in AODV for instance, fresher or shorter route) than the previously replied route, when the route established the source start send the packet's to the destination through shorter route .

The PRT structure and L_Prediction are established during *RECV* (received packet's procedure) and *RREP_ACK* phase that we will explain more in section3.

2.2 Routing Maintenance and L_prediction

Data packets are delivered through the primary route unless there is a route disconnection. When PRT detects a packet's received time is bigger than the route life time and does not receive hello packets for a certain period of time the L_prediction send *RERR* to the source to initiate a route rediscovery. The reason for reconstructing a new route is to build a fresh and optimal route that reflects the current network situation and topology. The L_prediction also mark the disconnect route and delete it from the packet header. Data packets therefore can be delivered through fresh routes and are not dropped when route breaks occur.

2.3 Packets Receiving & Packet's Sequence Numbers

The destination node (DN) continues receiving packet's until the link is broken. The *DN* receives different packet's type from the source node (SN) or upstream node. Each destination (node) maintains a monotonically increasing sequence number, which serves as a logical time at that node. Also every route entry includes a destination sequence number which indicates the "time" at the destination node when the route was created. The protocol uses sequence numbers to ensure that nodes only update routes with "newer" ones.

All *RREQ* messages include the originator's sequence number and its (latest known) destination sequence number. Nodes receiving the *RREQ* add/update routes to the originator with the originator sequence number assuming this new number is greater than that of any existing entry. If the node receives an identical *RREQ* message via another path the originator sequence numbers would be the same, so in this case the node would pick the route with the smaller hop count. If a node receiving the *RREQ* message has a route to the desired destination then the sequence numbers used to determine whether this route is "fresh enough" to use as a reply to the route request. *RREQ* messages, *RREP* messages also include destination sequence numbers. This is so nodes along the route path can update their routing table entries with the latest destination sequence number.

3. PRT and prediction algorithm

Two Ray Ground reflection approximations are used as radio propagation model in [9]. The Two Ray Ground model uses formula (1) to calculate signal strength at the receiver's end.

$$\left[P_r = \frac{P_t * G_r * (h_t * h_r)^2}{d^4} \right] \ldots\ldots\ldots\ldots (1)$$

Where: P_r is the received signal power, P_t is the transmitted signal power, G_t is the transmitter antenna gain, G_r is the receiver antenna gain, h_t is the transmitter antenna height, h_r is the receiver antenna height, It is assumed that P_t is a constant. Assume that the ground is flat to remove dependence of h and d values on the geography of the simulation area. So equation above can be simplified under the conditions of ad hoc wireless network simulation to:

$$\left[P_r = k \frac{P_t}{d^4} \right] \quad \text{……………………………} (2)$$

Where k is constant

$$k = G_t * (h_t^2 * h_r^2) \quad \text{……………………………} (3)$$

This equation shows that the signal power at the receiver node has relation $1/_{d^4}$ with the distance between the sender node and receiver node.

As we mentioned before, GPS and signal strength methods use physically measured parameters to predict the link status. GPS currently is not a standard component of mobile devices and the signal can be too weak to be received. Supposing the route has already been established and the first packet delivered, our algorithm starts recording packet received times, and based on this data, predicting link breakages using the following formula:

$$R_{pt} = C_t - S_{pt} \quad \text{……………………………} (4)$$

Where R_{pt} is the packet received time for the current packet, C_t is the current time and S_{pt} is the packet's send time. We suppose to receive 3 packets on destination node to predict the link state. In this case we repeat equation (4) on the future packets and save it in the table using equation (5). In this case we will increment the packet flag *p_flg* by one and save the packet receive time on the receive table as flowing:

$$T_{rpt}[p\_flag] = R_{pt} \quad \text{……………………………} (5)$$

The packet received time average over all neighborhoods and all time calculated using the following formula:

$$P_{rt} = \sum_{t=0}^{T} \sum_{n0}^{N} \frac{C_t}{T_{rpt}[p\_flag] * LL_t} \quad \text{…………} (6)$$

Where P_{rt} is packet received Time average on destination node, we are using the C_t is the current time defined on AODV original protocol calculated during transmitting and receiving packets, LL_t is link life time, T total time arrives at the destination, and N number of hop.

Substituting (6) into equation (2) the received signal power on the distinction node calculated as following:

$$\left[P_r = k \frac{P_{rt}}{d^4} \right] \quad \text{……………………………} (7)$$

We added PRT procedure to AODV protocol, in this procedure when the destination node received the first packet, PRT start save packet time on the received packet time table, increment the packet flag, calculate packet signal power and wait for the next packet from the upstream and repeat equation (5), (6), (7) to next packet and compare it with previous packet that is already on the table.

If the current packet's signal power is greater than the pervious packet's signal power, that means the nodes are moving closer to each other otherwise if the current packet signal power is equal to the pervious packet signal power that means the nodes are quiescence so the packet flag will be zero and do not need prediction algorithm.

On the other hand if current packet signal power is weaker than the previous packet signal power, prediction algorithm maintenance marks the current route as idle to delete it from the packet header when a new route is established and send RERR upstream to locally maintain the route, or to the source node to establish RREQ to find a fresh and optimal route to the destination that reflects the current network situation and topology.

In the implementation, each destination nodes will keep an array as showed in (5) of signal info objects. Each table holds three packets with information such as signal power strength and reception time for the same neighboring mobile nodes. When node B receives packets from node A, it updates its table array according to:

$$P_3 \le P_2 \le P_1 \quad and \quad T_3 \le T_2 \le T_1 \quad \text{…..} (8)$$

When two mobile nodes are moving closer, the latest signal power strength will be greater than the previous one. In this case, we set P_1 to the latest signal power value and set P_2 and P_3 to zero, no prediction is necessary.

4. Simulation and Analysis

All simulations were run using the NS-2 simulator [13], [14] and numerous simulations were chosen to illustrate the performance advantage gained by using AODV_PRT over AODV. The simulation experiments can be classified broadly as CBR (UDP) based simulations and simple formal model of TCP over MAC 802.11 based simulations. The routing protocols were tested using both CBR and TCP traffic to get a more complete picture of their performances. Both the CBR and TCP based simulations were run with two mobility models.

The simulations using RW (random waypoint) model were run in a 1500m by 300m area with 20 nodes under varying conditions of mobility and load. The

communication model consisted of 8 CBR connections, with a packet size of 512 bytes for each set of simulations. All statistics were based up on 10,000 data packets and the rate of sending is 0.25, 0.5, 0.75 and 1second. Our simulations were conducted by varying both maximum velocity and pause time.
Maximum velocity varied as 1, 4, 8, 12,16, 20 m/s
Pause time varied as 0, 50, 100, 150, 200 seconds

Figure 4.1 show that the End-to-End delay for AODV_PRT has longer delays than AODV for high mobility. The reason for that we measure delays for data packets that survived to the destination. AODV_PRT delivers more packets, and those packets that are delivered in AODV_PRT but not in AODV, our explanation when route going to break the AODV_PRT possibly uses longer alternate paths to deliver packets that are dropped in AODV.

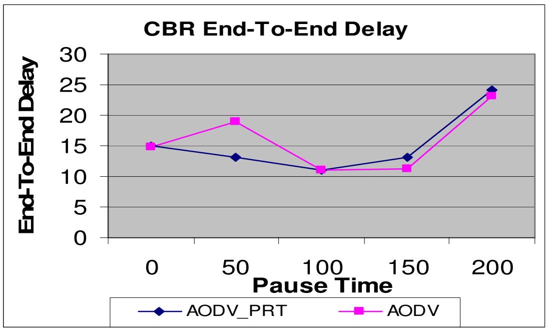

Figure 4.1 End-to-End delays vs. pause time

Figure 4.2 shows that AODV_PRT delivers more packets on high mobility, when the nodes are moving fast on the topology area, that mean increase the possibility of link breakage. Figure 4.2 shows the AODV with PRT Algorithm will detect the link breakage and delivers those packets lost on AODV protocol.

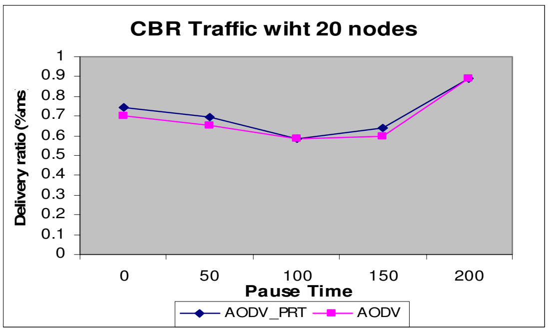

Figure 4.2 Packet delivery ratios vs. pause time

Figure 4.3 shows that the AODV_PRT losses fewer packets than AODV that's increase the packets delivery thru the AODV_PRT than AODV.

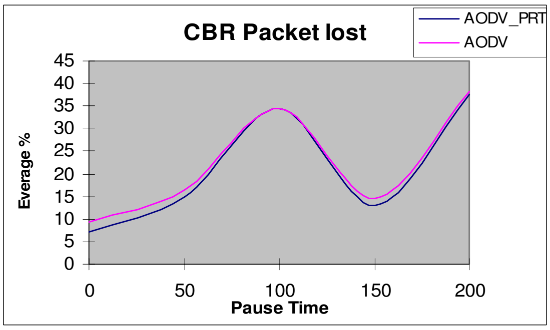

Figure 4.3 CBR packets lost

Figure 4.4 shows that the AODV and AODV_PRT have delivered almost the same amount of packets in low mobility and when the nodes are moving fast AODV_PRT deliverers more packets. AODV_PRT has more amounts of control messages. The hop count obtained with CBR traffic is a true measure of the average hop count of all active routes in the simulation, as the traffic source is independent of the network condition, while the hop count obtained with TCP traffic is not. This is because, in the absence of congestion, the rate of TCP transmissions is very sensitive to the number of hops, because the rate depends on the mean round trip time (*rtt*) of each connection, which is largely dependant on the number of hops.

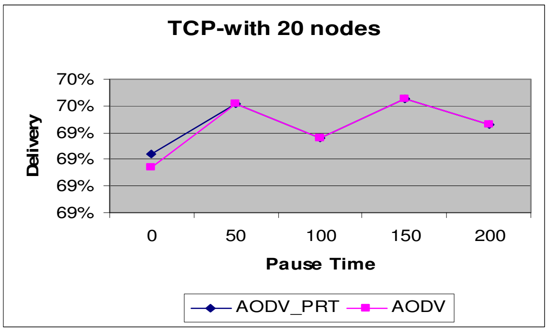

Figure 4.4 TCP Packet delivery ratios vs. pause time

Hence at lower hop counts, TCP transmits at a very high rate, while the rate rapidly drops at higher hop counts. Thus, the average hop count in TCP tends to be similar for all simulations just as the average hop count across all CBR simulations are comparable. Since TCP operates as a feedback system, TCP has a lower average hop count than the average hop count with CBR traffic for the same mobility scenario [6], [7], [8] and [13].

Figure 4.5 shows the delivery packets ratio for TCP && CBR with deferent pause time, figure 4.5 is not compare between TCP and CBR. It shows the packets delivery for each traffic with AODV original and AODV_PRT.

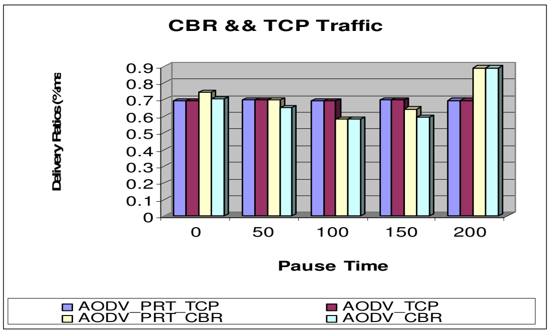

Figure 4.5 Packet delivery ratio with CBR && TCP vs. pause time

Figure 4.6 shows that the End-to-End delay for AODV_PRT and AODV have the same end to end delay. But AODV_PRT deliver more packets.

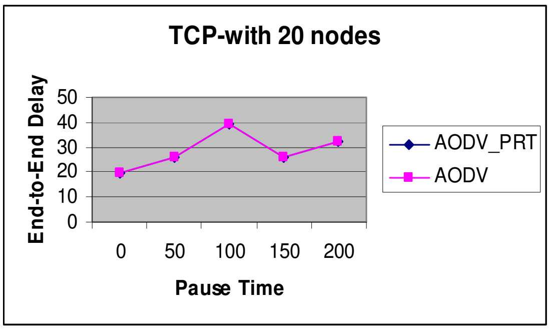

Figure 4.6 TCP End-To-End Delay vs. pause time

Figure 4.7, 4.8 and table [2] are shows that the AODV have More drop packets than AODV_PRT when no routes because the AODV_ PRT detect the link break and discover a new route to continue send that's packets dropped in AODV, AODV_PRT more efficient to deliver the data and avoid the packets dropping that's mean its more efficient than AODV.

Table [2] show that the total data dropped and the data dropped because there is no route to the destination during the simulation experiments for the original AODV and AODV_PRT.

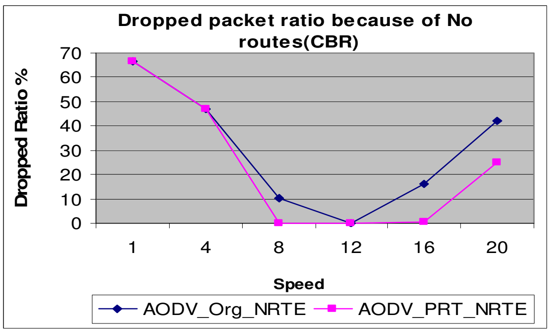

Figure 5.7 Dropped packets on CBR

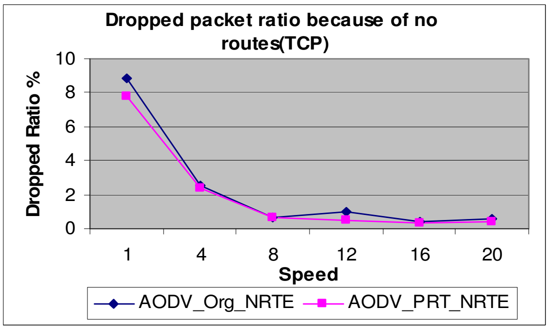

Figure 5.8 Dropped packets on TCP

Protocol/Reason		Speed				
AODV Original		4	8	12	16	20
CBR	Total Dropped Pk	273	277	148	278	376
	Dropped No route	128	28	0	45	159
TCP	Total Dropped Pk	239	2156	402	937	1057
	Dropped No route	6	14	4	4	6
CBR	No route CBR %	46.886	10.108	0	16.187	42.287
TCP	No route TCP %	2.5104	0.6493	0.9950	0.4268	0.5676
AODV_PRT		4	8	12	16	20
CBR	Total Dropped Pk	273	271	148	332	511
	Dropped No route	128	0	0	1	128
TCP	Total Dropped Pk	255	2156	402	2094	995
	Dropped No route	6	14	2	7	4
CBR	No route CBR %	46.886	0	0	0.3012	25.048
TCP	No route TCP %	2.3529	0.6493	0.4975	0.3342	0.4020

Table [2]: Packet Dropped TCP & CBR

5. Conclusions and future work

Prediction algorithm is one of the best approaches to avoid link breakage; it has been widely used in schemes aimed at improving performance of ad-hoc networks [4], [5], [6], [7], [9]. As reviewed in this paper, most of this work depends on node density, radio transmission range, and GPS and signal strength. But the GPS and signal strength methods both use physically measured parameters to predict the link status. The performance could be further improved using the received packet signal. This paper has given our new method to improve the performance of ad-hoc network as the following:

- Our simulation compared between CBR traffic OADV original protocol with AODV_PRT protocol and TCP AODV original protocol with AODV_PRT protocol. We were not compared between CBR traffic and TCP traffic performance on simulation results.

- The CBR simulation shows that AODV_PRT delivers more packets, and those packets that are delivered in AODV_PRT but not in AODV, take alternate and possibly longer delay more than AODV,

- The TCP simulation shows that AODV and AODV_PRT have delivered almost the same amount of packets in low mobility , and AODV_PRT deliverers more packets than AODV in high mobility,

- AODV_PRT has more amounts of control messages. The hop count obtained with CBR traffic is a true measure of the average hop count of all active routes in the simulation, as the traffic source is independent of the network condition, while the hop count obtained with TCP traffic is not,

- Compared with original AODV, our simulation experiments show that our approach with CBR and TCP traffic is more beneficial, delivered more packets, less lost packets and packets dropped.

More still remains to be done to improve the performance on ad-hoc network protocols. Reduce overhead, control messages and also implementing our scheme in the real world scenario.

References

1. C.-K Toh. "Associativity-Based Routing for Ad-Hoc Networks", Wireless Personal Communications Journal, Special Issue on Mobile Networking and Computing Systems, vol. 4, no. 2, pp. 103-139, March 1997.

2. W. Su and M. Gerla, "IPV6 Flow Handoff in Ad-Hoc Wireless Networks Using Mobility Prediction", Proc. of IEEE Global Communications Conference, Rio de Janeiro, Brazil, pp. 271-275, December 1999.

3. T. Goff and N. B. Abu-Ghazaleh etc., "Preemptive Routing in Ad Hoc Networks", Proc. of the Seventh Annual International Conference on Mobile Computing and Networking, Rome, pp. 43-52, Italy, July 2001.

4. A. B. McDonald and T. Znati, "A Path Availability Model for Wireless Ad-Hoc Networks", Proc. of IEEE Wireless Communications and Networking Conference, New Orleans, LA, pp. 35-40, September 1999.

5. S.M. Jiang, D.J. He and J.Q. Rao, "A Prediction-Based Link Availability Estimation for Mobile Ad-Hoc Networks", Proc. Of IEEE Infocom, Anchorage, AK, pp. 1745-1752., April 2001

6. C. E. Perkins. Ad Hoc Networks, USA, Addison-Wesley, Dec. 2000.

7. S. R. Das, C. E. Perkins and E. M. Royer, "Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks", Proc. of the IEEE Conference on Computer Communications.

8. Daehyoung Hong and Stephen Rappaport, "Traffic Models and Performance Analysis for Cellular Mobile Radio Telephone Systems with Prioritized and Non-Prioritized Handoff Procedures", IEEE Transactions on Vehicular Technology, Vol. VT-35, No.3, pages 77-92

9. He Dajing, Jiang Shengming and Rao Jianqiang, "A Link Availability PredictionModel for Wireless Ad Hoc Networks", Proceedings of the International Workshop on Wireless Networks and Mobile Computing, D7-D11, Taipei, Taiwan, April 2000.

10. B. Narendran, P. Agrawal and D. K. Anvekar, "Minimizing Cellular HandoverFailures without Channel Utilization Loss", Proceedings of IEEE GlobalCommunications Conference, pages 1679-1685, vol. 3, December 1994.

11. The Network Simulator - ns-2 http://www.isi.edu/nsnam/ns

12. Kevin Fall and Kannan Varahan, editors. NS Notes and Documentation. The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC, November 1997.

13. David A. Maltz, Josh Broch, Jorjeta Jetcheva and David B. Johnson, "The Effects of On-Demand Behavior in Routing Protocols for Multihop Wireless Ad Hoc Networks",IEEE Journal on Selected Areas in Communications Special Issue on Mobile and Wireless Networks, pages 1439-1453, August 1999.