The 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'07)

DESIGN A MULTI-PATH ROUTING ALGORITHM IN AD HOC NETWORKS IN ORDER TO IMPROVE FAULT TOLERANCE

Shabnam Vahedi
Iran University of Science and Technology
Tehran, Iran

Maryam Mohseni
Iran University of Science and Technology
Tehran, Iran

Amir Darehshoorzadeh
Iran University of Science and Technology
Tehran, Iran

ABSTRACT

In this paper an on-demand multipath source routing scheme in mobile ad hoc networks called “Split MultiPath – Dynamic Source Routing” (SMP-DSR) is proposed. The proposed protocol establishes and utilizes multipath maximally disjoint routes in order to improve fault tolerance and to provide load balancing via multipath routing. Based on simulation results, our protocol performs well over a variety of environmental conditions such as host density and movement rates, particularly it achieves a high rate of successful packet delivery in all cases.

I. INTRODUCTION

Mobile ad hoc networks consist of a collection of wireless mobile nodes which dynamically exchange data among themselves without the reliance on a fixed base station or a wired backbone network. In such networks, nodes are typically distinguished by their limited power, processing, and memory resources as well as high degree of mobility. Due to the limited transmission range of wireless network nodes, multiple hops are usually needed for a node to exchange information with any other node in the network. Thus routing protocols play an important role in ad hoc network communications.

On-demand routing is the most popular routing approach in ad hoc networks. Instead of periodically exchanging routing messages in proactive routing protocols which brings in excessive routing overhead [1,5], on-demand routing algorithms discover routes only when a node needs to send data packet to a destination and does not have any route to it. Most of the existing on-demand routing protocols (for example, Dynamic Source Routing (DSR) and Ad hoc On-demand Distance Vector (AODV) build and rely on single path for each data session. So route recovery process is needed after each route failure, which causes to lose transmitted data packets, in such protocols.

Multipath routing allows the establishment of multiple paths between a single source and single destination node. It is typically proposed in order to increase the reliability of data transmission (i.e., fault tolerance) or to provide load balancing [6, 7, 8]. In such protocols, traffic is not distributed into multiple paths; only one route is primarily used and alternate paths are utilized only when the primary route is broken.

An on-demand multipath source routing protocol called Split Multipath Dynamic Source Routing (SMP-DSR) is proposed in this paper that builds maximally disjoint paths. In case of necessity, data receiving control process of this scheme splits traffic into multiple routes to use network resources efficiently. DSR [2] and SMR [3] protocols have been considered in design of the proposed protocol.

The rest of this paper is organized as follows. Section II describes the protocol mechanism in detail. The simulation results are presented in section III in order to evaluate the performance of the protocol and conclusion remarks are made in section IV.

II. SPLIT MULTIPATH DYNAMIC SOURCE ROUTING

A. Route Discovery

In Split Multipath Dynamic Source Routing (SMP-DSR) each mobile host maintains route caches, in which it caches routes that it has learned. When one host wants to send a packet to another host, the sender first checks for a source route to a destination. If a route is found, the sender uses this route to transmit the packet. Otherwise the sender attempts to discover one using the route discovery process (route request/reply cycles). While waiting for route discovery to complete, the host continues normal processing. The host buffers the original packet in order to transmit it once the route is discovered. Each entry in the route cache is deleted after an expiration period.

1) Route Request Process

In this process, the sender broadcasts Route Request (RREQ) packets which may be received by those hosts within wireless transmission range of it.

Each RREQ packet contains source ID and a sequence number which are used to uniquely identify the packet. It also contains a route record, in which is accumulated a record of the sequence of hops taken by the RREQ packet as it is propagated through the ad hoc network.

When any node other than the final destination receives a RREQ packet for the first time, it appends its ID to the route record of the packet and re-broadcasts it. Since dropping all duplicate RREQs only generates multiple paths that are mostly overlapped, whenever a duplicate RREQ packet reaches an intermediate node, instead of dropping it, the receiving node forwards duplicate packet that traversed through a different incoming link, and whose hop count is not larger than previous received RREQs.

In this approach intermediate nodes are not allowed to send RREPs back to the source even when they have valid routes to the destination in their caches. In this way, destination will know the entire path of all available routes, therefore it can select maximally disjoint routes to prevent certain nodes from being congested, and to utilize the available network resources efficiently. This approach has a disadvantage of propagating more RREQ packets.

Thus the RREQ propagates through the ad hoc network until it reaches the target host of the route discovery.

2) *Route Reply Process*

Whenever the first RREQ packet reaches the target host, route reply process is started. When the destination receives the first RREQ, it records its path and sends first Route Reply (RREP) packet, which contains the shortest delay route, back to the source. The route reported by RREP packet is cached by intermediate nodes and all nodes which receive RREP packet in promiscuous mode.

After transmitting the first RREP, destination node waits a predefined duration of time to receive more RREQs. Then it selects two routes which are maximally disjoint to the first received route. In order to choose maximally disjoint routes, the destination node does as follows: If there are more than two routes that are maximally disjoint to the first route, the routes with the shortest hop distance are chosen. If there still remain multiple routes that meet the condition, the quickest of them is selected. Then the destination sends the selected RREPs back to the source.

B. *Data Transfer Process*

In our algorithm each node has two caches. All of the routes requested and received by the intermediate nodes and nodes, which are in promiscuous mode, are cached in the first cache. This cache contains routes which are applied for transmitting control messages. Moreover, the cached routes in the first cache are alternatives of the active routes which are in use at the same time. Active routes are cached in the second cache. Each node sends data packets by the routes in its second cache.

When the sender receives the first RREP, it appends the reported route to both of its caches and starts to transmit buffered data packets by the reported route. Other routes reported later, are only cached in the first cache.

C. *Data Delivery Control Process*

When the destination receives the first data packet, a timer is set and a counter starts to count the received data packets in the destination node.

After a predefined duration of time, the number of received data packets in the destination node and number of transmitted data packets from the source node are compared to each other. Then the timer and counter will be reset. If the number of received packets is less than the number of transmitted packets, a control packet called ‘data delivery control packet’ will be produced by the destination and sent to the source node.

When the control packet reaches the source, the source node recognizes that using the first route alone is not efficient enough to transfer data packets. So it appends a new route from its first cache to its second cache, in case of the existence of any related route in the first cache, and then sends data packets via both of them. Drawback of this scheme is out of order delivery and re-sequencing burden on the destination. However, the cost-effective reordering buffers are easily implementable. Furthermore it is fairly difficult to obtain the network condition such as available bandwidth in ad hoc networks to apply more sophisticated schemes.

Since the probability of validation of cached routes in the first cache will be decreased as times, by using them in this way, available network resources will be utilized efficiently. Furthermore the number of received data packets by the destination node will be increased.

The periodic data delivery control process repeats until the end of data exchange.

D. *Route Maintenance Process*

A route may be broken due to fading effects, battery drainage, congestion or packet collisions which cause to disrupt routing functionalities.

In SMP-DSR the route invalidation is recognized by the immediate upstream node of the broken link by receiving a link layer feedback from IEEE 802.11 [4], after delivering data packets to the next hop of the route. In such a case, the recognizing node removes all routes containing the disconnected link, from its caches. Then in order to prevent data loss, this node searches its caches to find a valid route to the destination and forwards the data packet through this new path, in case of existence of the related route. Otherwise the data packet will be dropped.

Then, a Route Error (RERR) packet containing the route to the source, and immediate upstream and downstream nodes of the broken link, is produced by the recognizing node and will be sent to the upstream direction of the route. Source node, intermediate nodes and all nodes which receive RERR packet in promiscuous mode, remove every route that uses the broken link from their caches.

Whenever a route is removed from the second cache of a node, a new route to the related destination, from first cache (if there is anyone), is appended to second cache, to prevent ineffective routing.

In SMP-DSR, route recovery process is initiated only when all of the routes to the destination of data packets are broken. In this way, less control overhead will be produced.

III. PERFORMANCE EVALUATION

A. *Simulation Environment*

In this section the performance of DSR and SMP-DSR protocols is compared and evaluated.

The simulator was implemented within the Global Mobile Simulation (GloMoSim) library [9]. In this simulation a network of randomly placed mobile hosts within a 1000m×1000m area was modeled. Each node had a radio propagation range of 250 meters and channel capacity was 2 Mb/s. Each run executed for 300 seconds of simulation time.

A free space propagation model with a threshold cutoff was used in this simulation. IEEE 802.11 was used as the medium access control protocol. Random waypoint model was used as the mobility model. The sources and the destinations were randomly selected. A traffic generator was developed to simulate constant bit rate (CBR) sources. There were twenty data sessions, which generated 25 data packet per second. Each data session had 300 data packets to send and the size of data payload was 1024 bytes.

B. Results and Analysis

In order to evaluate and compare the performance of DSR and SMP-DSR protocols two scenarios have been considered. In both of them the throughput of each protocol in packet delivery ratio has been presented. Packet delivery ratio is the ratio of the number of data packets successfully received by the destinations, and the number of data packets sent from the sources. The control overhead in routing load is also presented. Routing load is obtained by dividing the number of control packets received by every node in the network, by the sum of number of received control packets by every node and the number of data packets correctly received by the destinations.

In the first scenario, the effect of network density in different speeds on each protocol is considered. In this scenario the simulation area is constant (1000m×1000m) and the number of mobile hosts and their speeds are variable. The pause time value is also constant and is equal to 1 second.

Figures 1, 2 and 3 illustrate the performance of each protocol in this scenario. It can be observed from the packet delivery results that SMP-DSR considerably outperforms DSR. This improvement however, is usually gained by more control overhead in SMP-DSR. These figures report that as the number of mobile hosts of the network increases, more nodes are placed in radio propagation range of each node. So building maximally disjoint routes in each route discovery process is more probable which causes to improve the fault tolerance and packet delivery ratio in each protocol. As the number of mobile hosts in radio propagation of each node increases, more control packets are propagated in each route discovery process which cause to increase the control overhead of the network.

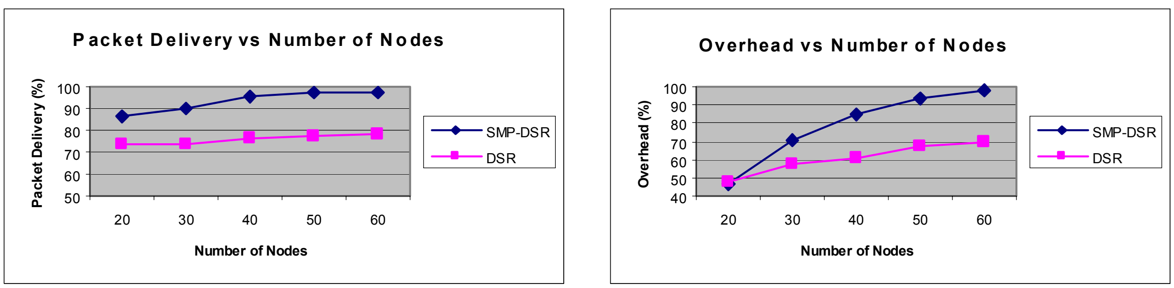

Figure 1: Packet Delivery Ratio and Control Overhead (min and max speeds are set to zero and 10m/s, respectively)

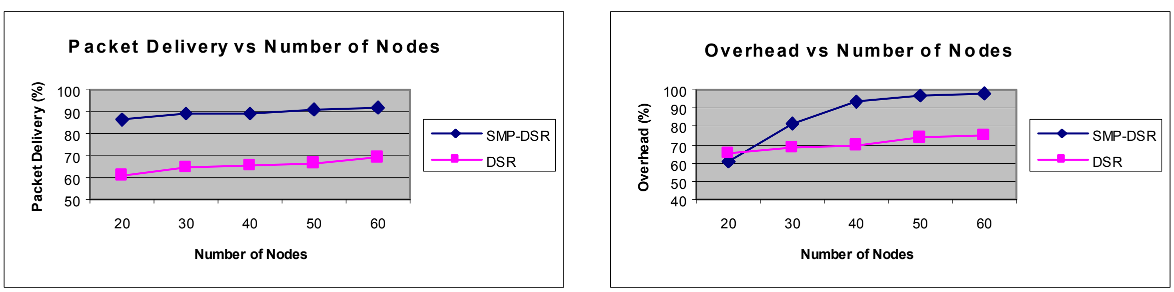

Figure 2: Packet Delivery Ratio and Control Overhead (min and max speeds are set to zero and 30m/s, respectively)

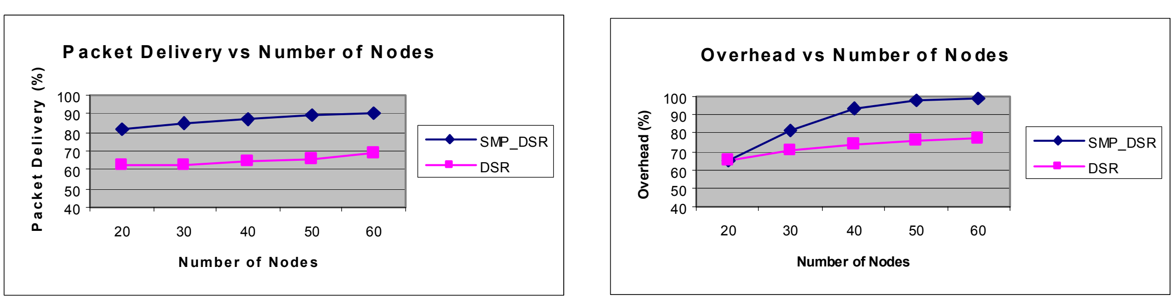

Figure 3: Packet Delivery Ratio and Control Overhead (min and max speeds are set to zero and 50m/s, respectively)

By comparison among figures 1, 2 and 3, it is observed that the speed increase of mobile hosts causes to decrease the packet delivery and increase the control overhead in each protocol.

The fast mobility of nodes brings in route breaks and route recovery process. As the SMP-DSR is a multipath routing algorithm and does not allow replying from cache of intermediate nodes, it generates more control packets while building multiple routes. Additionally as the speed of mobile hosts increases, packet delivery ratio of both protocols decreases and control overhead of both of them increases.

On the other hand since SMP-DSR is a multipath routing protocol, the number of route reconstruction process in DSR protocol is more than that of SMP-DSR. Additionally DSR transmits considerably more RERR packets than SMP-DSR scheme, because DSR has more route disconnections and route recoveries.

In the second scenario the effect of pause time variation in different speeds is considered. In this scenario the number of nodes is equal to 40 and the simulation area dimensions are constant (1000m×1000 m).

Figures 4, 5 and 6 show packet delivery ratio and control overhead of each protocol. According to these figures data delivery ratio in SMP-DSR is considerably better than that of DSR in all cases. The control overhead of SMP-DSR is more than that of DSR.

These figures illustrate that as the value of pause time increases, the amount of packet delivery ratio, in both protocols increases and control overhead decreases.

IV. CONCLUSION AND FUTURE WORK

In this paper Split Multipath – Dynamic Source Routing (SMP-DSR) protocol was presented in order to improve fault tolerance in ad hoc networks. This scheme uses three routes for each data session; the shortest delay route and two maximally disjoint routes to the shortest delay route to efficiently utilize the available network resources.

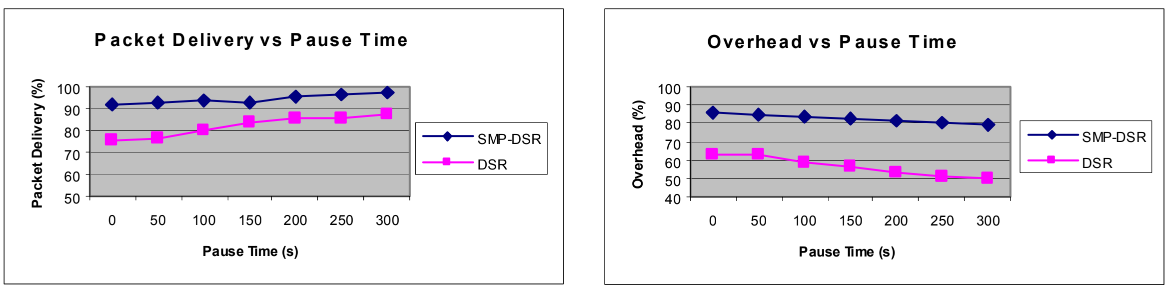

Figure 4: Packet Delivery Ratio and Control Overhead (min and max speed are set to zero and 10m/s, respectively)

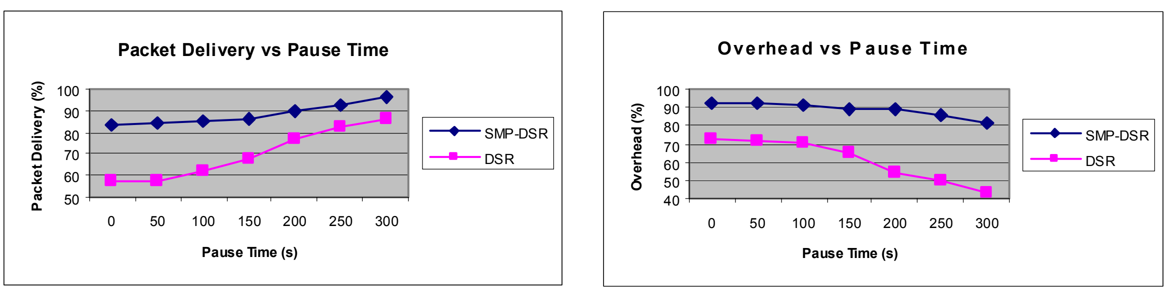

Figure 5: Packet Delivery Ratio and Control Overhead (min and max speed are set to zero and 30m/s, respectively)

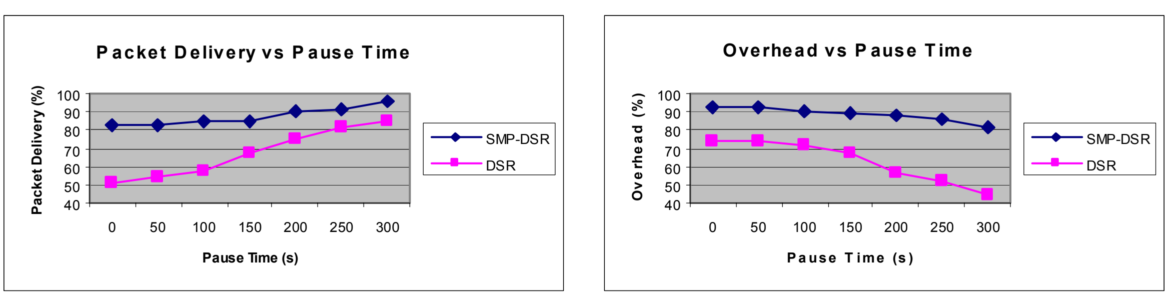

Figure 6: Packet Delivery Ratio and Control Overhead (min and max speed are set to zero and 50m/s, respectively)

Furthermore two caches in each node were used. All requested and received routes are cached in the first cache. The second cache contains active routes for data transfer.

In addition to RREQ, RREP and RERR, data delivery control packet was used to improve data delivery process.

Two scenarios were simulated to evaluate the performance of this protocol. In one of them the effect of network density in different speeds was considered, while in the other one the effect of pause time variation in different speeds was focused.

Simulation results show that SMP-DSR considerably improves fault tolerance and load balancing. It also achieves a higher rate of successful packet delivery than DSR in all cases. So this protocol will be applicable for reliable transmission of sensitive data.

We are going to include SMR protocol in our comparisons and decrease the control message overhead of our protocol in future work.

V. REFERENCES

[1] J. Broch, D.A. Maltz, D.B. Johnson, Y.-C. Hu, and J.Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," Proceedings of ACM/IEEE MOBICOM '98, Dallas, TX, Oct.1998, pp.85-97.

[2] D. Johnson, et. al., "The Dynamic Source Routing Protocol for Mobile Ad Hoc Network Internet Draft," www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt, July 2004.

[3] S. Ju-Lee, M. Gerla, "Split Multipath Routing with Maximally Disjoint Paths in Ad Hoc Networks," IEEE International Conference on Communications, Vol. 10, PP. 3201-3205, Helsinki, Jun. 2001.

[4] IEEE Computer Society LAN MAN Standards Committee, "Wireless LAN Medium Access Protocol (MAC) and Physical Layer (PHY) Specification," IEEE Std 802.11-1997. The Institute of Electrical and Electronics Engineers, New York, NY, 1997.

[5] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, "Scenario-based Performance Analysis of Routing Protocols for Mobile Ad hoc Networks," Proceedings of ACM/IEEE MOBICOM'99, Seattle, WA, Aug. 1999, pp. 195-206.

[6] S.-J. Lee and M. Gerla, "AODV-BR: Backup Routing in Ad hoc Networks," Proceedings of IEEE WCNC 2000, Chicago, IL, Sep. 2000.

[7] A. Nasipuri and S.R. Das, "On-Demand Multipath Routing for Mobile Ad Hoc Networks," Proceedings of IEEE ICCCN'99, Boston, MA, Oct. 1999, pp. 64-70.

[8] V.D. Park and M.S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," Proceedings of IEEE INFOCOM' 97, Kobe, Japan, Apr. 1997, pp. 1405-1413.

[9] UCLA Parallel Computing Laboratory and Wireless Adaptive Mobility Laboratory, "GloMoSim: A Scalable Simulation Environment for Wireless and Wired Network Systems," http://pcl.cs.ucla.edu/projects/domains/glomosim.html.