Multipath Routing Protocols in Mobile Ad Hoc Networks: A Comparative Survey

ilker Basaran, Mart Molle
Department of Computer Science and Engineering
University of California, Riverside
ibasaran,mart@cs.ucr.edu

Abstract—In order to achieve better throughput, load balancing and congestion avoidance multipath routing has been widely studied and used in wired networks. The good results inspired the researchers in mobile Ad Hoc area and many multipath routing protocols have been proposed. Each of these protocols has a different approach to problem and a different objective to achieve. To best of our knowledge, there is not a comprehensive comparative study among these protocols. In our study, we have examined 11 multipath routing protocols and compared them with respect to our framework. There is no winner of the comparison, but there are important inferences for researchers who will design new routing protocols.

Keywords—**MANET, Multipath Routing**

1. Introduction

There have been many multipath routing protocols proposed for mobile ad hoc networks throughout years. At first sight, one can classify those protocols according to their basic functioning, such as distance vector based or source routing based. But, in fact, there are many more subtle differences among them that affect their performance.

These differences mostly arise because of the objective the protocol tries to achieve. Congestion avoidance, better utilization of nodes in the network, load-balancing, lower frequency of route inquiries, lower delay and reliability of paths and robustness to node failures can be mentioned in this objective list.

In a recent paper, [GK04], in which there is a comparison of single path routing versus multipath routing with respect to load balancing in MANETs, it is stated that simply using multiple shortest paths instead of a single path does not improve the load balance. Also a daring claim is as follows: "In any ad hoc network with huge number of nodes, when the first K shortest paths are used for routing, multipath routing can balance the load better than single path routing only if a very large number of paths between any source-destination pair of nodes." And the conclusion of the idea is that the discovery and maintenance of such a large number of paths is very costly and infeasible. But when we look closely to the assumptions they made, we see that they are using K shortest paths between randomly chosen source-destination pairs in a circular region. It is almost obvious that the shortest paths will go through the center of the circular region causing congestion, reducing throughput and unbalancing the load. After the assumption they made, the results they have are expected.

Instead of using shortest K paths, we will see that many multipath routing algorithms are using low delayed, less congested paths which do not have to be shortest between source and destination. The usage of such multiple paths will employ nodes from different regions of the communication area, and hence enable load balancing and better utilization of the network. Thus, we realize the importance of careful design of a distributed multipath routing protocol rather than the un-necessity of it with the guidance of this paper.

In our study, first we are going to define three main groups of routing protocols according to their basic functioning and then categorize the multipath routing protocols. The summaries of each multipath protocol will provide us better understand their differences among each other and how they achieve their objectives. In order to compare the routing protocols, we are going to have a framework that will help us draw a conclusion at the end.

The rest of the paper is organized as follows: Next section will classify the multipath routing algorithms into three main groups and then briefly summarize their characteristics. In Section 3, the framework that we are going to use to compare the multipath routing protocols will be described. The comparisons among the protocols can also be found in Section 3. Section 4 will conclude our work.

2. Multipath Routing Algorithms

The broad classification of multipath routing protocols can be made according to their underlying routing scheme; distance vector based, source initiated or hybrid case, which mixes the two. To avoid being cliché, we wanted to categorize the protocols with respect to some other criteria rather than underlying routing scheme. The idea was grouping the protocols according to their usage of multiple paths; that is, if they are using multiple paths simultaneously or not. The outcome was interesting; because almost all the distance vector based protocols (except [CHAMP]) were using extra paths as alternate (backup) paths (i.e. usage of a backup path is bound to the failure of the primary path), while all the source routing based ones tried to use multiple paths simultaneously.

We must admit that, in the beginning of our study, we did not expect this sharp distinction between distance-vector based and source initiated routing protocols. As a result, we will continue with the classic classification and categorize the multipath routing protocols into three main groups; distance vector based, source initiated and hybrid case, which mixes the two.

The following are the brief summaries of the protocols that will provide us better understand their differences among each other and how they achieve their objectives. For each cluster of protocols, we will give the descriptions of related underlying routing protocol that are [AODV], [DSR] and [ZRP].

2.1. Distance Vector Based Protocols

The **Ad Hoc On-Demand Distance Vector** (AODV) routing protocol described in [AODV] minimizes the number of required broadcasts by creating routes on a demand basis. The authors of AODV classify it as a pure on-demand route acquisition system, since nodes that are not on a selected path do not maintain routing information or participate in routing table exchanges [AODV].

The route discovery process is initiated when the source does not already have a valid route to destination. Broadcast messages are relayed until either the destination or an intermediate node with a "fresh enough" route to the destination is located. To ensure loop freedom and path freshness, AODV utilizes sequence numbers. Intermediate nodes can reply to the route request only if they have a route to the destination whose corresponding destination sequence number is greater than or equal to that contained in the request. Duplicate route requests are simply discarded. The received request is cached for a certain amount of time in order to establish a reverse path if needed.

At the occurrence of a link failure, the immediate upstream node informs its upstream neighbors, up to the source node. Source node, then, may choose to re-initiate route discovery if that route is still needed. As a final note, we must mention the usage of periodic HELLO messages that are used to maintain the local connectivity of a node. This, of course, incurs considerable control overhead to the network.

The following seven multipath routing protocols are distance vector based routing schemes. AODV-BR, SMLDR, AODVM and AOMDV are modifications on top of AODV whereas TORA, ROAM and CHAMP are not connected with it. But since they keep the characteristics of distance vector based routing protocols, we put them in this section.

The **[AODV-BR]** (Backup Routing in Ad Hoc Networks) protocol is based on [AODV] and maintains multiple paths. After the broadcast of route request, the multiple paths are established during route reply phase. Also a mesh is structured from the overheard packets and the neighboring nodes are recorded as the next hops to destination in corresponding node's alternate route table. Alternate paths are use only when the primary link fails and to prevent packets tracing a loop, the mesh nodes forward a data packet only if the packet is not from their next hop to destination. Since one path is used at a given time, [AODV-BR] is not a genuine multipath idea. There is no simultaneous usage of multiple paths.

[SMLDR] (Shortest Multipath Routing Using Labeled Distances) is a distance vector based routing protocol. It enables loop freedom by maintaining the ordering of distance invariants. During the route discovery process the intermediate nodes record distinct previous hops to be able to identify multiple paths back to source along which the route replies can be relayed. The rest of discovery phase is similar to other distance vector based protocols. Route maintenance has not been discussed in [SMLDR] paper. It is just stated that "hellos or keep-alive packets are required to maintain the data path freshness of alternate paths." One important aspect we have to mention is that [SMLDR] uses alternate path routing and does not maintain disjoint paths.

[AODVM] (Ad Hoc On-Demand Distance Vector Routing – Multipath) has modifications on top of AODV to enable multiple node disjoint paths. The modifications are mostly to route request and reply processes while route recovery and maintenance are similar to AODV's. Only the destination node replies to a request to ensure node disjoint ness. Rather than edge disjoint ness, node disjoint ness is selected on purpose due to the fact that multiple paths passing through one intersection node might fail simultaneously upon that node's failure. This is of course a valid concern and important issue to address from the reliability point of view. Furthermore, the authors note that as the distance between source and destination is increased, the number of paths connecting them is very limited even at moderate node densities. So trying to exploit the use of these paths will be beneficial for routing concerns and issues. The main purpose AODVM is to "primarily design a multipath routing framework for providing enhanced robustness to node failures." In order to provide the reliability of paths, AODVM introduces *reliable path segments*, which is formed by reliable nodes. They achieve their goal and verify it with simulations.

[AOMDV] (Ad Hoc on-demand Multipath Distance Vector Routing) is basically multipath extensions on top of AODV. The route discovery process has been modified to enable multiple paths. They stress on link disjoint ness of multiple paths such that the paths may share nodes but no edges. Also the loop freedom property of paths is guaranteed by using sequence numbers of nodes. After mentioning link disjoint ness with a high importance, it is interesting that the authors prefer to use one path at a time rather than simultaneous usage of multiple paths. Their reason to choose single path at a time is the requirement of addressing issues, splitting traffic along each path and packet reordering at the destination. And as a different aspect of AOMDV than AODV, the usage of periodic HELLO messages to detect stale paths can be mentioned.

[TORA] (A Highly Adaptive Distributed Routing Algorithm) is a distributed routing protocol based on a "link reversal" algorithm. It is designed to discover routes on demand, provide multiple routes to a destination by maintaining a directed acyclic graph (DAG), establish routes quickly, and minimize communication overhead by localizing algorithmic reaction to topological changes when possible. Shortest-path routing is considered of secondary importance, and longer routes are often used to avoid the overhead of discovering newer routes. Each node has an assigned height with respect to destination by the protocol. Hence the data packets flow through the destination as water flows downhill towards a sink. The usage of the paths is not simultaneous, but as backup routes. The path freshness is detected via periodic messages and when a link breaks, the upstream node adjusts its height so that it is a local maximum with respect to its neighbors and transmits an update packet. The routes are locked down until it receives replies from all his neighbors. The control overhead incurred with this maintenance mechanism is not feasible beyond very low mobility.

The **[ROAM]** routing protocol uses inter-nodal coordination along acyclic sub-graphs, which is derived from the nodes' distances to destination. This operation is referred to as a "diffusing computation" which is originated by source and propagated by each node that has no entry for the destination. The advantage of this protocol is that it eliminates the search-to-infinity problem by stopping multiple flood searches when the required destination is no longer reachable. Among the multiple paths discovered, the shortest one is used primarily and the authors stated that "multiple paths can be used for forwarding packets, but not used in path calculations to prevent loops". At the occurrence of a link breakage, local maintenance is initiated by upstream node. Path freshness is maintained by data flow. As in TORA, the reliable delivery of control packets is required. The authors claim the protocol to work for both wired and wireless networks but, this requirement makes it unviable in wireless networks unless the mobility is very low.

[CHAMP] (Cooperative Packet Caching and Shortest Multipath Routing) can be considered as the most interesting routing protocol among distance vector based ones due to its usage of multiple paths. Its route discovery is similar to the "diffusing computation" as in ROAM. Interestingly, CHAMP uses *shortest equal cost multiple paths* based on paths of equal length. But, this limitation will substantially reduce the number of multiple paths that can be selected. If we are able to choose multiple paths of equal length, then we can use them simultaneously and forward the data in a simple round-robin fashion. The data flow will keep the links fresh. The route maintenance routine will be initiated locally, only if a node loses all its active routes to the destination. The caching of the packets will help salvaging them when an available path is found to the destination. Cooperative packet caching and rerouting of data packets are used to improve throughput.

2.2. Source Routing Based Protocols

The **Dynamic Source Routing** (DSR) protocol presented in [DSR] is an on-demand routing protocol that is based on the concept of source routing. Mobile nodes are required to maintain route caches that contain the source routes of which the source is aware. Entries in the route cache are continually updated as new routes are learned. When a mobile node has a packet to send to some destination, it first consults its route cache to determine whether it already has a route to the destination. If it has an unexpired route to the destination, it will use this route to send the packet. On the other hand, if the node does not have such a route, it initiates route discovery by broadcasting a route request packet. A route reply is generated when the route request reaches either the destination itself or an intermediate node which contains in its route cache an unexpired route to the destination [DSR]. The reply is sent by a cached route to source or if such route does not exist by the reverse route in the route record assuming the links are bi-directional. If the links are unidirectional the node may initiate its own route discovery and piggyback the route reply on the new route request.

The action taken against link failures is as follows: A route error packet is generated by the upstream node where the broken link is. When the route error packet is received by the neighboring nodes, they remove the broken link and all routes containing that link from their route cache. Data path freshness is verified by passive acknowledgements where a node is able to detect the next hop relaying the packet along the path.

[SMR] (Split Multipath Routing) establishes and utilizes multiple paths of maximally disjoint routes. The main reason behind this strict maximally disjoint ness idea is to prevent certain nodes from being congested. In route discovery phase, the intermediate nodes do not reply even if they have a path to destination. This is due to the algorithm of [SMR] which is based on selection of paths by destination node only. The duplicate route requests are not dropped if they are received from a different incoming link and whose hop count is less than or equal to the link from which the first received request. Then the destination selects *two* maximally disjoint paths from which many received. The route maintenance is initiated in one of two cases: start the route discovery process when any route of the session is broken or start route discovery process only when both routes of the session are broken. In [SMR], after stating that two paths will be chosen by the destination, the authors say that the number of paths may be modified to a higher number. But it is questionable that the algorithm will be scalable in that case. Because the maximally disjoint path selection by the destination will require a long time and higher processing power to select more than two paths.

Multipath Source Routing, **[MSR]**, is based on [DSR] and uses its route discovery routines. In the paper, the discovery algorithm of multiple paths and the preference of node disjoint or edge disjoint paths are not mentioned. MSR distributes the data load among discovered multiple paths by measuring RTT. According to delay of a route, weighted round robin model is used for granularity of packet sending. As an optimization, it is said that the intermediate nodes can re-schedule the packets on the fly. By making sure of loop freedom, an intermediate node can change the path a packet will follow after itself. This will allow cascaded multipath routing, which makes full use of network.

Another multipath routing algorithm based on DSR is **[MultRout]** (Multipath Routing for MANETs). The route maintenance and recovery processes are same as DSR's while path discovery has its own novelties. The algorithm tries to find node disjoint and loosely correlated among different paths. Here correlation is defined to be the number of links connecting the two paths. Also broadcast scheme is altered to enable multiple paths. In this selective broadcast method, a route request query is relayed only if it is the first to receive it or the path included in current message is node disjoint with the paths included in previously cached same route query messages. When the destination receives a query message, a reply is sent back to source only if the length of received route and the primary route is less than a given threshold. During routing multiple paths are used simultaneously with a probability inversely proportional to the length of the path.

2.3. Hybrid Protocols

In **Zone Routing Protocol** [ZRP], the nodes have a routing zone, which defines a range (in hops) that each node is required to maintain network connectivity proactively. This means that for nodes within the routing zone, routes are kept in a table and therefore are immediately available. For the outside nodes, routes are discovered on-demand (reactively) and any on-demand routing protocol can be used to find a route to the destination. It can be seen that, the control overhead is much less than pure proactive routing protocols. The delay for discovering routes is also improved with respect to pure reactive protocols such as DSR by allowing routes to be discovered faster. This is due to the fact that the boundary node of a routing zone will have the information of required destination proactively. So, the route request will only have to travel to a boundary node of the destination's zone. The route maintenance procedure for out-of-zone destinations will be same as the used reactive protocol's maintenance, where, inside the zone periodic updates will handle this issue proactively.

The idea of alternate path routing stems from traditional circuit-switched telephone networks. In **[APR]** (Alternate Path Routing), it says that when the primary route is better than the alternate routes, the primary route should be used until congestion occurs, at which time excess traffic can be diverted to secondary routes. For its underlying single path routing it uses Zone Routing Protocol, [ZRP], which is actually a hybrid routing protocol that uses a mix of reactive and proactive routing. It is also mentioned that ZRP is well-suited for source based alternate path routing. In the paper, path discovery process is touched on briefly while there is no description of path maintenance routine. The simulation results of APR indicate that its benefit is highly dependent on both the network topology and the channel access methods.

3. Comparisons

The following six concepts are description of fundamental issues to multipath routing algorithms. As studied earlier in [SMLDR] the consideration of these issues is necessary in any multipath design. The table that holds all the comparative data can be found in the Appendix. For the figures within this section, the legend is given below. Due to lack of space, we are able to provide it only once.

AODV-BR	⓿	MultRout	❻
AODVM	❶	ROAM	❼
AOMDV	❷	SMLDR	❽
APR	❸	SMR	❾
CHAMP	❹	TORA	❿
MSR	❺		

Table 1 — Legend for Figures

Multiple Path Discovery is the process by which multiple paths are discovered. This is similar to the route discovery mechanism used in single path routing protocols with the route replies relayed back to the source along the reverse routes established by the requests. However the intermediate nodes as well as the destination node now treat each request received from distinct previous hops as potential multipaths.

Most of the routing algorithms are using the route discovery process of underlying routing protocol with modifications on top of it. TORA introduces directed acyclic graphs into play and designs its own path discovery while two other distance vector based protocols, namely ROAM and CHAMP, uses diffusing search algorithm. The situation is similar in source initiated and hybrid protocols (SMR, MSR, MultRout, and APR). They either use the same or slightly modified version of path discovery process of the underlying routing protocol.

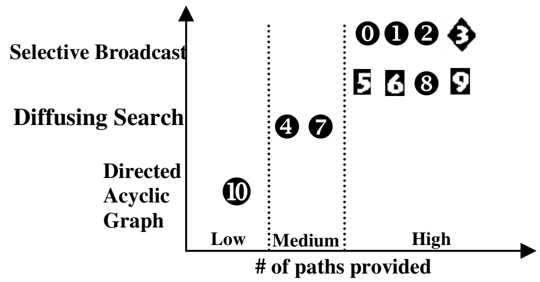
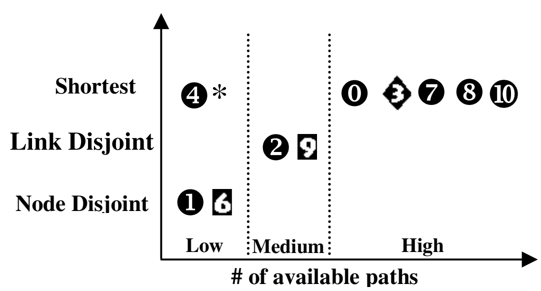


Figure 1

As seen in Figure 1, the number of paths provided varies with different heuristics. The least quantity is TORA's, due to its directed graph assumption. Also the requirement of reliable delivery of control messages makes it even less attractive. The diffusing search algorithm uses less number of broadcast messages in order to find a valid path to the destination than that of selective broadcast. But selective broadcast will provide more paths since diffusing search stops when an available path to destination is found. So there is a trade-off between the traffic overhead versus number of paths provided. If the objective of a routing protocol is load balancing and network utilization, our idea is to use selective broadcast.

Path Selection is the option of choosing certain paths with higher *benefit* against choosing all the paths that become available. Some filtering options are i) shortest multipaths, ii) disjoint paths, iii) feasible loop free paths, and iv) threshold on number of paths. In dense networks often a combination of one or more options provides the most effective route pruning.



*Shortest paths with equal lengths

Figure 2

Here, important considerations include whether to choose link disjoint, node disjoint or shortest paths. As displayed in Figure 2, this choice will return us as number of available paths to choose from. We showed that the shortest path selection gives more than the others, because even when we discard the currently used paths, there is always another shortest path in the network. But at this point, one should consider that; all source-destination pairs will try to employ shortest paths in between, and this will create congestion undoubtedly. We should also mention that trying to select shortest multiple paths with *equal* costs, as in CHAMP, will definitely limit the number of paths that one can choose from.

Another point is the debate of choosing between link disjoint paths and edge disjoint paths among different protocols. Our opinion on this is as follows: Carefully considering the behavior of MAC layer (802.11) we know that the inherent RTS/CTS mechanism of, blocks the transmission of all nearby nodes where a communication takes place. So even if we try to have node disjoint or link disjoint paths to prevent bottlenecks, because of the inherent characteristic of wireless communications, there is still a high probability that the packets of the same flow will compete for the same bandwidth. The following example will help us clarify this further.

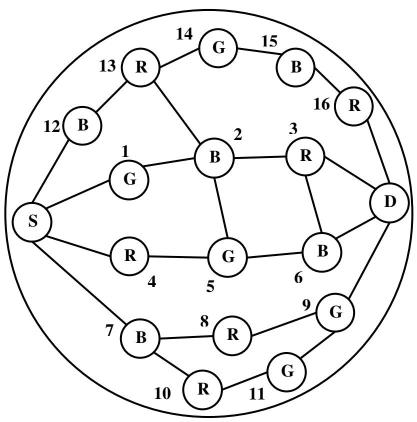


Figure 3 — Example network for path selection comparisons

In Figure 3, hearing graph of a network is shown. This is a similar network used in [GK04]. For simplicity, we are going to assume the absence of MAC layer. All possible paths (less than 10 hops) between source (S) and destination (D) are as follows:

p_1: S, 1, 2, 3, D p_2: S, 4, 5, 6, D
p_3: S, 7, 8, 9, D p_4: S, 1, 2, 5, 6, D
p_5: S, 1, 2, 3, 6, D p_6: S, 4, 5, 2, 3, D
p_7: S, 4, 5, 6, 3, D p_8: S, 12, 13, 2, 3, D
p_9: S, 7, 10, 11, 9, D p_{10}: S, 12, 13, 14, 15, 16, D
p_{11}: S, 12, 13, 2, 5, 6, D
p_{12}: S, 12, 13, 2, 3, 6, D
p_{13}: S, 1, 2, 13, 14, 15, 16, D
p_{14}: S, 4, 5, 2, 13, 14, 15, 16, D

The letters inside the nodes represent the colors, namely **R**ed, **G**reen and **B**lue. The colors are assigned such that, in the absence of MAC layer, the data flow through a particular path will be continuous, i.e. when the source sends a packet through a path, intermediate nodes of that path will be able to relay the packets without buffering. When a node has two same colored neighbors, then contention will occur, because all same colored nodes will attempt to send data in the same time slot.

We are going to look through all protocols one by one finding out which paths they will choose and if they will be able to use those paths simultaneously. First of all, the protocols, SMLDR, AODV-BR, AOMDV, TORA and APR use the extra paths for alternate (backup) routing, so there is no simultaneous usage of available paths. Hence, we will ignore those protocols in our examination.

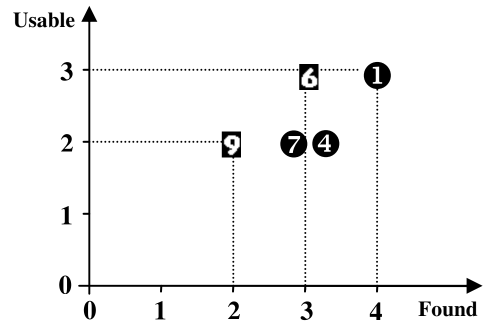


Figure 4 — Number of paths found vs. number of usable paths

Number of paths found by a protocol vs. number of usable paths is depicted in Figure 4. Following arguments explain how we obtained the numbers corresponding to each protocol.

AODV-M uses reliable path segments and requires strict node disjointness. Assuming all nodes are reliable and the protocol will choose from all available routes, p_1, p_2, p_3 and p_{10} will be selected by AODV-M. But, due to coloring, simultaneous usage of p_1 and p_2 *and* p_1 and p_{10} will cause contention at node 2 and at destination, respectively. Hence, at best, p_2, p_3 and p_{10} can be used simultaneously.

ROAM uses primarily shortest of multiple paths. So p_1, p_2 and p_3 will be selected for simultaneous usage. Due to previous coloring and contention arguments, only p_2 and p_3 can be used.

For this particular example, CHAMP behaves similar with ROAM. Its path selection method is shortest multiple routes with equal lengths. In our example p_1, p_2 and p_3 are shortest and their lengths are same. Hence the protocol will try to use those paths for simultaneous date transfer, but similarly, only p_2 and p_3 can be used.

SMR chooses only two paths to use at the same time. Those paths are the first arrived request and one more which is maximally disjoint with the first one. Assuming p_1 is the first arrived request, it will choose p_2 or p_3, which are maximally disjoint with p_1. If p_2 is selected, then it will not be able to be used due to contention at node 2. Only if p3 is selected, then the two selected paths are available for simultaneous usage.

Finally, MultRout will choose all available node disjoint paths of which their correlation is less than a constant *k*. Hence either p_1 or p_2 will be ignored due to correlation constant whereas p_3 and p_{10} are available. Consequently, p_2, p_3 and p_{10} are going to be used.

It is seen that the protocols prune the available paths in order to avoid contention but they can not achieve it even in this naïve example network which does not have a MAC layer. This is due to the fact that the communication between two nodes can be heard by neighboring nodes in a wireless environment and contention is almost unavoidable. This shows that the physical characteristics of wireless ad hoc networks prevent the protocols to achieve their goals in pruning multiple available paths.

Moreover, it has been stated in [AODV-M] that as the distance between a source and its destination is increased, one could find no more than a very limited number of paths between them. So before recklessly discarding these paths one should try to exploit their use.

Thus, the trade-off here is deciding how harsh to behave in pruning the available paths. Being too relaxed may not work out as intended while being too strict may bound the protocol with a very limited number of paths.

Path Usage Policy describes whether the available paths would be used simultaneously or one at a time. The former requires data to be forwarded along all the paths. This lends itself naturally to load balancing and traffic engineering approaches. The latter forwards data only along the primary (potentially shortest) path and when the primary path fails alternate paths are employed.

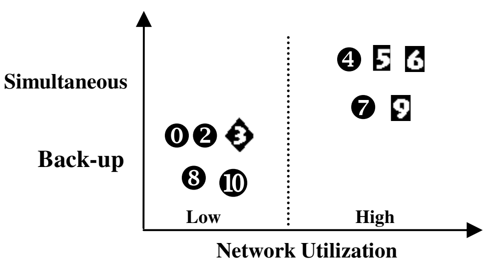


Figure 5

Figure 5 shows a clear cut among protocols in the sense of path usage but in some papers (AODV-M); it is not evident if the protocol is using the multiple paths simultaneously or as backup routes. SMLDR, AODV-BR, AOMDV, TORA, and APR are the protocols that use the multiple paths as alternate (backup) routes. The ones that have simultaneous usage of more than one path are ROAM, CHAMP, SMR, MSR, and MultRout. The important point here is that there is no guarantee that the backup route will still be available when a congestion or link breakage occurs. So the protocols that use the extra paths as backup may end up with re-requesting paths to destinations. If we want to utilize the networks resources as much as we can and distribute the load balance, we should try to exploit available multiple paths by their simultaneous usage. As an extra benefit of this, early detection of path congestion or failure will be possible.

Path Maintenance refers to how the multiple paths would be maintained. For example, if number of available paths goes below a threshold, initiating a new route discovery ensures continuous availability of multiple paths to the destination.

The maintenance of the routes is initiated on different circumstances. If the protocol is using one path a time and the alternate paths are also lost to destination, then the maintenance procedure is employed, where; for the protocols that use multiple paths simultaneously, maintenance is initiated either upon the breakage of a single path or failure of all paths. The maintenance can be done in two ways; locally or by informing the source. It is safer to inform the source and letting it handle the situation rather than local recovery of path failure. The reason why it is safer to do so can be explained by an example.

As depicted in Figure 6-a, the path is passing through the nodes A, B and C to reach from S to D. Now suppose node D started to move through the dashed arrow, so that the link between C and D breaks and D enters the power range of E. If the intermediate node C tries to *locally* recover the broken link, it will broadcast a Route Request and end up with a path as shown in Figure 6-b. As we can see, since C tried to recover the broken link, the total length of the path increased, whereas if the source had initiated a Route Request, the algorithm would find the path S-A-E-D which is a much better choice. If this worst case scenario repeats a couple of times, then we will have an unnecessarily long path that will dramatically increase the transmission time as well as introducing a higher probability of congestion, packet loss and link breakage.

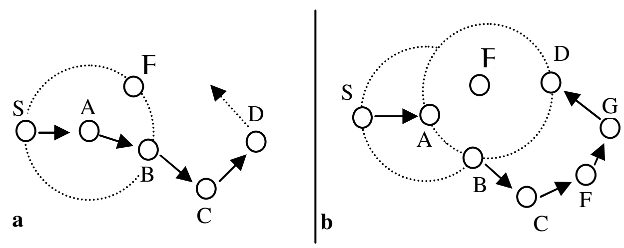


Figure 6

Hence, we can conclude that either we should let the source handle the problem or try to recover locally in a very careful manner trying to avoid such situations. The protocols which are using local maintenance are SMLDR, AODV-BR, TORA, ROAM, and CHAMP. Other than TORA, the rest may encounter with the above defined problem. TORA is immune against this problem due to its path recovery with DAG calculation, but the control overhead incurred with this maintenance mechanism is not feasible beyond very low mobility. The remaining multipath routing protocols (AODV-M, SMR, MSR, MultRout and APR), inform the source and it is the source's responsibility to start the maintenance procedure.

Data Path Freshness Strategy ensures that the data paths are still valid. The freshness maintenance strategy is closely tied to the path usage policy. When all the paths are simultaneously used the data packets flowing along these paths automatically update the lifetime. When the paths are used one at a time the primary path's lifetime always gets updated but to keep the alternate paths alive, HELLO packets are needed.

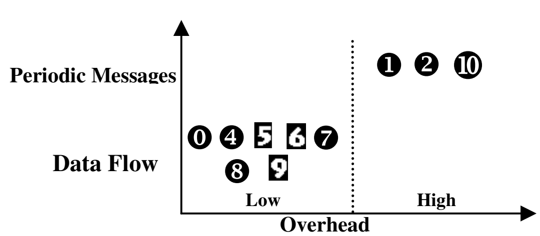


Figure 7

Almost all the multipath routing protocols use the data flow to update the time-tags of the paths and keep them fresh. Only AOMDV and AODV-M use periodic HELLO messages to detect the links are valid. The usage of periodic control messages introduce considerable overhead, but since the objective of AODV-M is reliability and robustness, then its usage of periodic messages can be condoned. The protocols SMR, MSR, MultRout and APR did not mention how they kept the paths fresh but we assume that they inherit it from their underlying routing protocols which are respectively DSR for the first three of them and ZRP for APR. Since DSR and ZRP use data flows to keep the paths fresh, they are similar to most of the other protocols in this respect.

Data Forwarding refers to the way in which the data is to be forwarded over the multiple paths. This property is meaningful only when the available paths are used simultaneously. Examples of a few schemes include the simple round robin and heuristic forwarding based on path lengths.

For the routing protocols that do not employ simultaneous usage of multiple paths, their data forwarding mechanism cannot be argued. Out of CHAMP, MSR, SMR and MultRout, *MSR* seems to be the best with its weighted round robin distribution of packets. The weights are assigned by measuring delay of each path. Simple round robin or proportionality with respect to hop count is not a good choice for a data forwarding mechanism because sending the packets without the knowledge of congestion level of a particular path is not wise. The delay will reveal this information to us.

4. Conclusions and Future Work

As opposed to their single path counterparts, on-demand routing protocols with multipath capability can effectively deal with mobility-induced link failures in mobile ad hoc networks. The outcome of this fact is the multipath routing protocols that have been proposed for mobile ad hoc networks throughout years.

To best of our knowledge, there is not a comparative study amongst these protocols. We have grouped and summarized the multipath routing protocols and then compared them with respect to our described comparison framework. The results were interesting, because many routing protocols used the available multiple paths as backup routing information and continued to send the data through a single path until a link failure occurs.

Also in a recent paper, [GK04], it is discussed that using multiple paths does not balance the load better than single path routing. Load balancing is one of the main objectives of multipath routing protocols and this statement does not reflect the effort of them. With careful consideration it is understood that the assumptions of the authors lead them to this conclusion. They chose to use *K shortest* paths between randomly chosen source-destination pairs in a circular region. Once this assumption is made, it is inevitable to come out with the discussed result.

The usage of multiple paths as backup routes and the discussion on the inability of load balancing by multipath routing protocols, forces us to think that multipath routing is unnecessary in mobile ad hoc networks. Our opinion is just on the contrary. There are multipath routing protocols that use the available paths simultaneously and, moreover, load balancing *can* be achieved by those protocols with reasonable considerations. Thus, the two concerns mentioned above are unfounded.

Carefully designed multipath routing protocols promise congestion avoidance, load balancing, improved throughput, fast recovery from link failures hence a more robust network, and a better utilization of the network resources.

Our future work will focus on the design of a multipath routing protocol that will consider the advantages and weaknesses of the protocols mentioned in this paper. We will pay special attention to simultaneous usage of paths, data forwarding mechanism considering the delay of the available paths and a neat local maintenance procedure.

5. References

[AODV] Elizabeth M. Royer and Charles E. Perkins. Ad-hoc On-Demand Distance Vector Routing. In *2nd IEEE Workshop on Mobile Computing Systems and Applications*, Feb 1999.

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

[AODVM] Ye, Z., Krishnamurthy S.V., and Tripathi, S.K., A Framework for Reliable Routing in Mobile Ad hoc Networks, *IEEE INFOCOM 2003*.

[AOMDV] M. K. Marina and S. R. Das. On-demand multipath distance vector routing in ad hoc networks. *Proceedings IEEE ICNP*, pages 14–23, 2001.

[APR] M.R. Pearlman, Z.J. Haas, P. Scholander, and S.S. Tabrizi, Alternate Path Routing in Mobile Ad Hoc Networks, *IEEE MILCOM'2000*, Los Angeles, CA, October 22-25, 2000

[CHAMP] A. Valera, W. Seah, and S. V. Rao. Cooperative packet caching and shortest multipath routing in mobile ad hoc networks. In *Proceedings IEEE INFOCOM*, pages 260–269, 2003.

[DSR] D. Johnson, D. Maltz, Y Hu and J Jetcheva. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. *IETF MANET Internet Draft*, Feb 2002.

[GK04] Yashar Ganjali and Abtin Keshavarzian. Load balancing in ad hoc networks: Single-path routing vs. multi-path routing. In *Proceedings of the IEEE INFOCOM'04*, Hong Kong, March 2004.

[LDR] J. J. Garcia-Luna-Aceves, M. Mosko, and C. E. Perkins. A new approach to on-demand loop-free routing in ad hoc networks. *Proceedings PODC*, pages 53–62, 2003.

[MSR] Wang, L., Shu, Y., Dong, M., Zhang, L., Yang, O.W.W.: Adaptive Multipath Source Routing in Ad Hoc Networks. *IEEE International Conference on Communications, Vol. 3 (2001)*

[MultRout] K. Wu and J. Harms, On-demand multipath routing for mobile ad hoc network, *EPMCC 01, Vienna, Austria*, Feb. 2001, pp.1-7.

[NCD01] A. Nasipuri, R. Castaneda, and S. R. Das. Performance of route caching strategies in dynamic source routing for on-demand protocols in mobile ad hoc networks. *ACM/Kluwer Mobile Networks and Applications (MONET)*, 6(4):339–349,2001.

[ROAM] J. Raju and J. J. Garcia-Luna-Aceves. A new approach to on-demand loop-free multipath routing. *Proceedings IC3N*, pages 522–527, 1999.

[SMLDR] C. Balasubramanian and J.J. Garcia-Luna-Aceves, Shortest Multipath Routing Using Labeled Distances, *The 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems, Fort Lauderdale, Florida*, October 25-27, 2004.

[SMR] S. Lee and M. Gerla. Split multipath routing with maximally disjoint paths in ad hoc networks. *Proceedings IEEE ICC*, pages 3201–3205, 2001.

[TORA] V. D. Park and M. S. Corson. A Highly Adaptive Distributed Routing Algorithm for Mobile Wirelss Networks. In *IEEE INFOCOM, Japan*, Apr 1997.

[ZRP] Zygmunt J. Haas and Marc R. Pearlman. The Performance of Query Control Schemes for the Zone Routing Protocol. *IEEE/ACM Transactions on Networking*, Aug 2001

APPENDIX

	Discovery	Selection	Usage	Maintenance	Freshness	Forwarding
SMLDR	Request is broadcasted and a timer started for discovery of multiple paths	Source Selects the shortest and the next shortest paths	Backup	Intermediate nodes inform the source on link breakage	Data Flow (Soft State)	N/A
AODV-BR	Same as AODV, but during replies, nodes overhear comm. and establish alternate paths	Same as AODV	Backup	Only if a link break occurs, locally	Data Flow (Soft State)	N/A
AOMDV	Modifications on top of AODV to discover multiple paths	Strictly link disjoint determined locally	Backup	Similar to AODV's	Periodic HELLO messages	N/A
AODV-M	Modifications on top of AODV to discover multiple paths	Strictly node disjoint chosen by destination	Reliable Path Segments, simultaneous (when available)	Similar to AODV's	Periodic HELLO messages	N/A
TORA	Provides multiple paths by building a directed acyclic graph rooted at destination	Directed graph – hence 1 available path at a time	Backup	upstream node adjusts its height to a local maximum and transmits an update packet	Periodic Messages	N/A
ROAM	Diffusing Search	Primarily, shortest of multiple paths	Multiple paths can be used	Local maintenance by upstream node	Data Flow (Soft State)	N/A
CHAMP	Diffusing Search	Shortest Multiple Routes with equal lengths	Simultaneous (when multiple paths available)	Locally, only when all active routes broken	Data Flow (Soft State)	Simple Round-Robin
SMR	Selective Broadcast on top of DSR to discover multiple paths	Destination replies to first request and one more which is maximally disjoint with the first one	Simultaneous (2 paths)	Two options; recover even a single break occurs or only when both fails	Not mentioned	Split into two
MSR	Same as DSR	Path selection not mentioned in the paper	Simultaneous	Similar to DSR's (source-based)	Not mentioned	Weighted round robin, calculated by measuring RTT
MultRout	Selective Broadcast on top of DSR to discover multiple paths	Node disjoint and correlation should be less than a constant between two paths	Simultaneous usage	Similar to DSR's (source-based)	Not mentioned	inversely proportional to length of path
APR	Same as in ZRP	*k* best set of routes are selected (in the paper k is set to 2)	Backup	Inform the source and let it handle	Not mentioned	N/A