

Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges^{*}

Stephen Mueller¹, Rose P. Tsang², and Dipak Ghosal¹

¹ Department of Computer Science, University of California, Davis, CA 95616

² Sandia National Laboratories, Livermore, CA 94551

Abstract. Mobile ad hoc networks (MANETs) 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. MANET nodes are typically distinguished by their limited power, processing, and memory resources as well as high degree of mobility. In such networks, the wireless mobile nodes may dynamically enter the network as well as leave the network. 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 is a crucial issue to the design of a MANET. In this paper, we specifically examine the issues of multipath routing in MANETs. 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. Load balancing is of especial importance in MANETs because of the limited bandwidth between the nodes. We also discuss the application of multipath routing to support application constraints such as reliability, load-balancing, energy-conservation, and Quality-of-Service (QoS).

1 Introduction

Mobile ad hoc networks (MANETs) 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. MANETs have potential use in a wide variety of disparate situations. Such situations include moving battlefield communications to disposable sensors which are dropped from high altitudes and dispersed on the ground for hazardous materials detection. Civilian applications include simple scenarios such as people at a conference in a hotel where their laptops comprise a temporary MANET to more complicated scenarios such as highly mobile vehicles on the highway which form an ad hoc network in order to provide vehicular traffic management.

MANET nodes are typically distinguished by their limited power, processing, and memory resources as well as high degree of mobility. In such networks,

^{*} This research was funded in part by a grant from Sandia National Laboratories, CA, USA.

the wireless mobile nodes may dynamically enter the network as well as leave the network. 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 is a crucial issue to the design of a MANET.

Routing protocols in conventional wired networks are usually based upon either distance vector or link state routing algorithms. Both of these algorithms require periodic routing advertisements to be broadcast by each router. In distance vector routing [1], each router broadcasts to all of its neighboring routers its view of the distance to all other nodes; the neighboring routers then compute the shortest path to each node. In link-state routing [2], each router broadcasts to its neighboring nodes its view of the status of each of its adjacent links; the neighboring routers then compute the shortest distance to each node based upon the complete topology of the network.

These conventional routing algorithms are clearly not efficient for the type of dynamic changes which may occur in an ad-hoc network. In conventional networks, routers do not generally move around and only rarely leave or join the network. In an environment with mobile nodes, the changing topology will not only trigger frequent re-computation of routes but the overall convergence to stable routes may be infeasible due to the high-level of mobility.

Clearly, routing in MANETs must take into consideration their important characteristics such as node mobility. Work on single path (or unipath) routing in MANETs has been proposed in [3] [4]. In this paper, we specifically examine the issues of multipath routing in MANETs. Multipath routing allows the establishment of multiple paths between a single source and single destination node. Multipath routing is typically proposed in order to increase the reliability of data transmission (i.e., fault tolerance) or to provide load balancing. Load balancing is of especial importance in MANETs because of the limited bandwidth between the nodes.

The rest of the paper is organized as follows. In Section 2, we provide background into the area of multipath routing for wired networks. In Section 3, we present an overview of the characteristics of MANETs. We discuss techniques for supporting multipath routing in MANETs in Section 4. In Section 5, we discuss the application of multipath routing to support application constraints such as reliability, energy-conservation, and Quality-of-Service (QoS). Finally in Section 6, we provide the conclusion.

2 Background on Multipath Routing

Multipath routing has been explored in several different contexts. Traditional circuit switched telephone networks used a type of multipath routing called alternate path routing. In alternate path routing, each source node and destination node have a set of paths (or multipaths) which consist of a primary path and one or more alternate paths. Alternate path routing was proposed in order to decrease the call blocking probability and increase overall network utilization.

In alternate path routing, the shortest path between exchanges is typically one hop across the backbone network; the network core consists of a fully connected set of switches. When the shortest path for a particular source destination pair becomes unavailable (due to either link failure or full capacity), rather than blocking a connection, an alternate path, which is typically two hops, is used. Well known alternate path routing schemes such as Dynamic Nonhierarchical Routing and Dynamic Alternative Routing are proposed and evaluated in [5] [6].

Multipath routing has also been addressed in data networks which are intended to support connection-oriented service with QoS. For instance, Asynchronous Transfer Mode (ATM) [7] networks use a signaling protocol, PNNI, to set up multiple paths between a source node and a destination node. The primary (or optimal) path is used until it either fails or becomes over-utilized, then alternate paths are tried. Using a crankback process, the alternate routes are attempted until a connection is completed.

Alternate or multipath routing has typically lent itself to be of more obvious use to connection-oriented networks; call blocking probability is only relevant to connection oriented networks. However, in packet-oriented networks, like the Internet, multipath routing could be used to alleviate congestion by routing packets from highly utilized links to links which are less highly utilized. The drawback of this approach is that the cost of storing extra routes at each router usually precludes the use of multipath routing. However, multipath routing techniques have been proposed for OSPF [2], a widely used Internet routing protocol.

3 Overview of Mobile Ad Hoc Networks

In MANETs³ communication between nodes is done through the wireless medium. Because nodes are mobile and may join or leave the network, MANETs have a dynamic topology. Nodes that are in transmission range of each other are called neighbors. Neighbors can send directly to each other. However, when a node needs to send data to another non-neighbor node, the data is routed through a sequence of multiple hops, with intermediate nodes acting as routers. An example ad hoc network is depicted in Figure 1.

There are numerous issues to consider when deploying MANETs. The following are some of the main issues.

1. **Unpredictability of environment:** Ad hoc networks may be deployed in unknown terrains, hazardous conditions, and even hostile environments where tampering or the actual destruction of a node may be imminent. Depending on the environment, node failures may occur frequently.
2. **Unreliability of wireless medium:** Communication through the wireless medium is unreliable and subject to errors. Also, due to varying environmental conditions such as high levels of electro-magnetic interference (EMI) or inclement weather, the quality of the wireless link may be unpredictable.

³ In this paper, we will use the terms MANETs and ad hoc networks interchangeably.

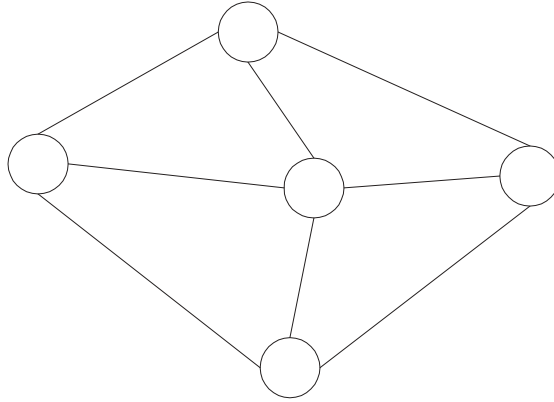


Fig. 1. An example ad hoc network, with circles representing nodes. Two nodes that are in transmission range of each other are connected by a line.

Furthermore, in some applications, nodes may be resource-constrained and thus would not be able to support transport protocols necessary to ensure reliable communication on a lossy link. Thus, link quality may fluctuate in a MANET.

3. **Resource-constrained nodes:** Nodes in a MANET are typically battery-powered as well as limited in storage and processing capabilities. Moreover, they may be situated in areas where it is not possible to re-charge and thus have limited lifetimes. Because of these limitations, they must have algorithms which are energy-efficient as well as operating with limited processing and memory resources. The available bandwidth of the wireless medium may also be limited because nodes may not be able to sacrifice the energy consumed by operating at full link speed.
4. **Dynamic topology:** The topology in an ad hoc network may change constantly due to the mobility of nodes. As nodes move in and out of range of each other, some links break while new links between nodes are created.

As a result of these issues, MANETs are prone to numerous types of faults including,

1. **Transmission errors:** The unreliability of the wireless medium and the unpredictability of the environment may lead to transmitted packets being garbled and thus received in error.
2. **Node failures:** Nodes may fail at any time due to different types of hazardous conditions in the environment. They may also drop out of the network either voluntarily or when their energy supply is depleted.
3. **Link failures:** Node failures as well as changing environmental conditions (e.g., increased levels of EMI) may cause links between nodes to break.

4. **Route breakages:** When the network topology changes due to node/link failures and/or node/link additions to the network, routes become out-of-date and thus incorrect. Depending upon the network transport protocol, packets forwarded through stale routes may either eventually be dropped or be delayed; packets may take a circuitous route before eventually arriving at the destination node.
5. **Congested nodes or links:** Due to the topology of the network and the nature of the routing protocol, certain nodes or links may become over-utilized, i.e., congested. This will lead to either larger delays or packet loss.

Routing protocols for MANETs must deal with these issues to be effective. In the remainder of this section, we present an overview of some of the key unipath routing protocols for MANETs.

3.1 Unipath Routing in MANETs

Routing protocols are used to find and maintain routes between source and destination nodes. Two main classes of ad hoc routing protocols are table-based and on-demand protocols [3] [4] [8] [9]. In table-based protocols [8] [9], each node maintains a routing table containing routes to all nodes in the network. Nodes must periodically exchange messages with routing information to keep routing tables up-to-date. Therefore, routes between nodes are computed and stored, even when they are not needed. Table-based protocols may be impractical, especially for large, highly mobile networks. Because of the dynamic nature of ad hoc networks, a considerable number of routing messages may have to be exchanged in order to keep routing information accurate or up-to-date.

In on-demand protocols [3] [4], nodes only compute routes when they are needed. Therefore, on-demand protocols are more scalable to dynamic, large networks. When a node needs a route to another node, it initiates a route discovery process to find a route. On-demand protocols consist of the following two main phases.

1. **Route discovery** is the process of finding a route between two nodes (see Figure 2)
2. **Route maintenance** is the process of repairing a broken route or finding a new route in the presence of a route failure (see Figure 3)

Most currently proposed routing protocols for ad hoc networks are unipath routing protocols. In unipath routing, only a single route is used between a source and destination node. Two of the most widely used protocols are the Dynamic Source Routing (DSR) [4] and the Ad hoc On-demand Distance Vector (AODV) [3] protocols. AODV and DSR are both on-demand protocols. Since most of the multipath routing protocols discussed in this paper are an extension of one of these two routing protocols, the following subsection gives a brief overview of DSR and AODV.

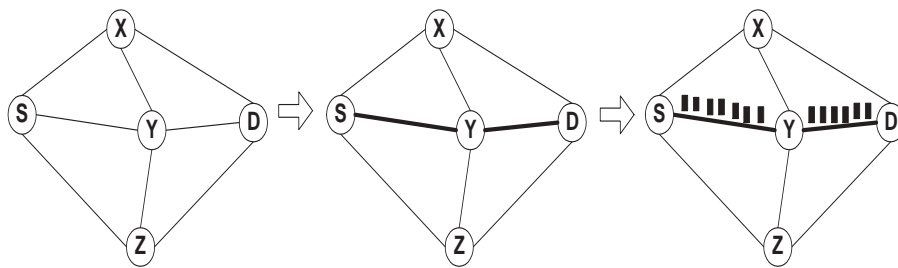


Fig. 2. An example of route discovery in an ad hoc network. In order for node S to send data to node D, it must first discover a route to node D. Node S discovers a route to node D going through node Y, and sets up the route. Once the route is established, node S can begin sending data to node D along the route.

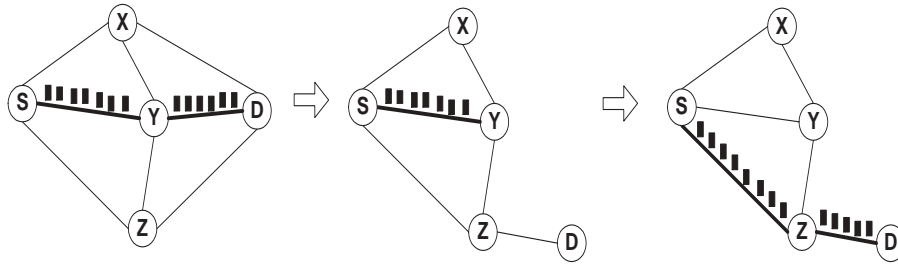


Fig. 3. An example of route maintenance in an ad hoc network. Node S sends data along an established route to node D through node Y. When node D moves out of range of node Y, this route breaks. Node S finds a new route to node D through node Z, and thus can begin sending data to node D again.

Dynamic Source Routing. DSR is an on-demand routing protocol for ad hoc networks. Like any source routing protocol, in DSR the source includes the full route in the packets' header. The intermediate nodes use this to forward packets towards the destination and maintain a route cache containing routes to other nodes.

Route discovery. If the source does not have a route to the destination in its route cache, it broadcasts a route request (RREQ) message specifying the destination node for which the route is requested. The RREQ message includes a route record which specifies the sequence of nodes traversed by the message. When an intermediate node receives a RREQ, it checks to see if it is already in the route record. If it is, it drops the message. This is done to prevent routing loops. If the intermediate node had received the RREQ before, then it also drops the message. The intermediate node forwards the RREQ to the next hop according to the route specified in the header. When the destination receives the RREQ, it sends back a route reply message. If the destination has a route to the source in its route cache, then it can send a route response (RREP) message along this route. Otherwise, the RREP message can be sent along the reverse route back to the source. Intermediate nodes may also use their route cache to reply to RREQs. If an intermediate node has a route to the destination in its cache, then it can append the route to the route record in the RREQ, and send an RREP back to the source containing this route. This can help limit flooding of the RREQ. However, if the cached route is out-of-date, it can result in the source receiving stale routes.

Route maintenance. When a node detects a broken link while trying to forward a packet to the next hop, it sends a route error (RERR) message back to the source containing the link in error. When an RERR message is received, all routes containing the link in error are deleted at that node.

Ad Hoc On-demand Distance Vector. AODV is an on-demand routing protocol for ad hoc networks. However, as opposed to DSR, which uses source routing, AODV uses hop-by-hop routing by maintaining routing table entries at intermediate nodes.

Route Discovery. The route discovery process is initiated when a source needs a route to a destination and it does not have a route in its routing table. To initiate route discovery, the source floods the network with a RREQ packet specifying the destination for which the route is requested. When a node receives an RREQ packet, it checks to see whether it is the destination or whether it has a route to the destination. If either case is true, the node generates an RREP packet, which is sent back to the source along the reverse path. Each node along the reverse path sets up a forward pointer to the node it received the RREP from. This sets up a forward path from the source to the destination. If the node is not the destination and does not have a route to the destination, it rebroadcasts the

RREQ packet. At intermediate nodes duplicate RREQ packets are discarded. When the source node receives the first RREP, it can begin sending data to the destination.

To determine the relative degree out-of-dateness of routes, each entry in the node routing table and all RREQ and RREP packets are tagged with a destination sequence number. A larger destination sequence number indicates a more current (or more recent) route. Upon receiving an RREQ or RREP packet, a node updates its routing information to set up the reverse or forward path, respectively, only if the route contained in the RREQ or RREP packet is more current than its own route.

Route Maintenance. When a node detects a broken link while attempting to forward a packet to the next hop, it generates a RERR packet that is sent to all sources using the broken link. The RERR packet erases all routes using the link along the way. If a source receives a RERR packet and a route to the destination is still required, it initiates a new route discovery process. Routes are also deleted from the routing table if they are unused for a certain amount of time.

Many multipath routing protocols have been proposed in literature. Most of these protocols are an extension of either DSR or AODV. In the following section, we first outline the key advantages of multipath routing and then discuss some of the proposed multipath extensions to both DSR and AODV.

4 Multipath Routing in MANETs

Standard routing protocols in ad hoc wireless networks, such as AODV and DSR, are mainly intended to discover a single route between a source and destination node. Multipath routing consists of finding multiple routes between a source and destination node. These multiple paths between source and destination node pairs can be used to compensate for the dynamic and unpredictable nature of ad hoc networks.

4.1 Benefits of Multipath Routing

As mentioned before, multiple paths can provide load balancing, fault-tolerance, and higher aggregate bandwidth. Load balancing can be achieved by spreading the traffic along multiple routes. This can alleviate congestion and bottlenecks.

From a fault tolerance perspective, multipath routing can provide route resilience. To demonstrate this, consider Figure 4, where node S has established three paths to node D. If node S sends the same packet along all three paths, as long as at least one of the paths does not fail, node D will receive the packet. While routing redundant packets is not the only way to utilize multiple paths, it demonstrates how multipath routing can provide fault tolerance in the presence of route failures.

Since bandwidth may be limited in a wireless network, routing along a single path may not provide enough bandwidth for a connection. However, if multiple

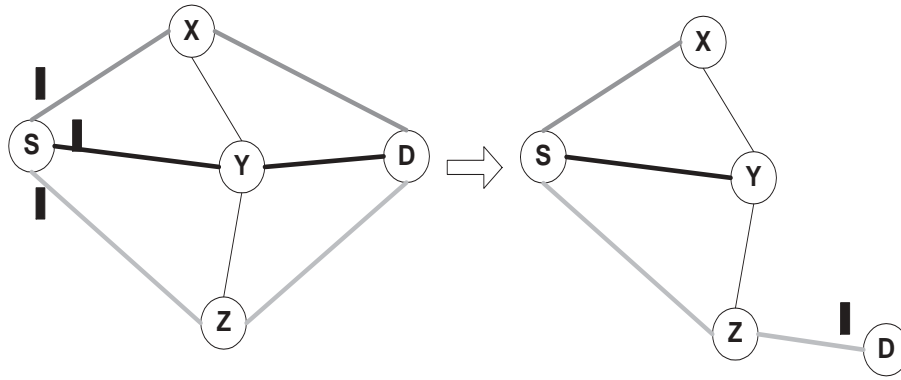


Fig. 4. Source node S routes the same packet to destination node D along the routes SXD, SYD, and SZD. When node D moves, routes SXD and SYD break, but route SZD is still able to deliver the packet to node D.

paths are used simultaneously to route data, the aggregate bandwidth of the paths may satisfy the bandwidth requirement of the application. Also, since there is more bandwidth available, a smaller end-to-end delay may be achieved. Due to issues at the link layer, using multiple paths in ad hoc networks to achieve higher bandwidth may not be as straightforward as in wired networks. Because nodes in the network communicate through the wireless medium, radio interference must be taken into account. Transmissions from a node along one path may interfere with transmissions from a node along another path, thereby limiting the achievable throughput. However, results show that using multipath routing in ad hoc networks of high density results in better throughput than using unipath routing [10]. We defer further discussion of the link layer issues in multipath routing to Section 4.3.

4.2 Multipath Routing Components

Multipath routing consists of three components: route discovery, route maintenance, and traffic allocation. We discuss these components in the following subsections.

Route Discovery and Maintenance. Route discovery and route maintenance consists of finding multiple routes between a source and destination node. Multipath routing protocols can attempt to find node disjoint, link disjoint, or non-disjoint routes. Node disjoint routes, also known as totally disjoint routes, have no nodes or links in common. Link disjoint routes have no links in common, but may have nodes in common. Non-disjoint routes can have nodes and links in common. Refer to Figure 5 for examples of the different kinds of multipath routes.

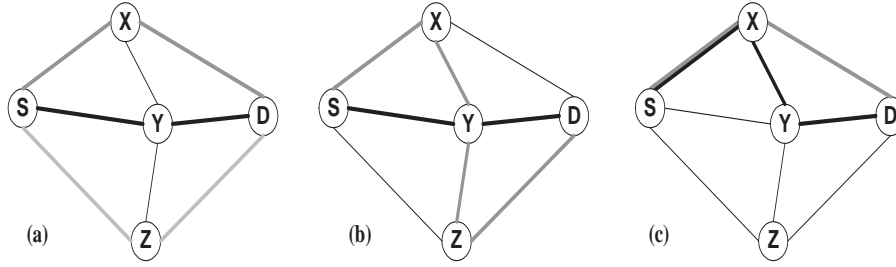


Fig. 5. Routes SXD, SYD, and SZD in (a) have no links or nodes in common and are therefore node disjoint. Routes SXYZD and SYD in (b) have node Y in common and are therefore only link disjoint. Routes SXD and SXYD in (c) have node X and link SX in common and are therefore non-disjoint.

Disjoint routes offer certain advantages over non-disjoint routes. For instance, non-disjoint routes may have lower aggregate resources than disjoint routes, because non-disjoint routes share links or nodes. In principle, node disjoint routes offer the most aggregate resources, because neither links nor nodes are shared between the paths. Disjoint routes also provide higher fault-tolerance. When using non-disjoint routes, a single link or node failure can cause multiple routes to fail. In node or link disjoint routes, a link failure will only cause a single route to fail. However, with link disjoint routes, a node failure can cause multiple routes that share that node to fail. Thus, node disjoint routes offer the highest degree of fault-tolerance.

The main advantage of non-disjoint routes is that they can be more easily discovered. Because there are no restrictions that require the routes to be node or link disjoint, more non-disjoint routes exist in a given network than node or link disjoint routes. Because node-disjointness is a stricter requirement than link-disjointness, node-disjoint routes are the least abundant and are the hardest to find. It has been shown that in moderately dense networks, there may only exist a small number of node disjoint routes between any two arbitrary nodes, especially as the distance between the nodes increases [11]. This is because there may be sparse areas between the two nodes that act as bottlenecks. Given the trade-offs between using node disjoint versus non-disjoint routes, link disjoint routes offer a good compromise between the two. In the following subsection, we review some of the proposed multipath protocols for finding node disjoint, link disjoint, and non-disjoint paths.

Intelligent path selection can be used to enhance the performance of multipath routing. For instance, a certain subset of paths may be selected for use based on a variety of criteria such as characteristics of the paths and interactions with the link layer. From a fault tolerance perspective, more reliable paths should be selected to reduce the chance of routes failures. Path selection also plays an important role for QoS routing. In QoS routing, only a subset of paths that together satisfies the QoS requirement is selected.

After a source begins sending data along multiple routes, some or all of the routes may break due to node mobility and/or link and node failures. As in unipath routing, route maintenance must be performed in the presence of route failures. As discussed previously for unipath routing, route discovery can be triggered upon failure of the route. In the case of multipath routing, route discovery can be triggered each time one of the routes fails or only after all the routes fail. Waiting for all the routes to fail before performing a route discovery would result in a delay before new routes are available. This may degrade the QoS of the application. However, initiating route discovery every time one of the routes fails may incur high overheads. Performing route discovery when N routes fail, where N is less than the number of paths available, may be a compromise between the two options.

Split Multipath Routing. Split Multipath Routing (SMR) proposed in [12] is an on-demand multipath source routing protocol. SMR is similar to DSR, and is used to construct maximally disjoint paths. Unlike DSR, intermediate nodes do not keep a route cache, and therefore, do not reply to RREQs. This is to allow the destination to receive all the routes so that it can select the maximally disjoint paths. Maximally disjoint paths have as few links or nodes in common as possible. Duplicate RREQs are not necessarily discarded. Instead, intermediate nodes forward RREQs that are received through a different incoming link, and whose hop count is not larger than the previously received RREQs. The proposed route selection algorithm only selects two routes. However, the algorithm can be extended to select more than two routes. In the algorithm, the destination sends an RREP for the first RREQ it receives, which represents the shortest delay path. The destination then waits to receive more RREQs. From the received RREQs, the path that is maximally disjoint from the shortest delay path is selected. If more than one maximally disjoint path exists, the shortest hop path is selected. If more than one shortest hop path exists, the path whose RREQ was received first is selected. The destination then sends an RREP for the selected RREQ.

AOMDV. AOMDV [13] is an extension to the AODV protocol for computing multiple loop-free and link-disjoint paths. To keep track of multiple routes, the routing entries for each destination contain a list of the next-hops along with the corresponding hop counts. All the next hops have the same sequence number. For each destination, a node maintains the advertised hop count, which is defined as the maximum hop count for all the paths. This is the hop count used for sending route advertisements of the destination. Each duplicate route advertisement received by a node defines an alternate path to the destination. To ensure loop freedom, a node only accepts an alternate path to the destination if it has a lower hop count than the advertised hop count for that destination. Because the maximum hop count is used, the advertised hop count therefore does not change for the same sequence number. When a route advertisement is received for a destination with a greater sequence number, the next-hop list and advertised hop count are reinitialized.

AOMDV can be used to find node-disjoint or link-disjoint routes. To find node-disjoint routes, each node does not immediately reject duplicate RREQs. Each RREQ arriving via a different neighbor of the source defines a node-disjoint path. This is because nodes cannot broadcast duplicate RREQs, so any two RREQs arriving at an intermediate node via a different neighbor of the source could not have traversed the same node. In an attempt to get multiple link-disjoint routes, the destination replies to duplicate RREQs regardless of their first hop. To ensure link-disjointness in the first hop of the RREP, the destination only replies to RREQs arriving via unique neighbors. After the first hop, the RREPs follow the reverse paths, which are node-disjoint and thus link-disjoint. The trajectories of each RREP may intersect at an intermediate node, but each takes a different reverse path to the source to ensure link-disjointness.

AODVM. AODVM [11] is an extension to AODV for finding multiple node-disjoint paths. Intermediate nodes are not allowed to send a route reply directly to the source. Also, duplicate RREQ packets are not discarded by intermediate nodes. Instead, all received RREQ packets are recorded in an RREQ table at the intermediate nodes. The destination sends an RREP for all the received RREQ packets. An intermediate node forwards a received RREP packet to the neighbor in the RREQ table that is along the shortest path to the source. To ensure that nodes do not participate in more than one route, whenever a node overhears one of its neighbors broadcasting an RREP packet, it deletes that neighbor from its RREQ table. Because a node cannot participate in more than one route, the discovered routes must be node-disjoint.

Comparison of Unipath and Multipath Routing. The main advantage of DSR over AODV is its simplicity. In DSR, while nodes do maintain route caches, they do not need to maintain routing tables with forwarding information, as in AODV. However, with DSR, more overhead is incurred in routing data packets, since the entire route must be specified in the packet header. The multipath extensions to DSR and AODV inherit these advantages and disadvantages from their parent protocols. Therefore, the main advantage of SMR, and any multipath extension to DSR, is simplicity.

Both AODV and DSR allow for intermediate nodes to respond to RREQs, which can reduce the time for route discoveries. However, both SMR and AODVM do not allow intermediate nodes to reply to route discoveries, in order to ensure that the destination can select disjoint paths. The primary advantage of AOMDV is that it allows intermediate nodes to reply to RREQs, while still selecting disjoint paths. All three multipath protocols do not reject duplicate RREQs at intermediate nodes. This allows for the discovery of more paths. However, it also results in more message overhead during route discovery due to increased flooding. Additionally, in the multipath protocols the destination replies to multiple RREQs, which results in more overhead. However, in SMR and AOMDV, the destination only replies to a subset of the received RREQs.

The primary disadvantages of multipath routing protocols compared to unipath protocols are complexity and overhead. In the case of multipath extensions

to AODV, maintaining multiple routes to a destination results in larger routing tables at intermediate nodes. In multipath routing, as discussed in the next section, the method by which packets are allocated to the multiple routes must be taken into account. Multipath routing can result in packet reordering. In unipath routing, traffic allocation is not an issue, since only one route is used.

Traffic Allocation. Once the source node has selected a set of paths to the destination, it can begin sending data to the destination along the paths. The traffic allocation strategy used deals with how the data is distributed amongst the paths. The choice of allocation granularity is important in traffic allocation. The allocation granularity specifies the smallest unit of information allocated to each path. For instance, a per-connection granularity would allocate all traffic for one connection to a single path. A per-packet granularity would distribute the packets from multiple connections amongst the paths. A per-packet granularity results in the best performance [14]. This is because it allows for finer control over the network resources. It is difficult to evenly distribute traffic amongst the paths in the per-connection case, because all the connections experience different traffic rates. If a round-robin traffic allocation approach is used, however, a per-packet granularity may result in packets arriving out-of-order at the destination. Packet reordering is an issue that needs to be dealt with in multipath routing, possibly at the transport layer.

4.3 Link Layer Issues

Nodes in an ad hoc network communicate through the wireless medium. If a shared channel is used, neighboring nodes must contend for the channel. When the channel is in use by a transmitting node, neighboring nodes hear the transmission and are blocked from receiving from other sources. Furthermore, depending on the link layer protocol, neighboring nodes may have to defer transmission until the channel is free. Even when multiple channels are used, the quality of neighboring transmissions may be degraded due to interference. Nodes within transmission range of each other are said to be in the same collision domain.

We now consider the situation where multipath routing is used and the multiple paths are used simultaneously to route data. Even if the multiple paths are node-disjoint, transmissions along the routes may interfere if some nodes among the routes are in the same collision domain. While node-disjointness may ensure failure independence, it does not ensure transmission independence. When choosing multiple paths, it is important to choose paths that are as independent as possible to ensure the least interference between the paths.

Multiple metrics can be used to calculate the relative degree of independence among a set of paths, namely correlation [15] and coupling [16]. The correlation factor between two node-disjoint paths is defined as the total number of links connecting the paths [15]. Note that the correlation factor only applies to node-disjoint paths. The coupling between two paths is calculated as the average number of nodes that are blocked from receiving data along one of the paths when

a node in the other path is transmitting [16]. The advantage of using coupling as a metric is that it can be used for both disjoint and non-disjoint routes. Non-disjoint routes are considered highly coupled. Choosing paths that have low coupling or correlation can improve the performance of multipath routing.

5 Applications of Multipath Routing

Multipath routing can be used to support a variety of applications in MANETs. In this section, we present proposed multipath routing methods that support reliability (fault tolerance), energy conservation, minimization of end-to-end delay, and satisfying bandwidth requirements. Finally, we discuss an example architecture which uses source coding that utilizes multipath route discovery and maintenance, along with traffic allocation to support video over MANETs.

5.1 Supporting Reliability (Fault Tolerance)

Satisfying Reliability Requirements. MP-DSR is a multipath QoS-aware extension to DSR proposed in [17]. The protocol attempts to provide end-to-end reliability as the QoS metric. End-to-end reliability is defined as the probability of sending data successfully within a time window. The end-to-end reliability is calculated from the reliabilities of the paths used for routing. The path reliability is calculated from the link availabilities. Link availability is defined as the probability that a link is available from time $t_0 + t$, given that it is an active link at time t_0 [18]. Path reliability is the product of the link availabilities along the path, assuming the link availabilities are independent. The end-to-end reliability is $1 - \prod_{k \in K} (1 - k)$, where k is the path reliability of a path, and K is the set of all paths. Essentially, the end-to-end reliability is the probability that at least one path does not fail within the given time window. Figure 6 shows an example of how to compute the end-to-end reliability given a multipath route and the link availabilities of each path. Notice that the end-to-end reliability is higher than any of the path reliabilities.

The operation of MP-DSR is as follows. The application first supplies MP-DSR with a path reliability requirement. MP-DSR then determines the number of paths needed and lowest path reliability requirement each path must provide. The source then sends RREQ messages intended for the destination node to its neighbors containing this requirement and the end-to-end requirement. The RREQ message also contains the traversed path, and the accumulated path reliability. The intermediate nodes check to see if the RREQ message still meets the reliability requirements. If so, the node updates the accumulated path reliability based on the availability of the link just traversed, and forwards the message to its neighbors. If the RREQ message no longer meets the requirements, the message is discarded. When the destination receives all the RREQ messages, it sorts the messages according to the path reliabilities, and selects a set of disjoint paths that together satisfy the end-to-end reliability requirement. An RREP

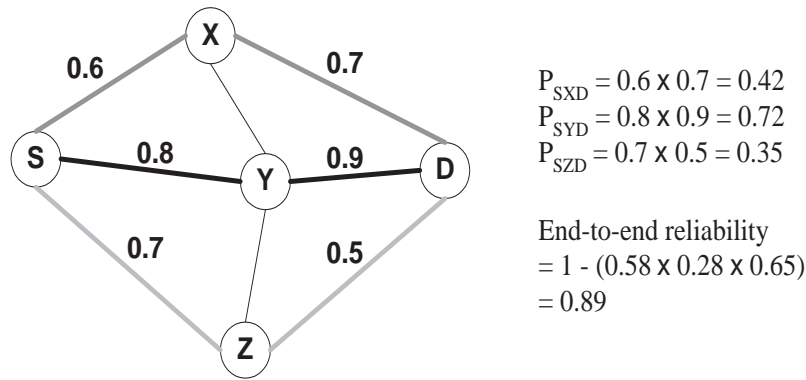


Fig. 6. Example network with multipath route consisting of three paths from S to D. The calculated path reliabilities are given to the right. The end-to-end reliability is calculated to be 0.89.

message is sent along each path back to the source. When the source receives the RREPs, it can begin using the multiple paths to route data.

Route maintenance can be performed when all routes fail or when the timer window expires. If all routes fail, the route discovery process is simply reinitiated. When the timer window expires, the source sends a route check messages along the paths to collect the path reliabilities. The destination replies to the route check messages. The source collects all the replies, and checks to see if the paths still meet the reliability requirement within a certain tolerance level. If validation is unsuccessful, then route discovery is triggered. One advantage of MP-DSR is that QoS characteristics are collected using local information available at intermediate nodes. Therefore, global knowledge is not required.

Hybrid Network for Enhanced Reliability. As discussed before, based on the results in [11], it may be difficult to find a suitable number of node disjoint paths between two nodes to provide the necessary fault tolerance and reliability. However, some ad hoc networks may contain heterogeneous nodes, where some nodes are more reliable than other nodes. For instance, in a battlefield, low-powered sensors or handhelds may be deployed in the field, with high-powered, reliable, and secure nodes located in tanks or large vehicles. A scheme is proposed in [11] that takes advantage of reliable nodes to construct a reliable path between two nodes. In this scheme, a reliable path essentially consists of reliable nodes or reliable nodes connected by multiple node disjoint paths. Specifically, a reliable path is formed by concatenating reliable segments. A segment is deemed reliable if it consists totally of reliable nodes, or if there are a certain number of disjoint paths connecting the two reliable nodes. Refer to Figure 7 for an illustration of this scheme. AODVM is used to find node-disjoint paths between the nodes.

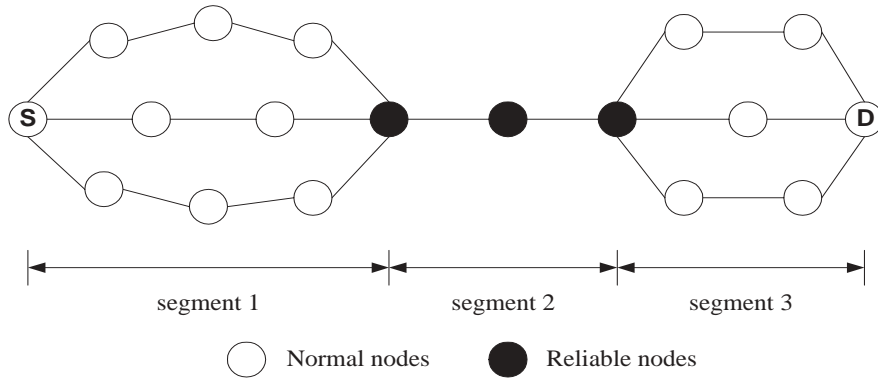


Fig. 7. Example of a reliable path between S and D [11]. The reliable path is formed by the concatenation of three reliable segments. Segment 1 and segment 3 are reliable because there are a suitable number of disjoint paths between the endpoints. In this example there must be three disjoint paths between the endpoints for the segment to be deemed reliable. Segment 2 is reliable because it consists entirely of reliable nodes.

In order to see performance gains, the reliable nodes should not be deployed randomly, especially when the number of reliable nodes is small. It is difficult to find multiple disjoint paths between two nodes because there may exist sparse areas between the nodes that act as a "bottleneck" area, as in Figure 7. Therefore, reliable nodes are deployed in these areas to provide reliable paths. The objective is to position reliable nodes such that the probability of establishing a reliable path between any two arbitrary nodes is maximized. The reliable nodes gather topology information from surrounding nodes to determine where to position themselves. Depending on the mobility of the nodes, the reliable nodes may have to reposition themselves. The reliable nodes should be faster than the other nodes, such that they can adapt to node mobility in a timely fashion. Since the reliable nodes are more powerful than other nodes, it is not unreasonable to believe that they are faster. For instance, reliable nodes may be located in large vehicles while the other nodes are carried by pedestrians. Results in [11] show that the deployment of only a few reliable nodes (10% of total nodes in the network) can result in an increase in the probability of establishing reliable paths between arbitrary nodes.

Packet Salvaging for Fault Tolerance. When using multiple paths for fault tolerance, if a source cannot send a packet due to a route failure, the packet can be routed along an alternate route. However, if only the source maintains multiple routes to the destination, when a route fails, a RERR message must propagate from an intermediate node all the way to the source before the packet can be routed along an alternate route. With packet salvaging [19], intermediate nodes maintain multiple routes to the destination, and a RERR message prop-

agates upstream only until an intermediate node can forward the packet along an alternate route. This obviously reduces the packet delay for recovering from a route failure.

A multipath extension to DSR that employs packet salvaging is proposed in [20]. In this scheme, the first RREQ received by the destination is selected as the primary route. The destination node then supplies each intermediate node in the primary route with a link disjoint route to the destination. To do this, the destination sends the RREP for a received RREQ directly to the intermediate node the RREQ traversed, instead of to the source. It is possible that not all intermediate nodes will receive an alternate route to the destination. Figure 8 depicts this scheme. Whenever an intermediate node along the primary route fails to forward a packet to the next node due to a link failure, it switches to its alternate route. It does this by replacing the unused portion of the route in the packet header with the new alternate route. For instance, referring to Figure 8, if link L_i fails, node n_i will replace $L_i - L_k$ in the packet header with P_i . When a link along P_i breaks, a RERR message is sent upstream until it reaches n_{i-1} , which then switches to path P_{i-1} . The intermediate node is subsequently responsible for modifying all packet headers to use the alternate route. When the source receives a RERR message, it means all alternate routes have failed, and therefore, a new route discovery must be performed.

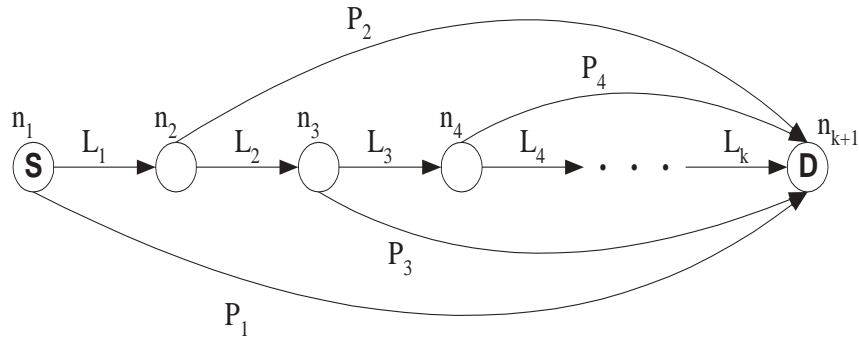


Fig. 8. An example of multipath protocol proposed in [20]. Each intermediate node n_i along the primary path has an alternate route P_i to the destination D.

The previous scheme only uses one route at a time. That is, alternate routes are only used when the current route fails. This may result in unused routes becoming stale. A similar multipath protocol called CHAMP (Caching and Multipath Routing) [19] uses round-robin traffic allocation to keep routes fresh. It also employs cooperative packet caching to improve fault tolerance. In CHAMP, the source broadcasts an RREQ when it doesn't have a route to the destination.

Intermediate nodes forward the RREQ until it reaches the destination. While receiving RREQs, an intermediate node keeps track of the neighbors that are on an equal shortest path back to the source. The destination sends an RREP if the route in the RREQ is less than or equal to the shortest route received. A node only accepts the shortest routes to the destination, and the routes must be of equal length. When routing a packet to a destination, a node sends the packet along the least used route, thereby spreading data packets over all the available routes. Routes are of equal length in order to help reduce out-of-order packets arriving at the destination. CHAMP allows for non-disjoint paths.

CHAMP takes advantage of temporal locality in routing, where a dropped packet is a recently sent packet. Each node keeps a cache of packets it recently forwarded. If a node is unable to forward a packet to the next hop neighbor along a route, the route is removed from the route cache. If an alternate route to the destination is available, the node forwards the packet along this route. If no alternate routes are available, the node broadcasts a RERR message with the packet information. When a node receives a RERR message for a particular packet, it removes the corresponding route from its route cache. It then checks to see if it has the packet in its data cache. If it has the packet cached and an alternate route to the destination is available, the packet is forwarded along this route. Otherwise, the node re-broadcasts the RERR message, and the process continues. If the source node of a packet receives a RERR message for that packet and no alternate routes to the destination are available, the source initiates a route discovery.

In both of the previous studies [19] [20], it was found that maintaining just one alternate route at each node was optimal in terms of performance. Maintaining multiple routes at intermediate nodes results in less frequent route discoveries. Both protocols demonstrate how link disjoint paths work adequately well for fault tolerance purposes.

Diversity Coding. A multipath traffic allocation scheme for ad hoc networks that uses M-for-N diversity coding is proposed in [21]. In this scheme, a packet is split up into N equal size blocks. Then, M blocks of the same size are added to the packet as overhead. The additional blocks are calculated from the N blocks, and provide redundant information. The $N + M$ total blocks are then allocated to the multiple available paths. Based on the M-for-N coding scheme, if no more than M blocks are lost, the original packet can be reconstructed from the received blocks.

The main idea behind this scheme is to allocate the blocks amongst the routes such that the probability of losing no more than M blocks is maximized. Therefore, it is reasonable to believe that more blocks should be assigned to more reliable routes. In this scheme, each path is assigned a probability of success. The paths are assumed to be total erasure channels, in which either all the data or no data sent along the channel is received at the destination. Therefore, if a path fails, all the blocks sent along that path are lost. Otherwise, all the blocks are received. An equation is developed in [21] to compute the probability

of success, or the probability that no more than M blocks are lost, based on the probabilities assigned to each path and how the blocks are allocated. The equation is maximized to determine which routes to use and how the blocks should be allocated to each route. An observation in [21] was made that in the limit as the number of paths becomes large, the probability of success is 100%.

This scheme requires the existence of a mechanism to assign path success probabilities. Mechanisms for calculating path reliability do exist [17] [18]. The coding scheme used should also be relatively fast. The amount of overhead added to each packet obviously should be less than the actual size of the packet in order to see performance gains.

5.2 Minimizing End-to-End Delay

A traffic allocation scheme is proposed in [22] that uses weighted round robin packet distribution to improve the delay and throughput. In this scheme, the source node maintains a set of multiple paths to the destination, and distributes the load amongst the paths according to the RTT of each path. Paths that have a smaller RTT are allocated more consecutive packets. It is assumed in this scheme that the RTT of each path is inversely proportional to the available bandwidth. To justify this, it is assumed that ad hoc networks have less heterogeneity than WAN, and the bandwidth-delay product is a constant. Therefore, traffic is distributed amongst the paths proportional to the available bandwidth of the paths. In addition to improving the end-to-end delay, the scheme can also improve network congestion. A multipath extension to DSR called MSR is used by the source node to find multiple paths to the destination. Disjoint paths are preferred to ensure path independence.

The scheme proposed in [23] attempts to minimize the end-to-end delay for sending a volume of data from source to destination by using multipath routing and intelligent traffic allocation. In this scheme, data is routed along multiple paths in sequential blocks. Since routes may break before all the data is transmitted, new paths must be discovered. The scheme divides the total data into chunks and uses a new set of multiple paths to route each chunk. Pre-emptive route discoveries are done to find new paths before route errors occur. In order to determine how to distribute data amongst the paths, the scheme uses a mechanism to predict the lifetime of a path. The lifetime of a path is determined by the link affinity, which is a prediction of the life span of a link. The affinity is calculated based on the transmission range of the nodes, distance between nodes, and the average velocity of the nodes. A path's stability is defined as the smallest link affinity along the path.

The route discovery process is essentially an extension to DSR. One extension to DSR is that intermediate nodes append the link affinity value to route reply packets. Therefore, when the source receives the route replies, it can determine the stability of each path. Also, the source node records the time it sends a route request and the times each route reply is received, so that it has the total time of route discovery. This time indicates the delay of the path due to congestion, packet transmission, and hop count. In the scheme, an optimization

problem is defined that minimizes the average network delay for sending the data, based on the constraints that data is distributed among multiple paths and the distributed data must reach the destination within the lifetime of those paths. Given the set of discovered stable paths with their corresponding delay and hop counts, the optimization problem selects a subset of the paths and determines how data will be distributed amongst the paths. A chunk of the total data is then distributed among the selected paths. Before transmission of the data chunk is complete, a new route discovery process is performed to find new routes. The route discovery is performed at a time such that the new routes are available just before transmission of the chunk is complete. The optimization problem is then calculated again for the new routes, and another data chunk is transmitted. The process continues until the total data volume is transmitted to the destination. This is a good example of a comprehensive scheme that employs mechanisms for route discovery, traffic allocation, and route maintenance to improve the QoS.

5.3 Satisfying Bandwidth Requirements

A bandwidth reservation scheme for ad hoc networks that uses multipath routing is proposed in [24]. The protocol attempts to find multiple paths that collectively satisfy the bandwidth requirements. The original bandwidth requirement is essentially split into multiple sub-bandwidth requirements. Each sub-path is then responsible for one sub-bandwidth requirement. This protocol is on-demand and uses the local bandwidth information available at each node for discovering routes. A ticket-based approach is used to search for multiple paths. In this approach a number of probes are sent out from the source, each carrying a ticket. Each probe is responsible for searching one path. The number of tickets sent controls the amount of flooding that is done. Each probe travels along a path that contains the necessary bandwidth.

The source initially sends a certain number of tickets each containing the total bandwidth requirement. The tickets are sent along links that contain sufficient bandwidth to meet the requirement. When an intermediate node receives a ticket, it checks to see which links have enough bandwidth to meet the requirement. If it finds some, it then chooses a link, reserves the bandwidth, and forwards the ticket on the link. If no links have the required bandwidth, the node reserves bandwidth along multiple links such that the sum of the reserved bandwidths equals the original requirement. In this way, the bandwidth requirement is split into sub-bandwidth requirements equaling the bandwidths reserved along each of the links. The original ticket is split into sub-tickets, with each sub-ticket being forwarded along one of the links. Each sub-ticket is then responsible for finding a multi-path satisfying the sub-bandwidth requirement. If no links can be found to satisfy the bandwidth requirements, the intermediate node drops the ticket. Links with more available bandwidth are preferred. An example of the route discovery process is given in Figure 9. The destination eventually receives multiple tickets or sub-tickets comprising a whole ticket. The destination chooses a single ticket, which represents a uni-path, or a group of sub-tickets comprising a whole ticket, which represents a multipath, and sends a route reply. The

route replies traverse the reverse paths taken by the sub-tickets, confirming the bandwidth reservations along the way.

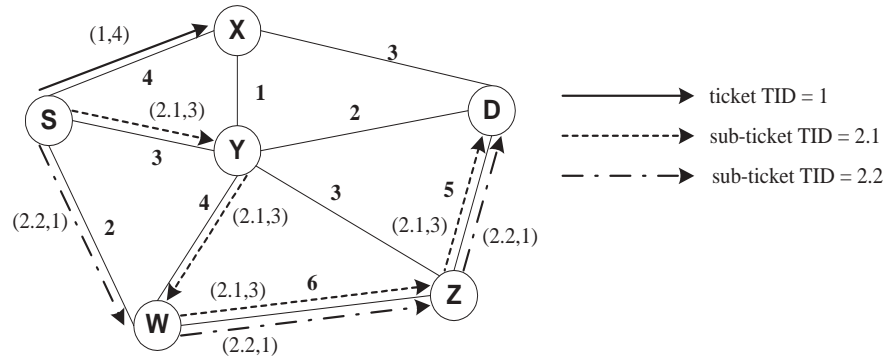


Fig. 9. Example of the route discovery process for bandwidth reservation scheme proposed in [24]. The links are labeled with their available bandwidth. Each message is labeled with the tuple (ticket ID, bandwidth requirement). Ticket IDs are assigned in a hierarchical fashion (Ticket 2.1 is a sub-ticket of ticket 2). Node S initially distributes two tickets. Node X drops ticket 1, because not enough bandwidth is available to proceed. Node S divides ticket 2 into sub-tickets 2.1 and 2.2, with sub-requirements 3 and 1 respectively. The sub-tickets merge at node W and take the same path to node D. D receives 2 paths to the destination: SYWZD and SWZD.

The above ticket-based approach does not specify how the available link bandwidth is determined. Specifically, the approach does not deal with the radio interference problem. To address this issue, a multipath protocol that uses the CDMA-over-TDMA model is proposed in [25]. In this model, bandwidth is calculated and reserved based on the free time slots of links. In this approach, the network is flooded during route discovery to search for paths. Each RREQ packet accumulates the link-state of the path it traverses. The link-state corresponds to the free time slot list for each of the traversed links. For each RREQ packet the destination receives, the maximal bandwidth available on the path is determined based on the free time slots. The destination keeps track of the accumulated bandwidth for the paths it has discovered. Once the destination receives enough paths to satisfy the bandwidth requirement, it can send the route replies and reserve the time slots along the paths.

In the time-slotted scheme, intermediate nodes do not drop the RREQ packets, as in the ticket-based scheme. The time-slotted approach accumulates the link-state of the entire path, and the destination can construct a partial network topology with corresponding link state in an on-demand fashion. The ticket-based approach makes hop-by-hop decisions based on only local link-state, and intermediate nodes drop RREQ packets when not enough bandwidth is avail-

able on the local links. Because intermediate nodes do not drop RREQs, the time-slotted approach can potentially find more paths. The bandwidth allocation along the paths can also be maximized, since the link-state is available for the entire path. However, these advantages come at the expense of higher message overhead, because the network is flooded and RREQs must store the entire path link-state. Both approaches allow for non-disjoint paths.

5.4 Minimizing Energy Consumption

Since a MANET may consist of nodes which are not able to be re-charged in an expected time period, energy conservation is crucial to maintaining the life-time of such a node. In networks consisting of these nodes, where it is impossible to replenish the nodes' power, techniques for energy-efficient routing as well as efficient data dissemination between nodes is crucial. An energy-efficient mechanism for unipath routing in sensor networks called directed diffusion has been proposed in [26]. Directed diffusion is an on-demand routing approach. In directed diffusion, a (sensing) node which has data to send periodically broadcasts it. When nodes receive data, they send a reinforcement message to a pre-selected neighbor which indicates that it desires to receive more data from this selected neighbor. As these reinforcement messages are propagated back to the source, an implicit data path is set up; each intermediate node sets up state that forwards similar data towards the previous hop.

Directed diffusion is an on-demand routing approach; it was designed for energy efficiency so it only sets up a path if there is data between a source and a sink. However, the major disadvantage of the scheme, in terms of energy-efficiency, is the periodic flooding of data. In order to avoid the flooding overhead, [27] proposes the setup and maintenance of alternate paths in advance using a localized path setup technique based upon the notion of path reinforcement. The goal of a localized reinforcement-based mechanism is for individual nodes to measure short term traffic characteristics and choose a primary path as well as a number of alternate paths based upon their empirical measurements. An alternate path is intended to be used when the primary fails. "Keep-alive" data is sent through the alternate paths even when the primary path is in use. Because of this continuous "keep-alive" data stream, nodes can rapidly switch to an alternate path without going through a potentially energy-depleting discovery process for a new alternate path.

A multipath routing technique which uses braided multipaths is also proposed in [27]. Braided multipaths relax the requirement for node disjointness. Multiple paths in a braid are only partially disjoint from each other and are not completely node-disjoint. These paths are usually shorter than node-disjoint multipaths and thus consume less energy resources; alternate paths should consume an amount of energy comparable to the primary path. A simple localized technique for constructing braids is as follows. A source sends out a primary path reinforcement to its (primary path) neighbor as well as alternate path reinforcements to its (alternate path) neighbors. Each node in the network performs this same neighbor and path selection process. The evaluation of the performance of

the proposed energy constrained algorithms is a function of the overall goal of minimizing energy resources. It was found that energy-efficient multipath routing using the braided multipath approach expends only 33% of the energy of disjoint paths for alternate path maintenance in some cases, and have a 50% higher resilience to isolated failures.

5.5 Example Application: Source Coding for Video Support

A system for transporting video over ad hoc networks using multipath routing and source coding is proposed in [28]. The system considers two types of source coding: *multiple-description coding* and *layered coding*. Both types of coding create multiple sub-streams out of a single video stream. In multiple-description coding, each sub-stream has equal importance, and therefore, each sub-stream contributes equally to the quality level. In layered coding, a base layer stream is created along with multiple enhancement layer streams. The base layer stream provides a basic level of quality, while each enhancement layer stream if correctly received adds to the quality. Therefore, in layered coding, base layer packets should be sent along more reliable paths to ensure that they are received at the destination.

In the system, the source maintains multiple paths to the destination and reserves bandwidth along the paths such that the total bandwidth falls within an acceptable range. The total number of paths is not necessarily equal to the number of streams. Therefore, a path may carry packets from different streams. Similarly, packets from one stream may be allocated to different paths. The task of the source is to allocate the packets from each stream among the paths such that a minimum level of quality can be observed at the receiver. Depending on the path conditions and application requirements, the source chooses to use multiple-description coding or layered coding. The source coder also must adjust the rate allocation to each stream depending on the available bandwidth. Using intelligent path selection and traffic allocation along with adaptive source coding, the system can adapt well to fluctuating network conditions caused by path failures or changes in available bandwidth. Refer to Figure 10 for a depiction of the system architecture.

A scheme to provide reliable transport for video specifically using layered coding along with multipath routing is proposed in [29]. In the proposed scheme, the video data is encoded into two layers: the base layer and one enhancement layer. The source uses two disjoint paths to the destination to route data. Base layer packets are sent along one path, while enhancement layer packets are sent along the other path. Base layer packets are protected using Automatic Repeat Request (ARQ). When a base layer packet is lost, the destination sends an ARQ request to the source. When an ARQ request is received, the source retransmits the base layer packet along the enhancement layer path to ensure timely arrival of the base layer packet, and the enhancement layer packet being transmitted at that time instance is discarded. The necessary bandwidth for the base layer and enhancement layer paths are reserved using a signaling protocol. This scheme is depicted in Figure 11.

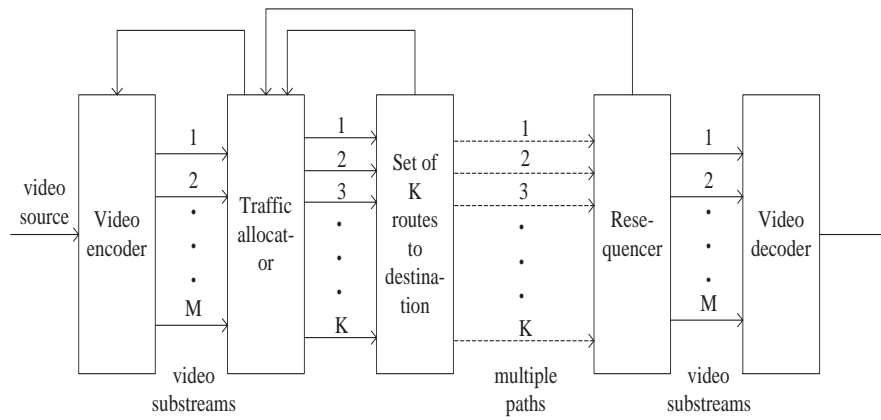


Fig. 10. Architecture for multipath source coding system proposed in [28].

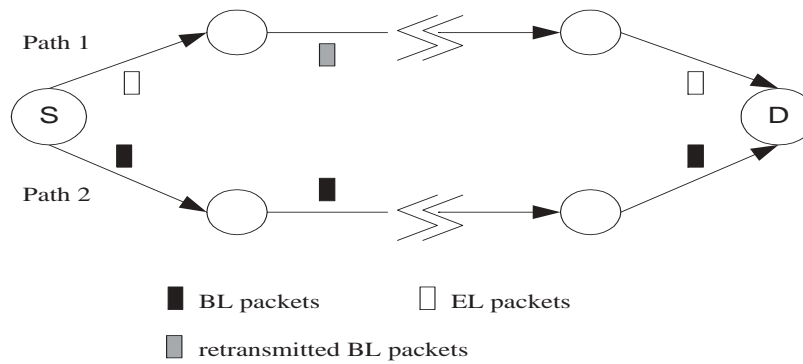


Fig. 11. Multipath layered coding scheme proposed in [29]. Base layer (BL) packets are sent along path 2, while enhancement layer (EL) packets and retransmitted BL packets are sent along Path 1.

As packet drops increase due to congestion or degraded network conditions, enhancement layer packets are dropped at the source in favor of retransmitted base layer packets. This attempts to ensure that a basic level of quality is always achieved at the destination. Using layered coding along with ARQ would work well when using lossy paths and if the extra delay for retransmissions is acceptable.

6 Future Work

In this paper, we have presented multipath routing in ad hoc networks mostly in terms of the network layer. We have not made mention of the interaction of multipath routing with the transport layer, in particular, TCP. The main issue that must be dealt with at the transport layer is the arrival of out-of-order packets when multiple paths are used in a round-robin fashion. In TCP, out-of-order packets are assumed to signal congestion in the network, at which point TCP reduces its window size. This can have a detrimental effect on the overall throughput seen by TCP connections. Therefore, the implementation of a TCP-friendly multipath protocol is necessary. We have discussed CHAMP, which uses equal length paths to reduce out-of-order packets. However, finding only equal length paths puts a restriction on the number of paths you can find. If unequal paths are chosen, there could also be ways to perform intelligent traffic allocation depending on path lengths and path delays such that out-of-order packets are minimized. For instance, sending later packets over shorter paths and earlier packets over longer paths may result in reduced out-of-order packets at the receiver. This implies intelligently sending packets out-of-order such that they arrive in-order at the receiver.

In our discussion of using multipath routing to support QoS, most of the protocols proposed only provide QoS in terms of specific metrics, such as bandwidth, delay, or reliability. However, it may be necessary to develop mechanisms to support QoS in terms of multiple metrics. For instance, when searching for multiple paths that have the required bandwidth, it is desirable to find reliable paths. Given the faulty nature of MANETs, constructing a multipath route that meets the bandwidth requirements while also meeting certain reliability requirements would result in better performance. Also, the mechanisms proposed for supporting QoS in terms of delay only attempt to minimize or improve on the delay. It would be desirable to develop a multipath protocol that can provide delay bounds or guarantees, which are required by some real-time applications. Using multipath routing to provide adaptive QoS using source coding is also a promising technique that can be expanded upon for applications other than video.

References

1. Hedrick, C.: Routing Information Protocol. Internet Request For Comments 1058 (1988)

2. Moy, J.: OSPF Version 2. Internet Request For Comments 1247 (1991)
3. Perkins, C.E., Royer, E.M.: Ad-hoc On-Demand Distance Vector Routing. Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (1999)
4. Johnson, D.B., Maltz, D.A.: Dynamic Source Routing in Ad Hoc Wireless Networks. *Mobile Computing*. (1996) 153-181
5. Ash, G.H., Kafker, A.H., Krishnan, K.R.: Servicing and Real-Time Control of Networks with Dynamic Routing. *Bell System Technical Journal*, Vol. 60, No. 8 (1981)
6. Gibbons, R.J., Kelley, F.P., Key, P.B.: Dynamic Alternative Routing - Modelling and Behavior. Proceedings of the 12th International Teletraffic Conference (1988)
7. ATM Forum. ATM user-network interface specification, Version 3.0.
8. Murthy, S., Garcia-Luna-Aceves, J.J.: An Efficient Routing Protocol for Wireless Networks, *Mobile Networks and Applications*, Vol. 1, No. 2 (1996)
9. Perkins, C.E., Bhagwat, P.: Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. *ACM SIGCOMM* (1994) 234-244
10. Pham, P.P., Perreau, S.: Performance Analysis of Reactive Shortest Path and Multi-path Routing Mechanism With Load Balance. *IEEE INFOCOM* (2003)
11. Ye, Z., Krishnamurthy, S.V., Tripathi, S.K.: A Framework for Reliable Routing in Mobile Ad Hoc Networks. *IEEE INFOCOM* (2003)
12. Lee, S.-J., Gerla, M.: Split Multipath Routing with Maximally Disjoint Paths in Ad Hoc Networks. *IEEE International Conference on Communications*, Vol. 10 (2001)
13. Marina, M.K., Das, S.R.: On-demand Multipath Distance Vector Routing in Ad Hoc Networks. Proceedings of the International Conference for Network Protocols (2001)
14. Krishnan, R., Silvester, J.A.: Choice of Allocation Granularity in Multipath Source Routing Schemes. *IEEE INFOCOM* (1993)
15. Wu, K., Harms, J.: Performance Study of a Multipath Routing Method for Wireless Mobile Ad Hoc Networks. Proceedings of Symposium on Modeling, Analysis and Simulation on Computer and Telecommunication Systems (2001) 99-107
16. Pearlman, M.R., Haas, Z.J., Sholander, P., Tabrizi, S.S.: On the Impact of Alternate Path Routing for Load Balancing in Mobile Ad Hoc Networks. Proceedings of the ACM MobiHoc, (2000) 3-10
17. Leung, R., Liu, J., Poon, E., Chan, A.-L.C., Li, B.: MP-DSR: A QoS-aware Multipath Dynamic Source Routing Protocol for Wireless Ad-hoc Networks. Proceedings of the 26th IEEE Annual Conference on Local Computer Networks (2001) 132-141
18. Jiang, S., He, D., Rao, J.: A Prediction-based Link Availability Estimation for Mobile Ad Hoc Networks. *IEEE INFOCOM* (2001)
19. Valera, A., Seah, W.K.G, Rao, S.V.: Cooperative Packet Caching and Shortest Multipath Routing in Mobile Ad Hoc Networks. *IEEE INFOCOM* (2003)
20. Nasipuri, A., Das, S.R.: On-Demand Multipath Routing for Mobile Ad Hoc Networks. Proceedings of the 8th International Conference on Computer Communications and Networks (1999)
21. Tsirigos, A., Haas, Z.J.: Multipath Routing in the Presence of Frequent Topological Changes. *IEEE Communications Magazine*, Vol. 39, No. 11 (2001)
22. 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)

23. Das, S.K., Mukherjee, A., Bandyopadhyay, S., Paul, K., Saha, D.: Improving Quality-of-Service in Ad hoc Wireless Networks with Adaptive Multi-path Routing. IEEE Global Telecommunications Conference (2000)
24. Liao, W.-H., Tseng, Y.-C., Wang, S.-L., Sheu, J.-P.: A Multi-path QoS Routing Protocol in a Wireless Mobile Ad Hoc Network. IEEE International Conference On Networking (2001)
25. Chen, Y.-S., Tseng, Y.-C., Shue, J.-P., Kuo, P.-H.: On-Demand, Link-State, Multi-Path QoS Routing in a Wireless Mobile Ad-Hoc Network. Proceedings of European Wireless (2002)
26. Intanagonwivat, C., Govindan, R., Estrin, D.: Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. Proceedings of ACM Mobicom (2000)
27. Ganesan, D., Govindan, R., Shenker, S., Estrin, D.: Highly Resilient Energy Efficient Multipath Routing in Wireless Sensor Networks. Mobile Computing and Communications Review, Vol. 1, No. 2 (2002)
28. Wang, Y., Panwar, S., Lin, S., Mao, S.: Wireless Video Transport Using Path Diversity: Multiple Description vs. Layered Coding. Proceedings of the IEEE International Conference on Image Processing (2002)
29. Mao, S., Lin, S., Panwar, S.S., Wang, Y.: Reliable Transmission of Video Using Automatic Repeat Request and Multi-path Transport. IEEE Vehicular Technology Conference (2001)