# SUMMARY OF ROUTING PROTOCOL IN MOBILE AD HOC NETWORKS

YI Jiazi

Polytechnic School of University of Nantes

yi.jiazi@gmail.com

Feb. 2007

École polytechnique de l'université de Nantes

UNIVERSITE DE NANTES

# CONTENTS

# Abstract

Mobile ad hoc networks (MANETs) consist of a collection of wireless mobile nodes which dynamically exchange data among without reliance on a fixed base station or a wired backbone network, which makes routing a crucial issue to the design of a MANET. In this summary, we examined the main protocols and issues of routing in MANETs and especially discussed the multipath protocol.

KEY WORDS: ad hoc networks, routing protocol, unipath, multipath

# 1. Introduction

This summary is based on the work of February, which is focused on reading papers about Ad hoc network and getting familiar with the domain before the research of the Master Subject "Multi-path Routing Protocol for Mobile Ad Hoc Network".

Mobile Ad hoc network (MANET), also called self-organized network, is a multi-hop wireless network where nodes can move arbitrary in the topology. It consists of a set of wireless nodes which dynamically exchange data among themselves without reliance on some fixed base stations or a wired backbone network. It has great differences between the wired network, including the unpredictability of environment, the unreliability of wireless medium, the resource-constrained nodes, the dynamic topology, limited bandwidth, and limited security [6].

Because of the features of the MANET, the research of the routing protocol has been one of the most concerned topics in the MANET. The conventional routing algorithms for wired networks are not efficient for the dynamic changes. For the recent years, people have developed a lot of routing protocol which can be used in MANET, and here some typical protocols are summarized.

The rest of the summary is organized as follows. In Section 2, the uni-path routing protocol is introduced, which includes reactive routing, proactive routing and hybrid routing. We introduced some multi-path routing protocols in Section 3. Most of them is developed from one of the uni-path routing protocols, which multi-path is implied to improve the reliability, the end-to-end delay or bandwidth, etc. Section 4 is some issues to be discussed or to be solved for multi-path routing. Sections 5 is some ideas and consideration about the multi-path routing, and in Section 6, some simulation problems and NS2 is discussed.

# 2. Uni-path Routing Protocol

Routing Protocols are used to discover and maintain routes between the source and destination nodes. For MANET, there are two main kinds of the routing protocol: on-demand protocols (also called reactive protocols) and table-based protocols (also called proactive protocols) [1] ~ [5].

For reactive protocols, nodes only compute routes when they are needed. Usually, caches are used to reduce the effort of route discovery[1][2][4].

For proactive protocols, 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.

What's more, some hybrid protocols are proposed. This is because both proactive and reactive routing have specific advantages and disadvantages that make them suitable for certain kinds of scenarios. The hybrid methods try to take the advantages of those two and achieve better performance.

In the rest part of this section, some typical protocols are introduced.

## 2.1. Reactive Routing

Because reactive routing only tries to find a route when necessary, it is believed that it is 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. Generally, it consists of the following two main phases[6]:
***Route discovery***: it is the process of finding a route between two nodes, whether directly reachable within wireless transmission range or reachable through one or more intermediate network hops through other hosts.
***Route maintenance***: it is the process of repairing a broken route or finding a new route in the presence of a route failure. Route maintenance procedure monitors the operation of the route and informs the sender of any routing errors.
Two of the most widely used reactive routing protocols: DSR and ADOV are introduced.

### 2.1.1. Dynamic Source Routing Protocol (DSR)

*Source routing* is a routing technique in which the sender of a packet determines the complete sequence of nodes through which to forward the packet; the sender of a packet determines he complete sequence of nodes through which to forward the packet; the sender explicitly lists this route in the packet's header, identifying each forwarding "hop" by the address of the next node to which to transmit the packet on its way to the destination host.

The DSR is designed for use in the wireless environment of an ad hoc network. There is no periodic router advertisement in the protocol. Instead, when a host needs a route to another host, it dynamically determines one based on cached information and on the results of a *route discovery* protocol [1][2].

#### 2.1.1.1. Route Discovery

When some node S originates a new packet destined to some other node D, it places in the header of the packet a source route giving the sequence of hops that the packet should follow on its way to D. Normally, S will obtain a suitable source route by searching its Route Cache of route previously learned. But if no route is found in its cache, it will initiate the *Route Discovery* protocol to dynamically find a new route to D. In this case, we call S the initiator and D the target of Route Discovery.

For instance, node A is attempting to find a route to node E. A transmits a ROUTE REQUEST message as a single local broadcast packet, which contains a unique *request id*. When another node receives a ROUTE REQUEST, if it is the target of the Route Discovery, it returns a ROUTE REPLY message to the initiator. If not, this node appends its own address to the route record in the ROUTE REQUEST message and propagates it by transmitting it as a local broadcast packet, until the packet reaches its destination(node E in this example).

When the packet gets to the destination, the node will returns a ROUTE REPLY message. The way the ROUTE REPLY message is returned can be starting a new Route Discovery or just using the reversed sequence of the nodes depends on the networks situation(unidirectional or bidirectional).

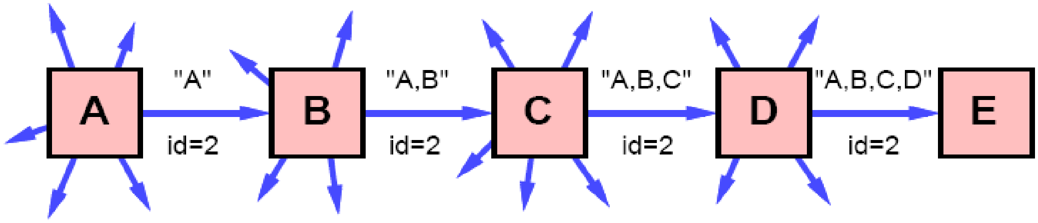



Figure 1 Route Discovery example: from Node A to Node E

When initiating a Route Discovery, the sending node saves a copy of the original packet in a local buffer called the *Send Buffer*. The Send Buffer contains a copy of each packet that cannot be transmitted by this node because it does not yet have a source route to the packet's destination. While a packet remains in the Send Buffer, the node should occasionally initiate a new Route Discovery.

There are some additional features introduced in [1] to improve the performance of Route Discovery:

Caching Overheard Routing Information;
Replying to ROUTE REQUESTS using Cached Routes;
Preventing ROUTE REPLY Storms;
ROUTE REQUEST Hop Limits.

#### 2.1.1.2. Route Maintenance

When originating or forwarding a packet using a source route, each node transmitting the packet is responsible for confirming that the packet has been received by the next hop along the source route; the packet is retransmitted (up to a maximum number of attempts) until this confirmation of receipt is received.

If the packet is retransmitted by some hop the maximum number of times and no receipt confirmation is received, this node returns a ROUTE ERROR message to the original sender of the packet, identifying the link over which the packet could not be forwarded. The source node then remove this broken link from its cache, and use another route in the Route Cache (if has one), or starts a new Route Discovery.

There are also some additional features introduced in [1] to improve the performance of Route Maintenance:

Packet Salvaging;
Automatic Route Shortening;
Increased Spreading of ROUTE ERROR Messages;
Caching Negative Information

For DSR, there are several advantages: First, unlike conventional routing protocols, it uses no periodic routing advertisement messages, thereby reducing network bandwidth overhead, particularly during periods when little or no significant host movement is taking place. Second, does not require transmissions between hosts to work bidirectionally. And what’s more, it is able to adapt quickly to changes such as host movement, yet requires no routing protocol overhead during periods in which such changes do not occur.

However, The Route Maintenance protocol does not locally repair a broken link. The broken link is only communicated to the initiator. The DSR protocol is only efficient in MANETs with less then 200 nodes. Problems appear by fast moving of more hosts, so that the nodes can only move around in this case with a moderate speed. Flooding the network can cause collusions between the packets. Also there is always a small time delay at the beginning of a new connection because the initiator must first find the route to the target [9].

### 2.1.2. Ad-hoc On-Demand Distance Vector Routing (AODV)

The Ad Hoc On-Demand Distance Vector Routing Protocol is a reactive routing protocol based on DSDV [5]. AODV uses a broadcast route discovery mechanism, as is also sued in the DSR algorithm. Instead of source routing, however, AODV relies on dynamically establishing route table entries at intermediate nodes. Opposed to DSR, which uses source routing, AODV uses hop-by-hop routing by maintaining routing table entries at intermediate nodes.

#### 2.1.2.1. Route Discovery

As in DSR, 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. The source node floods the network with a RREQ packet specifying the destination for which the route is requested. When the destination receives the RREQ packet, the node generates a 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.

#### 2.1.2.2. 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.

The advantages of AODV are: loop free routing, optional multicast and reduced control overhead. But in the same time, delay is caused by route discovery process and bidirectional connection needed in order to detect a unidirectional link[10].

## 2.2. Proactive Routing

In Proactive Routing, also called table-driven Routing, routes are calculated before one is needed. The protocol tries to keep routing information to all nodes every time up-to-date. The update of the tables can be periodically or driven by events.

### 2.2.1. Optimized Link State Routing Protocol (OLSR)

In [3], a proactive routing protocol, called Optimized Link State Routing for MANET is proposed. The protocol inherits the stability of the link state algorithm. Due to its proactive nature, it has an advantage of having the routes immediately available when needed. OLSR is an optimization of a pure link state protocol for MANET. First, it reduces the size of control packets: instead of all links, it declares only a subset of links with its neighbors who are its *multipoint relay selectors*. Secondly, it minimizes flooding of this control traffic by using only the selected nodes, called *multipoint relays (MPR)*, to diffuse its message in the network. This technique significantly reduces the number of retransmissions in a flooding or broadcast procedure, as shown in Figure 2.

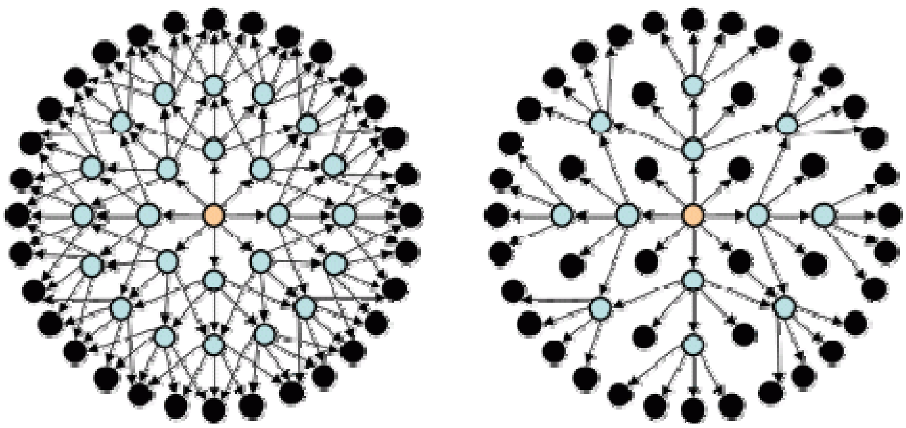


Figure 2 Multi point relays (light blue node in the right picture)

Only the MPRs of a node retransmit the packet from the node. For this purpose, each node maintains a set of its neighbors which are called MPR Selectors of the node.

The protocol functioning includes:

***Neighbor sensing***: Each node periodically broadcast its HELLO messages about its neighbor and their link status. These control messages are only one-hop and permit each node to learn the knowledge of its neighbors up to two hops.

***Multipoint Relay selection***: The MPR set is calculated in a manner to contain a subset of one hop neighbors which covers all the two hop neighbors.

***MPR information declaration***: In order to build the intra-forwarding database needed for routing packets, each broadcasts specific control messages called Topology Control (TC) messages. A TC message is sent periodically by each node in the network to declare its MPR Selector set. Each node of the network maintains a topology table, in which it records the information about the topology of the network obtained from the TC messages.

***Routing table calculation***: Each node maintains a routing table which allows it to route the packets for other destinations in the network. Because the routing table is based on the information contained in the neighbor table and the topology table, if any these tables is changed, the routing table is re-calculated to update the route information.

Figure 3 displays an overview of the information repositories in OLSR and relations to message processing, message generation and route calculation [4].

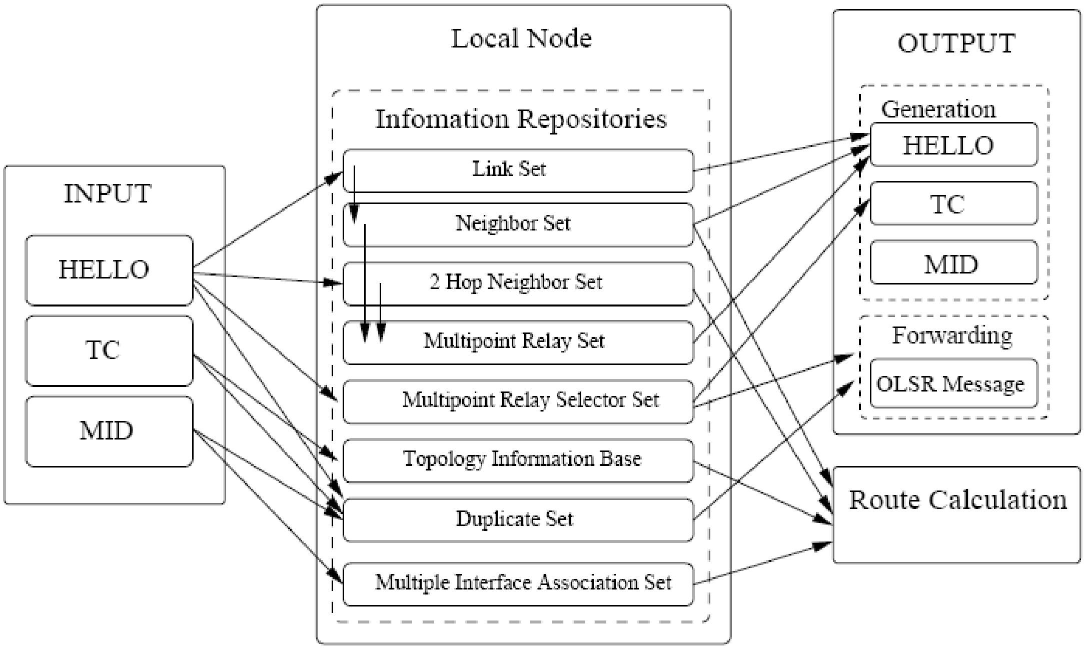



Figure 3: OLSR information repositories relation overview

OLSR has minimal latency, and is ideal in density and large networks. OLSR can easily be extended to QoS monitoring by including bandwidth and channel quality information in link state entries. Thus, the quality of the path (e.g., bandwidth, delay) is known prior to call setup.

But OLSR has high control overhead, high computation, and it is complex to implement.

## 2.3. Hybrid Routing

As explained above, both a purely pro-active and purely reactive approaches to implement a routing protocol for a MANET have their disadvantages. The hybrid routing scheme, try to take advantage of both of them. Normally, it use pro-active discovery within a node's local neighborhood, and using a reactive protocol for communication between these neighborhoods.

### 2.3.1. Zone Routing Protocol (ZRP)

In [7], a hybrid routing, called Zone Routing Protocol is proposed. ZRP divides the topology into zones and seek to utilize different routing protocols within and between the zones based on the weaknesses and strengths of these protocols. ZRP is totally modular, meaning that any routing protocol can be used within and between zones.

ZRP refers to the locally proactive routing component as the IntrA-zone Routing Protocol (IARP). The globally reactive routing component is named IntEr-zone Routing Protocol (IERP). IERP and IARP are not specific routing protocols.

The fact that the topology of the local zone of each node is known can be used to reduce traffic when global route discovery is needed. Instead of broadcasting packets, ZRP uses a concept called bordercasting. Bordercasting utilizes the topology information provided by IARP to direct query request to the border of the zone. The bordercast packet delivery service is provided by the Bordercast Resolution Protocol (BRP).

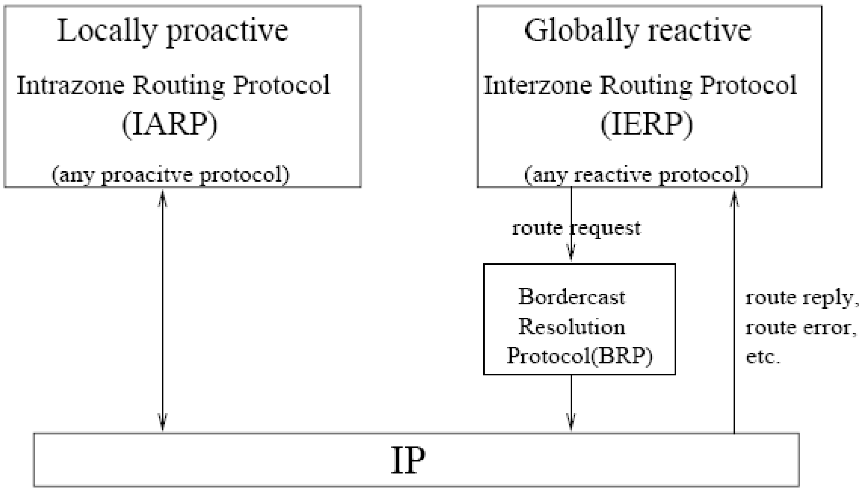



Figure 4: The different components of the ZRP

The performance evaluation in [11] shows that the traffic load, the mobile node mobility and the network density all have impact on the performance of the routing protocol. The proactive protocol offers better performances for CBR (Constant Bit Rate) sources given that it guaranties lowest delay and jitter. But it consumes more bandwidth. And when the mobility is low, the reactive protocol performances low delay and overhead. The hybrid ZRP is actually more of a framework than a routing protocol, and it relies on well defined and robust routing protocols to be utilized in and between the zones. The latest ZRP Internet draft expired January 2003, but work is still said to be done by the authors and others. The need for solutions like ZRP might arise when the basic protocols are well tested and their limitations have been proven.

# 3. Multi-path Routing Protocol

The routing protocols introduced above are uni-path routing. Based on those protocols, more and more multi-path routing protocols are proposed. These protocols consist 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.

The multi-path routing could offer several benefits: load balancing, fault-tolerance, higher aggregate bandwidth, lower end-to-end delay, etc. The rest of this section is going to introduce several multi-path routing protocol that have been proposed.

## 3.1. Alternative Path Routing (APR)

In [12], Alternate Path Routing (APR) is proposed. APR had its origins in the traditional circuit-switched telephone networks, where it reduced call blocking by providing multiple network routes for the initial call-setup messaging.

APR's ability to provide a load balancing and enhanced survivability makes it an attractive technique for bandwidth-limited MANETs that are designed as packet-radio extensions to the wired Internet. However, the APR performance gains achieved on the wired Internet do not necessarily carry over to MANETs. In particular, the overlapping radio-coverage of neighboring nodes can result in strong interdependence between alternate routes which limits APR's benefits to particular MANET topologies and channel access techniques.

APR is based on the ZRP. A link state version of the ZRP's IARP can provide each node with up-to-date connectivity within its routing zone. The IERP provides routes, as needed, for destinations that lie outside of the routing zone. If a route exists, there will be a reply explicitly defining a unique path to the destination. The ZRP Route-Reply messages effectively provide a partial snapshot of the network topology. From the perspective of APR route-set construction, the most value can be obtained from the route-replies by decomposing the reported routes into a collection of links. The links returned from the route-query can be pooled with valid links from other rout-queries, and the proactively-tracked links form within the source node's local routing zone, in order to maximize the effective routing information. As shown in Figure 5.

The potential benefits of APR make it appear to be an ideal candidate for the bandwidth limited and dynamic mobile ad-hoc networks. The investigation of APR in the MANET environment has revealed that APR can, in some circumstances, provide notable improvements to end-to-end capacity. Quite often, however, the network topology and channel characteristics limit what APR is able to achieve.

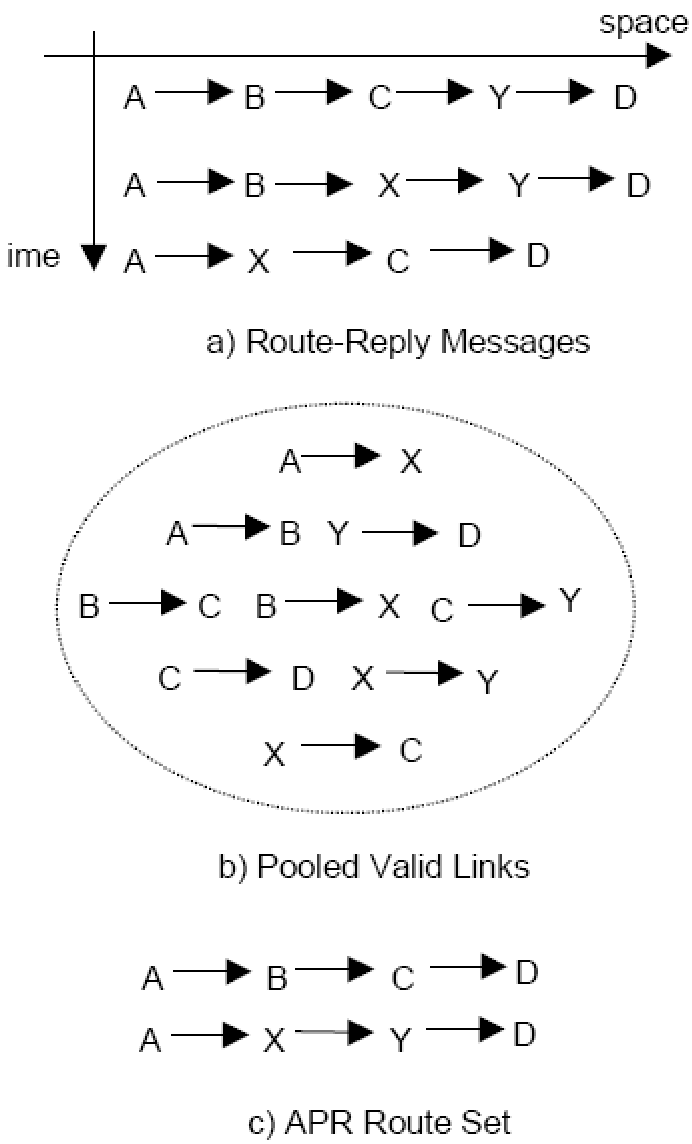



Figure 5: Decomposing Route Replies into Links States in order to Reconstruct Shorter, More Diverse APR Routes

## 3.2. AODV-BR

In [13], the author proposes a scheme to improve existing on-demand routing protocols by creating a mesh and providing multiple alternate routes. The algorithm establishes the mesh and multi-paths without transmitting any extra control message. And the scheme is applied to the Ad-hoc On-Demand Distance Vector (AODV) protocol.

An algorithm that utilizes a mesh structure to provide multiple alternate paths to existing on demand routing protocols without producing additional control messages is introduced. The mesh structure and alternate paths are established during the route reply phase. The AODV protocol in this procedure is slightly modified. Taking advantage of the broadcast nature of wireless communications, a node promiscuously overhears packets that are transmitted by their neighboring nodes. From these packets, a node obtains alternate path information and becomes part of the mesh as follows. When a node that is not part of the route overhears a RREP packet not directed to itself is transmitted by a neighbor (on the primary route), it records that neighbor as the next hop to the destination in its alternate route table.

For example, in Figure 6, the primary route <S-A-B-C-D> is selected. D sends a RREP to S. Y and Z overhears the packet and insert an entry to their alternative table. And so on, for W and X.

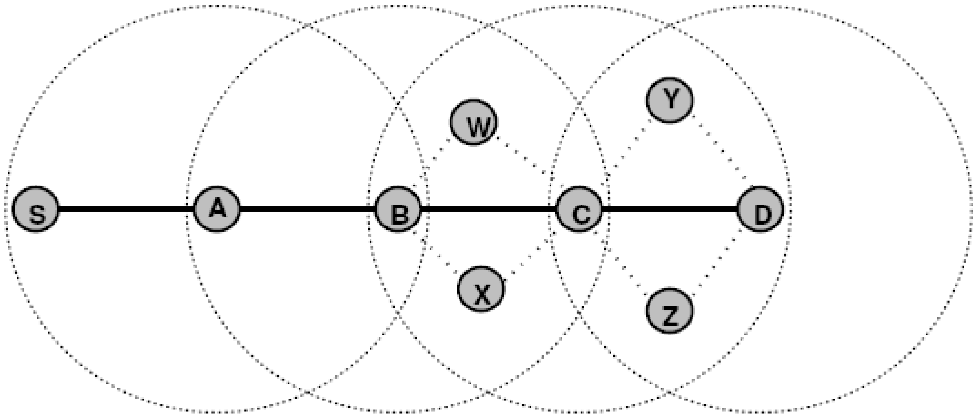



Figure 6: D is sending a Reply to S and the mesh is constructed

When the primary route is broken, for example, as shown in Figure 7, B moved out of the range of the next hop node C. After receiving the data packet from node A, B will find C is not reachable. Node B then broadcast the packet to its neighbors for alternative paths to salvage data. Nodes A and W will receive the packet, but node A drops it upon duplicate detection. Then W looks up in its alternate route table and finds C as the next hop to the destination.

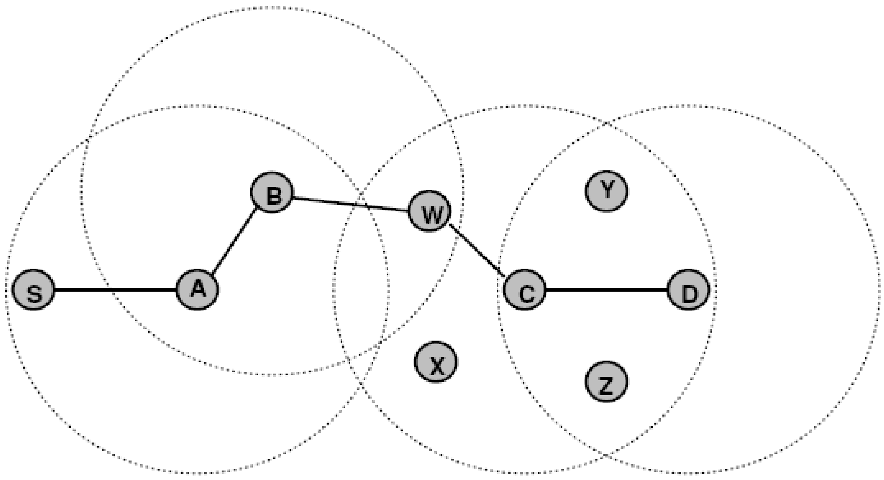



Figure 7: An example of route failure

The mesh configuration of AODV-BR provides multiple alternative routes and is constructed without yielding any extra overhead. It provides robustness to mobility and enhances protocol performance. But the protocol does not perform well under heavy traffic networks. And because it is based on AODV, it does not support uni-direction network either.

## 3.3. Braided Multi-path Routing

In [37], the author proposes a novel braided multipath scheme, which results in several partially disjoint multipath schemes. While disjoint paths have some attractive resilience properties, they can be energy inefficient. Alternate node-disjoint paths can be longer, and therefore expend significantly more energy than that expended on the primary path.

The braided multipath relaxes the requirement for node disjointedness. The definition for braided multipath is: For each node on the primary path, find the best path form source to sink that does not contain completely node-disjoint with the primary path. As shown in Figure 8.

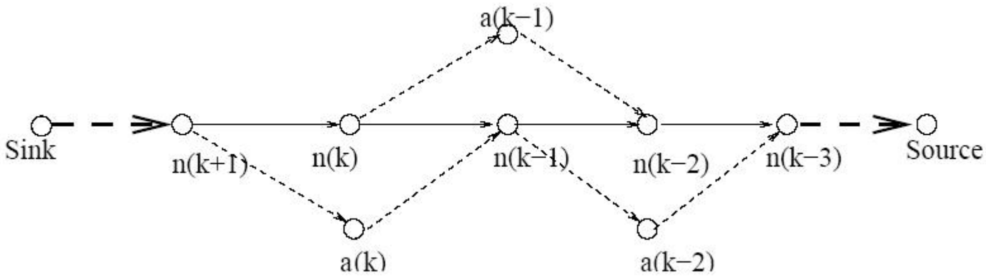



Figure 8: Braid Path

The paper mainly focuses on the use of multipath routing for energy-efficient recovery from nodes failures in wireless sensor networks. When a small number of multipaths are kept alive, failures on the primary path can usually be recovered from without invoking network-wide flooding for path discovery.

The protocols listed above build multiple routes on demand, but the traffic is not distributed into multi-paths: only one route is primarily used and alternate paths are utilized only when the primary route is broken.

In the rest part of this section, the parallel multi-path routing protocols are introduced, in which a set of paths is used at the same time to transmit the data.

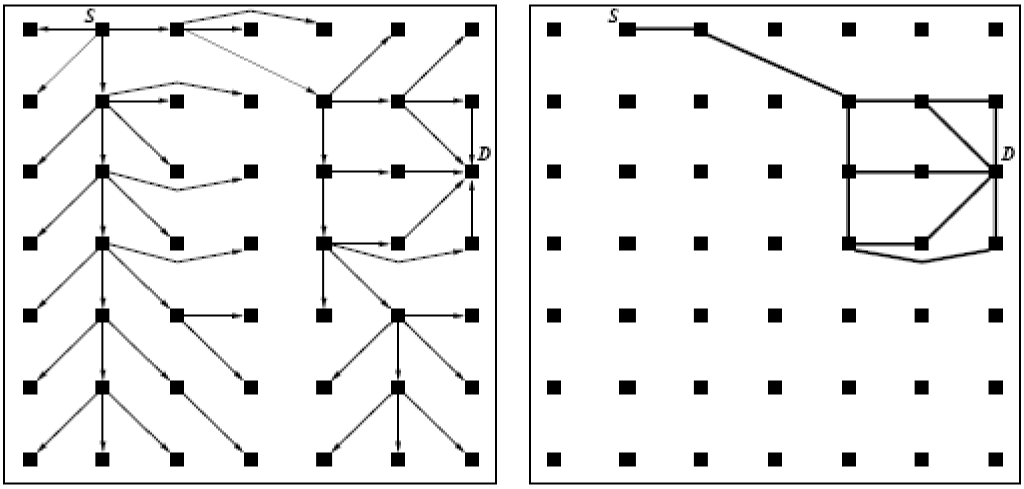
## 3.4. Split Multi-path Routing (SMR)

In [15], the author proposes an on-demand routing scheme called Split Multi-path Routing (SMR) that establishes and utilizes multiple routes of maximally disjoint paths. Providing multiple routes helps minimizing route recovery process and control message overhead. The protocol uses a per-packet allocation scheme to distribute data packets into multiple paths of active sessions. This traffic distribution efficiently utilizes available network resources and prevents nodes of the route from being congested in heavily loaded traffic situations.

***Route Discovery***

The main goal of SMR is to build maximally disjoint multiple paths. It is necessary to construct maximally disjoint routes to prevent certain nodes from being congested, and to utilize the available network resources efficiently. The destination must know the entire path of all available routes so that it can select the routes. The source routing approach is used.

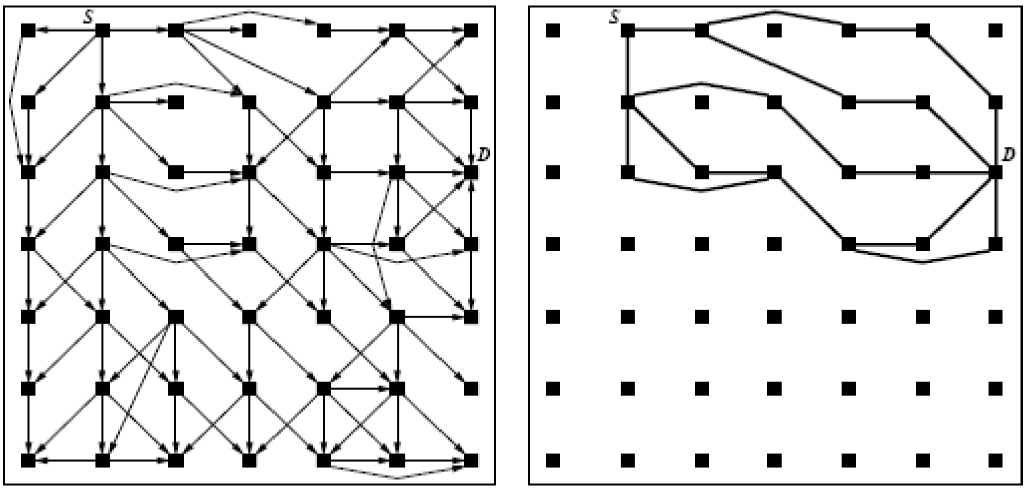
In the normal source routing approach, the duplicate RREQ packets are discarded. This may cause the multi-paths are mostly overlapped, as shown in Figure 9.




(a) RREQ propagation (b) Available paths

Figure 9: Overlapped multiple routes

In order to avoid this overlapped route problem, the author introduces a different packet forwarding approach, as shown in Figure 10. Instead of dropping every duplicate RREQs, intermediate nodes forward the duplicate packets that traversed through a different incoming link than the link from which the first RREQ is received, and whose hop count is not larger than that of the first received RREQ.




(a) RREQ propagation (b) Available paths

Figure 10: Multiple routes with maximally disjoint paths

For Route Selection, the destination selects two routes that are maximally disjoint. One of the two routes is the shortest delay route; the path taken by the first RREQ the destination receives. After this process, the destination waits certain duration of time to receive more RREQs and learn all possible routes. It then selects the route that is maximally disjoint to the route that is already replied. If there is more than one route that are maximally disjoint with the first route, the one with the shortest hop distance is chosen. If there still remain multiple routes that meet the condition, the path that delivered the RREQ to the destination the quickest between them is selected. The destination then sends another RREP to the source via the second route selected.

***Route Maintenance***

In SMR, when a node fails to deliver the packet to the next hop, a RERR message is send back to the source, and the source removes the correspond entry from its routing table. When the source is informed of a route disconnection and the session is still active, it may use one of the two policies in rediscovering routes:

- initiates the route recovery process when any route of the session is broken, or
- initiates the route recovery process only when both routes of the session are broken.

The first scheme reconstructs the routes more often and produces more control overhead than the second scheme, but the former provides multiple routes most of the time and be robust to route breaks.

The simulation indicates that SMR outperforms DSR because multiple routes provide robustness to mobility. The performance difference becomes evident as the mobility degree increases. SMR had considerably fewer packet drops compared with DSR. Splitting the traffic into multiple routes helps distribute the load to the network hosts. SMR also showed shorter end-to-end delay because route acquisition latency is not required for all route disconnections. Between the two Route Maintenance schemes of SMR protocol, the second scheme showed better efficiency as it performs fewer route recoveries and hence generates less control overhead.

## 3.5. MP-DSR

[17] defines a new QoS metric, end-to-end reliability in order to select a sub set of end-to-end paths to provide increased stability and reliability of routes. Multi-path dynamic source routing protocol (MP-DSR) for wireless ad-hoc networks tries to improve QoS support with respect to end-to-end reliability. The protocol forwards outgoing packets along multiple paths that are subject to a particular end-to-end reliability requirement.

***System Model***

The path reliability is calculated based on the link availabilities of all the links along a path. Link availability is defined as the probability that a link is available until time $t_0 + t$, given that it is an active link at time $t_0$. The calculation of link availability is based on the current node's movement. Path reliability is the product of link availabilities of all the links along the path under the assumption that all links are independent.

The end-to-end reliability is based on the link availability and path reliability model.

***Definition***: End-to-End reliability, P(t), is the probability of having a successful data transmission between two mobile nodes within the period of $t_0$ to $t_0 + t$, where $t_0$ is any time instant. It can be written as:

$$P(t) = 1 - \prod_{k \in K} (1 - \prod {}_{S,D}^{k}(t)$$

Where S is the source and D is the destination. K is a set of disjoint paths.

***Routing Protocol***

When an application uses MP-DSR for a route discovery, it supplies an end-to-end reliability requirement, $P_u$, where $0 \leqslant P_u \leqslant 1$. Given this requirement, MP-DSR determines two parameters for the route discovery: (1) the number of paths it needs to discover and (2) the lowest path reliability requirement that each search path must be able to provide in order to satisfy $P_u$.

They are referred to $m_0$ and $\prod_{lower}$ respectively.

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 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.

In MP-DSR, data transmission can then be soft provisioned with limited extra overhead. End-to-end reliability is also maintained throughout the whole transmission life time. Simulation results show that MP-DSR can offer higher and more consistent success delivery ratio than DSR. In addition, the lower error ratio of MP-DSR illustrates that its end-to-end transmission is more reliable. Finally, the control message overhead in MP-DSR is almost identical to that of DSR in average cases.

## 3.6. Graph-based Multipath Routing (GMR)

Graph-based Multipath Routing [34] is a novel multipath routing protocol that generates the network topology graph to compute all link disjoint paths in the networks. The destination node computes link disjoint paths using the local graph search algorithm.

GMR uses source routing mechanism just like DSR. So a RREQ message contains path information from source node to the intermediate node. But instead of single node list of DSR, each RREQ includes graph information which represents abstract network topology. We call this graph as Reverse Path Graph (RPG). The route discovery process of GMR performed with similar manner to that of DSR. The main difference is that each intermediate node which receives a new RREQ message should wait for some predetermined time to gather more RREQs, if any. If the waiting intermediate node receives more than one RREQ, it merges graph information of those duplicate RREQs with its previous graph information. After time-out, it re-broadcast one RREQ message which contains all information it gather until that moment.

Waiting some time from the first RREQ, the destination node gathers multiple RREQs. And after time-out, it computes link disjoint paths from the RPG using the local graph search algorithm and it replies to the source through multiple RREPs through multiple paths. Finally, the source uses source routing for data packet delivery. Figure 11 shows the graph information generated during a route discovery process. Node 4 receives upper edges from node 1 and lower edges from node 2 while it waits. And node 4 replays fully merged graph to node 6.

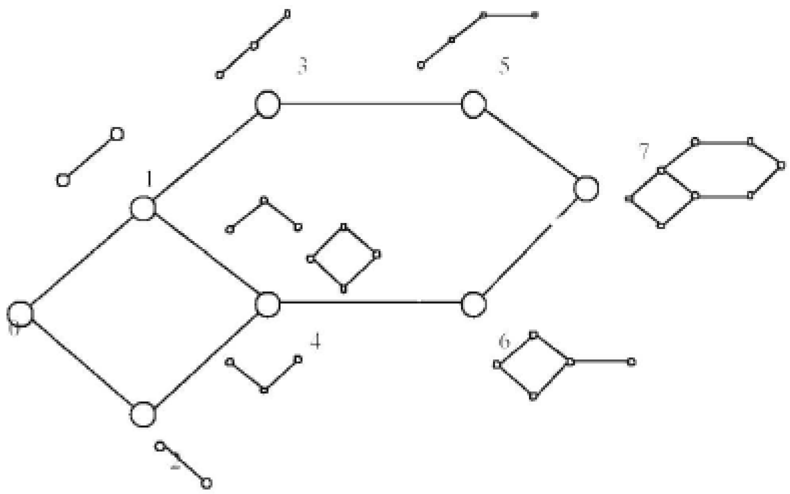



Figure 11: The basic operation of GMR

## 3.7. DMSR

Dynamic Multipath Source Routing [35] is proposed by extending the existing DSR to send packets simultaneously over multiple paths.

The author proposes a delay-bandwidth metric for evaluating the amount of extra bandwidth that may be used to achieve a certain reduction in delay and evaluates that 2 to 3 redundant paths will almost always give an optimal solution.

In DMSR, route discovery and maintenance are achieved in the same fashion as DSR. Since the protocol is still using source routing, only the source nodes need to have information about the multiple paths that can be followed. For m-path DMSR, copies of the packet are made at the source node, and sent along the m best (shortest) paths that the node has in its route cache. The subsequent nodes in the path do not need to do anything special for DMSR, and simply forward the packet to the next node according to the source route in the packet header.

In the simulation, at low network loads, DMSR is found to perform better than DSR as shown by the expected delay and delay-bandwidth metrics, although performance gain decreases with increase in the network load.

## 3.8. DPSP

In multi-path routing, the correlation between failures of the paths in a redundant path set should be as small as possible. But in the same time, selecting an optimal path set is an NP complete problem. So [19] proposes a Multipath routing algorithm, called Disjoint Pathset Selection Protocol (DPSP), to solve the problem.

The key insight behind DPSP is that a search algorithm can discover reliable path sets by temporarily considering overlapping paths. DPSP starts by locally constructing a partial view of the network based on the routes found by the underlying protocol. Each node in DPSP continuously monitors the reliability of each of its incident links and appends this information to the route reply packets returning to the source node. In order to acquire link reliability estimates with minimal overhead and exploit existing functionality, the authors use the measurement of the strength of the received signals (RSSI) from each neighbor.

The goal is: firstly, choose as many paths as possible and, secondly, at the same time include paths that are as reliable as possible. The solution is based on an iterative procedure of four steps:

1) A search for the most reliable s->t path.
2) A decision on whether this newly found path improves the path set reliability.

   Two metric is provided to decide when to augment a given path set with a new path. The motivation for the selection of the metric is given by the formulation of the edge-packing bound. The edge-packing of a graph G is defined as the problem of finding a (maximum) collection of edge disjoint sub-graphs of G. If $\lambda_i$ is the i-th out of S complete sets of edge-disjoint paths, and $p_j$ the reliability of the j-th edge, then, the best lower edge-packing bound is defined as:

$$L \triangleq \max_{1\le i\le S} L_i = \max_{1\le i\le S}\left\{\left[1-\prod_{P_k\in\lambda_i}(1-\prod_{j\in P_k} p_j)\right]\right\}$$

   The first metric, metric1, captures the reliability of the original path set and the candidate shortest path, and the second one, metric2, the reliability of the resulting path set after removing the interlacing. If metric1 is higher than metric2, then the interlacing is not removed. Or else:

3) A means of augmenting the path set.

4) A simple transformation of the underlying graph, so that the path search may temporarily use edges of paths already included in the set.

The direct benefits of DPSP include less frequent route discoveries, significantly lower routing overhead, lower transmission delays, and load balancing due the use of multiple paths. Further more, more accurate topology validation reduces the chance of using stale routing information. The incurred overhead, amortized per node, remains low due to the use of existing functionality (reliability measure estimation, no additional route replies) and the efficiency of the algorithm.

## 3.9. CHAMP

The Caching and Multipath Routing (CHAMP) [20] employs cooperative packet caching to improve fault tolerance and uses round-robin traffic allocation to keep routes fresh.

***Data Structures***

Every node maintains two data structures: a route cache to contain forwarding information; and route request cache for storing recently received and processed route requests. The route cache at node I is a list containing an entry for each active destination j. A route entry which has not been used for more than *RouteLifeTime* seconds is deleted. The route request cache at node I is a list containing an entry for every unique route request received and processed. In addition to the two data structures, every node also maintains two FIFO buffers: a send buffer for storing packets waiting for routes; and a data cache for storing recently forwarded data packets.

***Route Discovery and Maintenance***

CHAMP operates on-demand. 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.

The simulation shows that a small data cache is sufficient to improve packet delivery. In essence, cooperative packet caching enables distributed packet salvaging. The discovery mechanism selects non-disjoint shortest multipath routes of equal distance. It has been shown that having at most two routes for every destination is already the optimum. Having more routes does not give any additional benefit.

## 3.10. Others

The Multipath Source Routing (MSR) proposed in [23] uses the delay of the routes to calculate the number of the packets transmitted through each route at a time to balance distribute the load.

Manet Queue Length Based Multi Source Routing (MQLBMSR) [24] is an algorithm based on the queue length to make full use of the resources and reduce end-to-end delay. The protocol uses the length of the queue to determine the congestion of a node, and then adjust the traffic of different path to reduce the packet delay.

Hybrid Multipath Routing (HMPR) [25] does not require that the multiple routes to the destination be disjoint. HMPR uses a simple algorithm to estimate the wireless link statistics, which can aid route selection. A joint route selection criterion is then used to select one effective route to the destination.

The primary disadvantages of multipath routing compared to unipath protocols are complexity and overhead. The simulation in [27] showed that at low bit rate, the multipath has more end-to-end delay than unipath, because some paths selected might not be in the best situation.

# 4. Issues to be discussed

## 4.1. Paths Discovery and Selection

Paths discovery and selection have always been the most important issues in the multipath routing. Multipath routing protocols can attempt to find node disjoint, link disjoint or non-disjoint routes.

For a set of paths to achieve high reliability in aggregate, the correlation of failures between the paths in the set should be as low as possible. Shared links and nodes between paths present common failure points can disable many or all of the paths in the set. A lot of algorithms in Section 3 try to find disjoint paths in different ways, either link disjoint or node disjoint. However, a paths discovery method is still needed to find as many disjoint paths with high reliability as possible and in the same time, not cost too much.

In order to evaluate the routes, some methods are introduced, such as end-to-end reliability [17]. It is necessary to use some kind of metrics to estimate the path quality to make the right decision.

After the discovery, we get a set of routes and have to decide how many routes and which of them to be used to maximize the performance. SMR [15] uses two routes for the transform, and the simulation result from [20] shows that at most two routes for every destination is already the optimum. This problem is still to be concern in the future.

## 4.2. Failure Recovery

One of the benefits of multipath is that when a node fails to deliver a packet, the protocol could performance packet salvaging for fault tolerance. With packet salvaging [20][21] , intermediate nodes maintain multiple routes to the destination, and a RERR message propagates only until an intermediate node can not forward the packet along an alternate route. In CHAMP, cache and locality are utilized to performance failure recovery. Because of the mobility of Ad hoc networks, we need an efficient failure recovery mechanism to decrease the route discoveries.

## 4.3. Load Balancing

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 load balancing and traffic allocation strategy used deals with how the data is distributed among the paths.

The first problem is allocation granularity. It specifies the smallest unit of information allocated to each path. It is shown that a per-packet granularity results in the best performance. This is because it allows for finer control over the network resources.

Another problem is how the packets to be distributed to each path. If a round-robin traffic allocation 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. [24] proposed a scheme to allocate the packet according to the queue length. However, this problem still needs to be discussed in the future work.

## 4.4. Hidden Nodes

Hidden nodes [30] in a wireless network refer to nodes which are out of range of other nodes or a collection for nodes. Take a physical star topology with an Access Point with many nodes surrounding it in a circular fashion; each node is within communication range of the Access Point, however, not each node can communicate with each other. For example, it is likely that the node at the far edge of the circle can see the access point, which is known as r, but it is unlikely that the same node can see a node on the opposite end of the circle, 2r (or simply the diameter). These nodes are known as hidden. The problem is when node r and r2 start to send packets simultaneously to the access point. Since node r and r2 can not sense the carrier, Carrier sense multiple access with collision avoidance (CSMA/CA) does not work. To overcome this problem, handshaking is implemented in conjunction with the CSMA/CA scheme. The same problem exists in a MANET.

IEEE 802.11 uses 802.11 RTS/CTS acknowledgement and handshake packets to partly overcome the hidden node problem. RTS/CTS is not a complete solution and may decrease throughput even further (see benchmarks in external links), but adaptive acknowledgements from the base station can help too.

The other methods that can be employed to solve hidden node problem are:

1. Increase power to the nodes,

2. Use omnidirectional antennas,
3. Remove obstacles,
4. Move the node,
5. Use protocol enhancement software

## 4.5. TCP Friendly

The multipath routing protocols presented above in ad hoc are mostly in terms of the network layer. Sometimes, we should pay attention to 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.

CHAMP uses equal paths to reduce out-of-order problem, and some diversity coding techniques are introduced. Another way is to use UDP rather than TCP. However, in some circumstances, the order of the packets' arriving at the destination still has to be guaranteed.

## 4.6. QoS

In the 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.

# 5. Some Considerations

This section includes some of my considerations during reading all these papers which might be considered in the next step. Some of them are only of the first thought, and immature at the moment. These still have to be verified in the future.

## 5.1. Disjoint or Non-Disjoint?

In general, 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. What's more, node disjoint routes offer the highest degree of fault-tolerance.

The main disadvantage of non-disjoint routes is that they can not be easily discovered. 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 the nodes increases [33]. 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. However, because link disjoint routes might share some nodes, it might cause the "hidden nodes" problem, with have to be considered.

In the future work, we may try to find node-disjoint routes in the first place. And then, we take link-disjoint routes into consideration.

## 5.2. Multipath or Unipath?

It is generally believed that multipath has better performance than unipath. However, because of the complexity of the multipath protocols, under certain circumstances, unipath protocols outperform the multipath [27]. What's more, when using multipath, how many of them should be used is also a problem.

So it might be necessary to use a self-adaptive method to determine the path selection according to the real-time situation of the nodes and paths.

## 5.3. Agent?

The nodes in the Ad hoc networks are very similar to the concept of "Agent" in distributed AI, which are distributed, connected, cooperative, exoteric, and independent. So the Ad hoc network might be considered as a multi-agent System. We may get some ideas from the distributed AI.

## 5.4. Locality, cache, broken paths?

When a path is broken, it can be safely assumed that it is because one or several nodes are down because of moving out of the range or low power, rather than all the nodes in the path. So even a route is down, some of its information is still useful. Make use of these properties might enhance the route discovery.

And, when the mobile node moves, "it cannot move too far too soon". So there is spatial and temporal locality for the ad hoc network. So we might be able to utilize a cache mechanism to have better performance.

## 5.5. Bi-direction or Uni-direction? Heterogeneous Nodes?

In a lot of protocols, it is assumed that the paths between each node are bi-directional. However, this might be hard to be achieved in real. Each node in Ad hoc network has the same function. It can act as a router or a terminal, but the performance is not always the same. In a battlefield, low powered sensors or handhelds may be deployed in the field, with high powered, reliable and secure nodes located in large vehicles. Even in commercial environment, there is no wonder that a laptop is more powerful than a PDA or cell phone. How to make good use of these heterogeneous nodes in a uni-direction path is a issued to be discussed.

## 5.6. Independent movement or Dependent movement?

The movement of the nodes might be completely independent of each other. However, there are many situations in which it is necessary to model behavior of mobile nodes as they move together. For example, a group of soldiers in a military scenario may be assigned the task of searching a particular plot of land in order to destroy land mines, capture enemy attackers or simply work together in a cooperative manner to accomplish a common goal. So we many take the dependence into consideration to get better performance.

# 6. NS2 and Simulations

## 6.1. Network Simulator 2

NS is a discrete event simulator targeted at networking research. NS provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks.

Ns began as a variant of the REAL network simulator in 1989 and has evolved substantially over the past few years. In 1995 ns development was supported by DARPA through the VINT project at LBL, Xerox PARC, UCB, and USC/ISI. Currently ns development is support through DARPA with SAMAN and through NSF with CONSER, both in collaboration with other researchers including ACIRI. NS has always included substantial contributions from other researchers, including wireless code from the UCB Daedelus and CMU Monarch projects and Sun Microsystems. In my master subject, the NS2 will be used to simulate the wireless ad hoc.

In my master subject, NS2 will be used to simulate the ad hoc networks.

## 6.2. Mobility Models

In the performance evaluation of a protocol for an ad hoc network, the protocol should be tested under realistic conditions. So certain mobility model should be applied. The mobility models can be divided into two groups: entity mobility models, in which the movements of the nodes are independent, and, group mobility models, in which the movements are dependent on each other [32].

### 6.2.1. Random Walk

The Random Walk Mobility Model was first described mathematically by Einstein in 1926. In the mobility model, an Mobile Node (MN) moves form its current location to a new location by randomly choosing a direction and speed in which to travel. The new speed and direction are both chosen form predefined ranges respectively. Each movement in the Random Walk Mobility Model occurs in either a constant time interval t or a constant distance traveled d, at the end of which a new direction and speed are calculated.

Because the current speed and direction of an MN is independent of its past speed and direction, the algorithm may generate unrealistic movements such as sudden stops and sharp turns. However, Random Walk is still widely used by the researchers.

### 6.2.2. Random Waypoint

The Random Waypoint Mobility Model includes pause times between changes in direction or speed. An MN begins by staying in one location for a certain period of time. Once this time expires, the MN chooses a random destination in the simulation area and a speed that is uniformly distributed between the min speed and max speed. The MN then travels toward the newly chosen destination at the selected speed. Upon arrival, the MN pauses for a specified time period before starting the process again.

One of the problems of the Random Waypoint is that its initial random distribution of MNs is not representative of the manner in which nodes distribute themselves when moving. So we may use a save of the locations of the MNs after a simulation has executed long enough to be past this initial high variability, or discard the initial time of the procedure.

### 6.2.3. Reference Point Group Mobility Model

The RPGM model represents the random motion of a group of MNs as well as the random motion of each individual MN within the group. Group movements are based on the path traveled by a logical center for the group. The logical center for the group is used to calculate group motion via a group motion vector, $\overrightarrow{GM}$. The motion of the group center completely characterized the movement of its corresponding group of MNs, including their direction and speed. Individual MNs randomly move about their own predefined reference points, whose movements depend on the group movement. As the individual reference points move from time t to t+1, their locations are updated according to the group’s logical center. Once the updated reference points, RP(t+1), are calculated, they are combined with a random motion vector, $\overrightarrow{RM}$, to represent the random motion of each MN about its individual reference point.

Column Mobility Model, Nomadic Community Mobility Model and the Pursue Mobility Model can be obtained from RPGM model.

The simulation [32] has shown that, firstly, the performance of an ad hoc network protocol can vary significantly with different mobility models or the same mobility model with different parameters. Secondly, the selection of a mobility model may require a data traffic pattern that significantly influences protocol performance. Thirdly, the performance of an protocol should be evaluated with the mobility model that most closely matches the expected real world scenario.

## 6.3. Performance Criteria

The performance of routing protocol will be compared under different mobility. So we need several metrics to evaluate the performance. The following metrics might be used:

- ***Control overhead***: The control overhead is defined as the total number of routing control packets normalized by the total number of received data packets.
- ***Packet delivery ratio***: the ratio of the data packets delivered to the destination.
- ***Routing load***: gives the number of routing packets over the number of received data packets. Each routing packet sent or forwarded by a mobile is counted.
- ***Bandwidth cost for data***: The bandwidth cost for data is defined as the total number of data packets transmitted at all mobile hosts normalized by the total number of received data packets.
- ***Average end-to-end delay***: The end-to-end delay is averaged over all surviving data packets form the sources to the destinations. It includes queuing delay and propagation delay.
- ***Jitter***: gives the transmission delay variation.
- ***Activity distribution (A)***: is calculated as follow: $\sum_{i=0}^{N-1}\left|\frac{1}{N}-P_i\right|$ where $P_i=\frac{n_i}{\sum_j n_j}$ is the probability that the mobile node I make a sending or forwarding action. A=0 means that all mobile nodes fairly work.
- ***Load balancing***: We use a graph G=(V,E) to denote the network, where V is the node set and E is the link set. We define a state function of f: V→I where I is the set of positive integers. f(v) represents the number of data packets forwarded at node v. Let CoV(f)=standard variance of f / mean of f. We use CoV(f) as a metric to evaluate the load balancing. The smaller the CoV(f), the better the load balancing.
- ***Energy balancing***: As above, we use CoV(g) to evaluate the energy balancing, where g(v) represents the energy consumption at each node. The energy consumption in a node is calculated as the sum of Ti*Pi (i=0,1,2), where Ti represents the time spent in the three different modes (doze, receive, and transmit mode) and Pi represents the power consumption in the corresponding modes.
- ***Average energy consumption***: The energy consumption is averaged over all nodes in the network.

# Reference

[1] David B. Johnson, David A. Maltz, Josh Broch, “DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks,” Ad Hoc Networking, Addison-Wesley, 2001:139～172.

[2] David B. Johnson, David A. Maltz, “Dynamic Source Routing in Ad hoc Wireless Networks”, Mobile Computing. (1996) 153-181.

[3] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, L.Viennot, “Optimized Link State Routing Protocol for Ad Hoc Networks,” Proceedings of the 5th IEEE Multi Topic Conference (INMIC 2001)

[4] Andreas Tønnesen, “Implementing and extending the Optimized Link State Routing protocol,” Master thesis, 2004.

[5] Charles E. Perkins and Elizabeth M. Royer. "Ad hoc On-Demand Distance Vector Routing." Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, New Orleans, LA, February 1999, pp. 90-100

[6] Stephen Mueller, Rose P. Tsang, Dipack Ghosal. “Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges,” In Perfor-. mance Tools and Applications to Networked Systems, vol-. ume 2965 of LNCS. Springer-Verlag, 2004

[7] Nicklas Beijar, “Zone Routing Protocol,” http://www.tct.hut.fi/opetus/s38030/k02/Papers/08-Nicklas.pdf

[8] Frederick Ducatelle, Gianni Di Caro, Luca Maria Gambardella. “Ant Agents for Hybrid Multipath Routing in Mobile Ad Hoc Networks,” Wireless On-demand Network Systems and Services, 2005. WONS 2005. Second Annual Conference ,19-21 Jan. 2005

[9] Dynamic Source Routing, http://wiki.uni.lu/secan-lab/Dynamic+Source+Routing.html

[10] Ad Hoc On-Demand Distance Vector Routing Protocol, http://wiki.uni.lu/secan-lab/Ad+Hoc+On-Demand+Distance+Vector+Routing+Protocol.html

[11] Salima Hamma, Eddy Cizeron, Hafiz Issaka, Jean-Pierre Guedon, “Performance evaluation of reactive and proactive routing protocol in IEEE 802.11 ad hoc network, Next Generation Communication and Sensor Network 2006, Boston 2006.

[12] Marc R Pearlman, Zygmunt J Haas, Peter Sholander, et al. Alernate Path Routing in Mobile Ad Hoc Networks, Proc of IEEE INFOCOM 2002.

[13] S J Lee, M Gerla. AODV-BR: Backup Routing in Ad hoc Network. IEEE WCNC 2000. 2000. 1311-1316.

[14] Tracy Camp, Jeff Boleng, Vanesssa Davies, “Redundancy-Based Multi-path Routing in Ad Hoc Netorks”, IEEE, ICCN’99, 1999.

[15] S J Lee, M Gerla. Split Multi-Path Routing with Maximally Disjoint Paths in Ad Hoc Networks. ICC’ 01, 2001.

[16] Marina M K, Das S R, “On-demand Multipath Distance Vector Routing in Ad Hoc Networks,”, IEEE International Conference for Network Protocols(2001).

[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] A Tsirigos, Z J Hass. Multi-Path Routing in the Presence of Frequent Topological Changes. IEEE Communications Magazine, 2001, 39(11): 132-138.

[19] P Papadimitratos, Z J Haas, E G Sier, Path Set Selection in Mobile Ad Hoc Networks. ACM Mobihoc'02, 2002.

[20] Valera A, Seah, W K G, Rao, S V: Cooperative Packet Caching and Shortest Multipath Routing in Mobile Ad hoc Networks. IEEE INFOCOM(2003).

[21] Nasipuri A, Das S R, On-Demand Multipath Routing for Mobile Ad Hoc Networks. Proceeding of the 8th International Conference on Computer Communications and Networks (1999).

[22] A Valera, K G Seah, S V Rao, Champ: A highly resilient and enery-effcient routing protocol for mobile ad hoc networks, in Proc. IEEE MWCN'02, 2002, pp.43-47

[23] Y Shu, D Gao, L Wang, Multipath Source Routing in Wireless Ad Hoc Networks, ACTA Electronica Sinica, Vol. 30 No.2, Feb 2002. (舒炎泰，高德云，王雷，无线网络中的多径路由．电子学报, Vol. 30 No. 2, 2002 年 2 月)

[24] J Miao, D Sun, W Ding, Research on Multi-Path Source Routing Algorithm Based on Queue Length in Mobile Ad Hoc Networks, Journal of Jilin University (Information Science Edition), Vol. 24, No.4, July 2006. (苗建松，孙丹丹，丁炜，移动 Ad Hoc 网络中队列长度的多径源路由算法，吉林大学学报，信息科学版，第 24 卷 第 4 期 2006 年 7 月)

[25] Z Yao, Z Cao, P Fan, Hybrid Multipath routing in mobile ad hoc networks, Journal of Tsinghua University (Sci&Tech), 2004, Vol. 44, No. 7. (姚忠邦，曹志刚，樊平毅，移动 Ad hoc 网络中的混合式多路径路由算法，清华大学学报，自然科学版，2004 年第 44 卷第 7 期)

[26] H An, X Lu, A Review of Routing Protocols for Mobile Ad Hoc Networks, Computer engineering & Science, Vol.28, No. 2, 2006. (安辉耀，卢锡城，移动自主网络多路径路由技术研究进展，计算机工程与科学, 2006 年第 28 卷第 2 期)

[27] H Tan, F Xie, L Yang, P Zhang, Channel Adaptive Multi-path Routing for Ad Hoc Networks, Journal of Beijing University of Posts and Telecommunications, Vol. 28 No.1. (田辉，谢芳，杨宁，张平，Ad Hoc 网络中信道自适应多径路由算法，北京邮电大学学报，第 28 卷第 1 期, 2005 年 2 月)

[28] D Ganesan, R Govindan, S Shenker, D Estrin, Highly-Resilient, Energy-Efficient Multipath Routing in Wireless Sensor Networks, ACM SIGMOBILE Mobile Computing and Communications Review, Volume 5 , Issue 4 (October 2001).

[29] C Intanagonwiwat, R Govindan, D Estrin, Directed diffusion: A scalable and robust communication paradigm fo sensor networks. In Proc. ACM Mobicom, Boston, MA, 2000.

[30] Wikipedia, Hidden node problem, http://en.wikipedia.org/wiki/Hidden_node_problem

[31] Network Simulator 2, http://www.isi.edu/nsnam/ns/index.html.

[32] T Camp, J Boleng, V Davies, A survey of mobility models for ad hoc network research, Wireless communications and mobile computing, 2002, 2:483-502.

[33] Ye Z, Krishnamurthy S V, Tripathi S K, A Framework for Reliable Routing in Mobile Ad Hoc Networks, IEEE INFOCOM(2003).

[34] Gunyoung Koh, Duyoung Oh, Heekyoung Woo, A Graph-based Approach to Compute Multiple Paths in Mobile Ad Hoc Networks, Lecture notes in computer science (Lect.

notes comput. sci.) ISSN 0302-9743.

[35] Amit K Vyas, Amit K Gupta, A New Multipath Routing Protocol for Ad hoc Wireless Networks. http://www.stanford.edu/class/ee384y/projects/reports/

[36] C Intanagonwiwat, R Govindan, D Estrin, Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. MOBICOM 2000: 56-67

[37] D Ganesan, R Govindan, S Shenker, D Estrin, Highly-Resilient, Energy-Efficient Multipath Routing in Wireless Sensor Networks. MobiHoc 2001: 251-254