COMBINATORIAL OPTIMIZATION IN COMMUNICATION NETWORKS
Maggie Cheng, Yingshu Li and Ding-Zhu Du (Eds.) pp. – - –

Performance Optimization using Multipath Routing in Mobile Ad Hoc and Wireless Sensor Networks

Wenjing Lou
Department of Electrical and Computer Engineering
Worcester Polytechnic Institute, Worcester, MA 01609
E-mail: wjlou@ece.wpi.edu

Wei Liu
Department of Electrical and Computer Engineering
University of Florida, Gainesville, FL 32611
E-mail: liuw@ufl.edu

Yanchao Zhang
Department of Electrical and Computer Engineering
University of Florida, Gainesville, FL 32611
E-mail: yczhang@ufl.edu

Contents

1 Introduction **2**

2 Performance Benefits from Multipath Routing **4**
2.1 Reliability 5
2.2 Load/Energy Consumption Balancing 7
2.3 Routing Overhead 7
2.4 Quality of Service (QoS) 8
2.5 Security 10

3 Multipath Routing Protocols **12**
3.1 Partially Disjoint Paths 12
3.2 Disjoint Paths 14
3.2.1 Edge-disjoint Paths 15
3.2.2 Node-disjoint Paths 16

4 The SPREAD Scheme **18**
4.1 SPREAD Overview 18
4.2 End-to-end Multipath Routing 19
4.3 N-to-1 Multipath Routing 21
4.4 SPREAD Summary 23

5 Conclusion **24**

References

1 Introduction

Multipath routing, sometimes called *traffic dispersion* [14], has been one of the most important current directions in the area of routing. The current routing is based on the single path routing - between a source and a destination, the single minimum-cost path tends to be selected although different cost metrics may yield different paths. However, in a reasonably well-connected network, there may exist several paths between a source-destination pair. The concept of multipath routing is to give the source node a choice at any given time of multiple paths to a particular destination by taking advantage of the connectivity redundancy of the underlying network. The multiple paths may be used alternately, namely, traffic taking one path at a time, or they may be used concurrently, namely, traffic flowing through multiple paths simultaneously.

Multipath routing (or *dispersity routing* as termed by the author) was first proposed by Maxemchuk to spread the traffic from a source in space rather than in time as a means for load balancing and fault handling in packet switching networks [29–31]. The method was shown to equalize load and increase overall network utilization; with redundancy, it improves the delay and packet loss properties at the expense of sending more data through the network. Since then, the multipath routing technique has been applied to various types of networks, such as the communication networks, B-ISDN, ATM networks, etc., and for various network control and management purposes, such as aggregating the bandwidth, minimizing the delay, supporting the Quality of Service (QoS) routing, smoothing the burstiness of the traffic, alleviating the network congestion, and improving the fault tolerance, etc [3,6,9,39,40]. Interested readers are referred to [14] for a comprehensive survey of the earlier works on traffic dispersion in wired networks.

Mobile ad hoc networks (MANETs) and wireless sensor networks (WSNs) have received tremendous attention in the past few years. A MANET is a collection of nodes that can move freely and communicate with each other using the

wireless devices. For the nodes that are not within the direct communication range, other nodes in the network work collectively to relay packets for them. A MANET is characterized by its dynamic topological changes, limited communication bandwidth, and limited battery power of nodes. The network topology of a MANET can change frequently and dramatically. One reason is that nodes in a MANET are capable of moving collectively or randomly. When one node moves out of/in to the transmission range of another node, the link between the two becomes down/up. Another reason that causes the topological changes is the unstable wireless links, which might become up and down due to the signal fading (obstacles between the two end nodes), interference from other signals, or the changing of transmission power levels. Most of the mobile nodes are battery powered, when the nodes run out of the battery power, the node failure will also cause the topological changes. Although a close relative to MANETs, a WSN differs from an ad hoc network in many aspects [2]. The number of nodes in a WSN is usually much larger than that in an ad hoc network. Sensor nodes are more resource constrained in terms of power, computational capabilities, and memory. Sensor nodes are typically randomly and densely deployed (e.g., by aerial scattering) within the target sensing area. The post-deployment topology is not predetermined. Although in many cases the nodes are static, the topology might change frequently because the sensor nodes and the wireless channels are prone to failure.

Multipath routing has drawn extensive attention in MANETs and WSNs recently. The dense deployment of nodes in MANETs/WSNs makes the multipath routing a nature and promising technique to cope with the frequent topological changes and consequently unreliable communication services. Research efforts have also been made using multipath routing to improve the robustness of data delivery [41, 46], to balance the traffic load and balance the power consumption among nodes [13, 45], to reduce the end-to-end delay and the frequency of route discoveries [11,33], and to improve the network security [24,48], etc. Two primary technical focuses in this area are, (a) the multipath routing protocols that are able to find multiple paths with the desired properties, and (b) the policies on the usage of the multiple paths and the traffic distribution among the multiple paths, which very often involve coding schemes that help to split the traffic.

Communication security and reliability are two important issues in any network and they are seemingly contradict goals from the perspective of adding redundancies. On the one hand, traditionally reliability can be achieved by sending redundant data over multiple paths. On the other hand, the redundant data gives the adversaries better chances to intercept the information. To address this issue, we proposed a novel *Security Protocol for REliable dAta Delivery* (SPREAD) to enhance both security and reliability and we investigated the SPREAD scheme in both MANETs and WSNs. The goal of the proposed SPREAD scheme is to

provide further protection to the data delivery service, specifically, to reduce the probability that secret messages might be compromised/lost when they are delivered across the insecure/unreliable network. The basic idea is to transform a secret message into multiple shares by secret sharing schemes and then deliver the shares via multiple independent paths to the destination so that even if a small number of nodes that are used to relay the message shares are compromised/faulty, the secret message as a whole is not compromised/lost. In the SPREAD scheme, multipath routing is used to distribute the trust to multiple paths/nodes for security purpose and to diminish the effect of unreliable nodes/links for the purpose of reliability. An end-to-end multipath routing technique to find multiple node-disjoint paths between a source-destination pair for end-to-end data delivery was investigated in a MANET environment [24]. While in a WSN, noticing that a typical communication task is for every sensor node to sense its local environment and, upon request, sends the data of interest back to a base station, we proposed an efficient N-to-1 multipath discovery protocol that is able to find multiple node-disjoint paths from every sensor node to the base station simultaneously in one route discovery [26]. The SPREAD scheme combines both concurrent multipath routing and alternate multipath savaging techniques and provides more secure and more reliable data collection service. In other words, it makes the data delivery service more resistent to node/link failure and node compromise problems.

This chapter aims to introduce the readers with the concept and techniques of multipath routing in a MANET/WSN, with a focus on the performance gains from multipath routing and the techniques of the construction and the usage of the paths. The chapter is organized as follows. We identify some applications and examine their performance benefits from multipath routing in section 2. We review some multipath routing protocols in the literature in section 3. Then in section 4 we present the SPREAD scheme with moderate details. Section 5 concludes the chapter.

2 Performance Benefits from Multipath Routing

Multipath routing has been studied for various network control and management purposes in various types of networks. In this section, we outline some of the applications of multipath routing that improve the performance of an ad hoc network and a sensor network.

2.1 Reliability

By "reliability" we mean the probability that a message generated at one place in the network can actually be routed to the intended destination. Reliability is a big challenge in MANETs/WSNs because packets transmitted are subject to lost due to frequent topological changes, severe media access conflicts, and various kinds of interferences that affect the wireless transceivers to correctly decode the wireless signals.

Multipath routing in a MANET was originally developed as a means to provide route failure protection. For example, the Dynamic Source Routing (DSR) protocol [17] is capable of caching multiple routes to a certain destination. When the primary path fails, an alternate one will be used to salvage the packet. The Temporally Ordered Routing Algorithm (TORA) [34] also provides multiple paths by maintaining a destination-oriented directed acyclic graph (DAG) from the source node. Multipath extensions of some protocols that originally depend on the single path routing have also be proposed, such as the AODV-BR [27], Alternative Path Routing (APR) [36], and Split Multipath Routing (SMR) [28], etc., which improve the single path routing protocols by providing multiple alternate routes. In these cases, the multiple paths are not used simultaneously. The traffic takes one of the multiple paths at a time. Other paths are kept as backup in case the used one is broken. When all known paths are broken, a new multipath discovery procedure is initiated. Alternate path routing has also been adopted at link layer - when multiple next hops are available, the packet is routed through the one that exhibits best channel condition [19].

Another way of using the multiple paths is to have the traffic flow through multiple paths simultaneously. Concurrent multipath routing in a MANET has been developed to improve the throughput, reliability and achieve load balancing. Some type of source coding scheme is usually incorporated, particularly for reliability, such as the *Forward Error Correction* (FEC) codes, *Reed-Solomon* (RS) codes, etc. As shown in Fig. 1, a certain amount of redundancy is coded into the data traffic such that the decoding could tolerate a certain amount of data lost or even the failure of a complete path (paths). One application of concurrent multipath routing was proposed in [41] in which overhead information is added to the original data load by *diversity coding* [1], then segments consisting of both types of data are distributed over multiple independent paths simultaneously. The diversity coding has the property that upon receiving a certain amount of data, either original or overhead, the original information can be fully recovered. With an analytical framework, the scheme was shown to be effective in increasing the overall information delivery ratio. Another interesting application [15] was proposed which combined the multiple path transport (MPT) and *multiple description coding* (MDC) in order

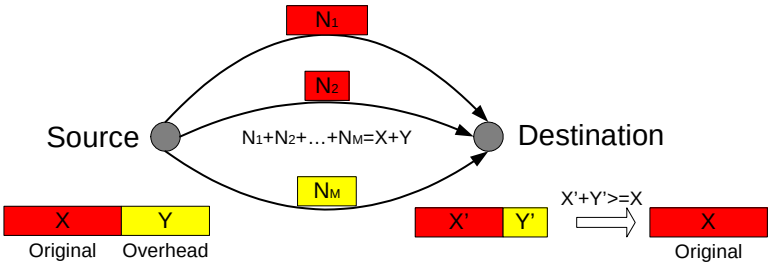


Figure 1: Concurrent Multipath Routing for Reliability

to send video and image information in a MANET. The MDC has the property that the picture quality is acceptable from any one description while any additional description enhances the picture quality accumulatively. The transmission of the multiple descriptions via multiple paths, on the one hand, helps to provide higher bandwidth required by the video/image transmission, on the other hand, helps to provide more robust end-to-end connection.

In [24] and [26], we combined the *threshold secret sharing* scheme and multipath routing and proposed the SPREAD idea to provide more reliable and more secure data delivery/collection services in a MANET/WSN. The secret sharing scheme has the similar property as the diversity coding from the reliability perspective - a (T, N) threshold secret sharing algorithm divides a piece of information into N segments, from any T out of N segments, one can reconstruct the original information. In addition, the secret sharing scheme has a desirable security property, that is, with less than the threshold, namely T, segments, one could learn nothing about the information and has no better chance to recover the original information than an outsider who knows nothing at all about it. The SPREAD idea was investigated in [24] for end-to-end data delivery in a MANET. It is noticed that the improved end-to-end data delivery ratio relies on the excessive information redundancy allocated to the multiple paths. The reliability of the overall end-to-end delivery is improved because it could tolerate a certain amount of packet loss or path loss, while the reliability of each path remains unimproved. This situation makes reliability and security contradict goals - high redundancy improves reliability but deteriorates the foundations of the security enhancement. The SPREAD scheme was extended in [26] by proposing a N-to-1 multipath discovery protocol. The distinct feature of the N-to-1 multipath discovery protocol is that it is able to find from every node to a particular destination (the sink node in a WSN) multiple node-disjoint paths in one route discovery efficiently. With the multipath available at every node, the SPREAD scheme achieves both reliability and security goals with little or none information redundancies. The end-to-end concurrent

multipath routing is applied at the source to spread the traffic onto multiple disjoint paths between the source and the destination for security purpose. The alternate path routing is applied while each packet is travelling on its designated path - once node/link failure is encountered, the packet is locally salvaged by using an alternate path. By this means, the reliability of each path is greatly improved thus the required end-to-end redundancy is greatly decreased. The dilemma between security and reliability are neatly resolved by the combination of concurrent multipath routing and alternate multipath routing. More details of the SPREAD scheme will be presented in section 4.

2.2 Load/Energy Consumption Balancing

Nodes in a MANET or WSN are typically powered by batteries which have limited energy reservoir. In some application scenarios, replenishment of power supplies might not be possible. The lifetime of the nodes show strong dependence on the lifetime of the batteries. In the multihop MANET/WSN, nodes depend on each other to relay packets. The lost of some nodes may cause significant topological changes, undermine the network operation, and affect the lifetime of the network.

Energy efficient routing has been the subject of intensive study in recent years. One goal of the energy aware routing (EAR) protocols is to select the best path such that the total energy consumed by the network is minimized [44]. A serious drawback of the minimum energy routing is that nodes will have wide difference in energy consumption. Nodes on the minimum energy paths will quickly drain out while the other nodes remain intact. This will result in the early death of some nodes. Another objective of the EAR is to maximize the system lifetime, which is defined as the duration when the system starts to work till any node runs out of energy, or till a certain number of nodes run out of energy, or till the network is partitioned, etc. For this purpose, multipath routing has been shown effective since it distributes the traffic load among more nodes and in proportion to their residual energies. When the energy consumption among nodes are more balanced, the mean time to node failure is prolonged, and the system lifetime is prolonged too [10, 13, 45].

2.3 Routing Overhead

Another benefit of multipath routing is the reduction of the routing overhead. Existing ad hoc routing protocols can be generally categorized into three classes: *table-driven* (or *proactive*, such as DSDV and WRP), *on-demand* (or *reactive*, such as DSR and AODV), and *hybrid* (the combination of the two, such as ZRP) [38]. Most of the performance studies indicate that on-demand routing protocols outper-

form table-driven protocols [5, 18]. The major advantage of the on-demand routing comes from the reduction of the routing overhead, as high routing overhead usually has a significant performance impact in low bandwidth wireless networks. An on-demand routing protocol attempts to discover a route to a destination "on-demand" when it is presented a packet for forwarding to that destination but it does not already know a path. It utilizes a route discovery process to find the path(s). Discovered routes are maintained by a route maintenance procedure until either the destination becomes inaccessible along every path from the source or until the route is no longer desired. The route discovery is a costly operation and it usually involves a network-wide flooding of route request packets since the node has no idea where the destination is. Typically three types of routing messages are used - *Route Request* (RREQ) and *Route Reply* (RREP) messages are used in the route discovery process to search for a route; *Route Error* (RERR) message is used to report the breakage of an intermediate link on a route back to the source. On-demand multipath protocols find multiple paths between a source and a destination in a single route discovery. A new route discovery is needed only when all the found paths fail. In [33], the authors proved that the use of multiple paths in DSR can keep correct end-to-end connection for a longer time than a single path. Therefore, by keeping multiple paths to a destination, the frequency of the costly route discovery is much lower. Moreover, in a single path routing case, when a node fails to transmit a packet to its next hop, a route error message will be sent back to the source indicating the breakage of the path. With multiple alternate paths available, nodes can actively salvage the packet by sending it to an alternate path, a route error will occur only when all the available paths fail. The occurrence of route error is therefore reduced too. Although the search for multiple paths may need more route request messages and route reply messages in a single route discovery process, the number of overall routing messages is actually reduced. Similar results have been reported in [11].

2.4 Quality of Service (QoS)

An important objective of multipath routing is to provide quality of service, more specifically, to reduce the end-to-end delay, to avoid or alleviate the congestion, and to improve the end-to-end throughput, etc.

It has been shown that multipath routing helps significantly in providing QoS by reducing the end-to-end delay for packet delivery [11]. The reduction in the end-to-end delay is not that intuitive and is attributed to multiple factors. Notice that the end-to-end delay is the latency between a packet sent at the source and received at the destination. Besides the ordinary transmission delay, propagation delay, and queuing delay, which widely exist in all IP networks, there are two types of latency

caused particularly by ad hoc on-demand routing protocols. One is the latency the protocol takes to discover a route to a destination when there is no known route to that destination. This type of latency is due to the on-demand behavior of the routing protocol and exists in all such protocols. As we mentioned in section 2.3, multipath routing effectively reduces the frequency of route discovery therefore the latency caused by this reason is reduced. The other one is the latency for a sender to "recover" when a route being used breaks. The latency resulting from broken routes could be very large because the amount of latency is the addition of the following three parts - the time for a packet to travel along the route to the node immediately before the broken link, the time for that node to detect the broken link, and the time for a route error message to travel from that node back to the source node. Among them, the time to detect a broken link could be very large because the failure of the link can only be determined after having made a certain number of attempts to transmit the packet over the broken link but failed to receive a passive or explicit acknowledgement of success. This latency caused by route errors is a significant component in the overall packet latency. Again, as we explained in section 2.3, multipath routing avoids or reduces the occurrence of route errors therefore the packet latency is further reduced. Some other factors contribute to the reduction in the end-to-end delay as well, such as the routing around the congested area, etc.

Multipath routing has been shown effective in wired networks for providing high bandwidth by allocating traffic onto multiple independent paths simultaneously so that the bandwidth of the multiple paths can be aggregated for a request which the bandwidth of any single path would not suffice. Its effectiveness is intuitive because the using of the multiple paths are independent of each other in wired networks. However, in a MANET or a WSN, shared wireless channels make the situation different. Wireless links are a relatively "soft" concept. When nodes are sharing a single wireless channel and using some medium access control (MAC) protocol such as the IEEE 802.11 to coordinate the access to the shared channel, the communication activities among the links are no longer independent. For example, as shown in Fig. 2, with the IEEE 802.11, when one node, say node 5, is transmitting to another node, say node 6, all the neighbors of both the transmitter and the receiver (i.e., nodes 1,2,3,4,7,8,12 in the example) have to keep quiet to avoid possible collision with the ongoing transmission. Therefore, node disjointness in shared channel networks does not imply the independence of the paths. Instead, the communication activities of the multiple paths affect each other very much. This problem has been referred to in [36] as *route coupling* problem and the results showed that the effect is so severe in single channel networks that it provides only negligible improvements in quality of service. The selection of multiple paths that cause less coupling is therefore an important challenge in the current

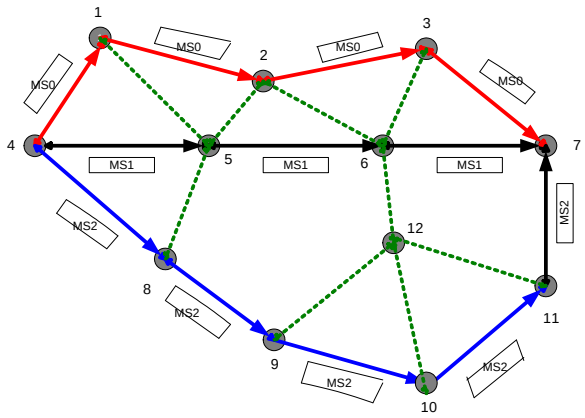


Figure 2: Route Coupling Problem in Shared Channel Multipath Routing

multipath routing protocol design. In [45] the authors considered the route coupling problem by selecting less correlated multiple paths. The correlation factor of two node-disjoint paths is defined as the number of the links connecting the two paths and it is an indicator of chances that the transmission along one path could interfere with the other. The results show that multipath routing is still an effective means of improving the throughput though the gain is not as significant as that from independent paths.

2.5 Security

A few efforts have been made to improve the network security by using multipath routing. While used for security purpose, multipath routing is often combined with secret sharing cryptography. As we mentioned in section 2.1, a (T, N) threshold secret sharing scheme has the nice property that it divides a secret into N pieces, called *shares* or *shadows*; One can derive nothing from any less than T shares, while with an efficient algorithm, the original secret can be reconstructed from any T shares [43]. Therefore, schemes combining multipath routing and secret sharing techniques typically involve the splitting of a secret by secret sharing schemes and the delivery of the shares by multipath routing. By this means, the trust is distributed to multiple nodes/paths in the network and the system is made more resilient to a collusive attack by up to a certain number of compromised nodes.

Secret sharing was originally proposed for key management in information security system [42]. Key management is also possibly the most critical and complex issue when talking about the security in a MANET or a WSN. The applicability of many other security services, such as the confidentiality and authentication, relies

on the effective and efficient key management. In [48], the authors used replication and threshold cryptography and built a more secure and more available public key management service to deal with the denial of service attacks in a MANET. The idea is to distribute the functionality of the certificate authority (CA) of a public key infrastructure (PKI) into multiple servers (or trusted nodes). In this way, both the availability and the security of the CA can be improved. In the proposed model, the threshold cryptography is used to split the system secret into multiple shares and each server holds one share; multiple servers collectively perform the functions such as signing a certificate and refreshing the key shares. The multipath routing means the routing of multiple shares from multiple servers to a single combiner. This approach was further investigated in [21] where CAs are further localized by distributing the servers more evenly in the network such that collective cryptographic operations can be done locally by neighbors of the requesting node. Another key management approach based on multipath routing is a probabilistic approach for the establishing of pairwise secret keys [8, 49]. Due to the intensive computational complexity, operations based on public key algorithms are too expensive in resource constrained MANETs/WSNs. The design of the approach is based on probabilistic key sharing and threshold secret sharing techniques. By probabilistic key sharing, every node in the network will be preloaded with a certain number of initial keys. With overwhelming probability, any pair of nodes would share one or more common keys. Then using the common keys as the seeds, the source node generates a new secret key and divides it into multiple pieces using secret sharing scheme. The multiple pieces of the secret key are then delivered to the destination via multipath routing. The multipath in their scheme is logical - multiple pieces may flow through the same physical paths while encrypted by different common keys. The SPREAD scheme we proposed is similar to the above approaches in that the scheme is based on the combination of the secret sharing and multipath routing too. However, the SPREAD scheme aims to protect the data traffic delivered across the insecure network assuming that the end-to-end encryption is either secure nor reliable. It is an enhancement to the end-to-end encryption. The multipath in our SPREAD is physically node-disjoint paths and desired path finding algorithms were proposed. The SPREAD can certainly be used to deliver the keys in stead of data traffic. However, by delivering the data traffic, SPREAD improves not only the security, but also the reliability which is a big challenge in MANETs/WSNs.

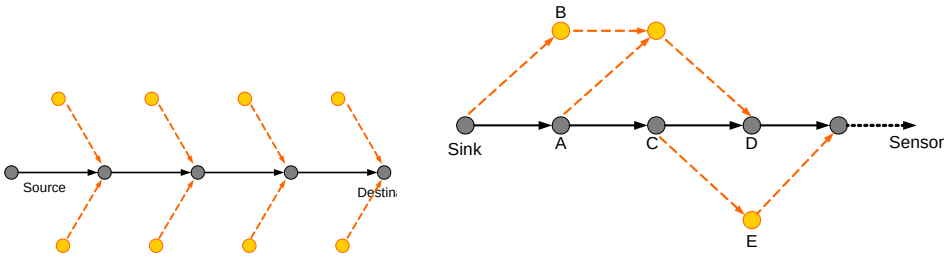
3 Multipath Routing Protocols

Routing in ad hoc networks presents great challenges. The challenge comes mainly from two aspects: constant node mobility causes frequent topological changes while limited network bandwidth restricts the timely topological updates at each router. On-demand routing has been widely developed in mobile ad hoc networks in response to the bandwidth constraints because of its effectiveness and efficiency. The multipath routing technique is another promising technique to combat problems of the frequent topological changes and link instability since the use of multiple paths could diminish the effect of possible node/link failures. Moreover, as we have discussed in section 2, multipath routing has been shown effective in improving the reliability, fault-tolerance, end-to-end delay, security, as well as in achieving load balancing, etc. However, how to actually achieve those performance benefits depends on the availability of the desired multiple paths and it further depends on the capability of the multipath finding/routing techniques. In this section, we review some multipath routing protocols available in the literature. Multipath routing is a more difficult issue than single path routing. How to find the right number of paths with desired property effectively and efficiently remains a big challenge in multipath routing research.

3.1 Partially Disjoint Paths

As we discussed in section 2, alternate path routing (APR) is an effective and efficient approach to improve the reliability of data delivery, it also helps to reduce the routing overhead and the end-to-end delay. In APR, nodes maintain multiple paths (either complete paths as in DSR or only next hop nodes as in AODV) to the destination. When the primary route fails, packets are shifted to an alternate path. A route error occurs only when all the available paths fail. For this category of multipath applications, the paths are not necessary to be completely disjoint. Partially disjoint paths can fulfill the task.

Several multipath routing protocols have been proposed in order to provide the desired alternate paths. AODV-BR (Backup Routing) [27] is one example of such routing protocols based on AODV. AODV is an on-demand single path routing protocol based on distance vector. In AODV, a source node starts the route discovery procedure by broadcasting a Route Request (RREQ) packet. Each RREQ packet contains an unique broadcast ID of the source node, which, along with the source node's IP, uniquely identifies a RREQ packet so that nodes can detect and drop duplicate RREQ packets. The RREQ packet also contains a destination-sequence number of the destination node which indicates the freshness of the packet so that nodes can detect and drop stale routing packets. An intermediate node, upon re-



(a) AODV-BR: a fish bone structure (b) Braided multipath routing

Figure 3: Alternate Path Routing

ceiving a fresh RREQ packet for the first time, needs to set up a reverse path by recording in its route table the address of the node from which the RREQ packet is received. It then rebroadcasts the RREQ packet. Duplicated RREQ packets that arrive later are simply dropped. Once the RREQ packet reaches the destination or an intermediate node which has a fresh enough route to the destination, the destination or the intermediate node unicasts a Route Reply (RREP) packet back following the reverse path established before. While the RREP packet travels along the reverse path, a forwarding route entry is set up at each node along the path. The proposed AODV-BR follows the propagation of Route Request in AODV exactly, while the route reply phase is slightly modified to construct the alternate route. Basically, the algorithm takes advantage of the broadcast nature of the wireless communication, nodes promiscuously overhear packets transmitted by their neighbors. Once a node overhears (i.e., the node is not the intended receiver of the packet) a RREP packet transmitted by its neighbor (which must be on the primary route), it records that neighbor as the next hop to the destination in its alternate route table. With this simple modification, AODV-BR is able to establish a primary route and alternate routes that look like a fish bone (see Fig. 3(a)). Data packets are delivered along the primary path. Once a node detects a link breakage, it locally broadcasts the data packet to its neighbors. Neighbor nodes that have an entry for the destination help to salvage the packet by unicasting the packet to their next hop node.

Another protocol that aims to find partially disjoint paths is the braided multipath routing proposed in [13] in order to increase resilience to node failure in WSNs. The general data dissemination follows the *directed diffusion* [16] paradigm. As with the basic directed diffusion scheme, each node computes the *gradient* and locally determines its most preferred neighbor in the direction of the intended sensor node, based on some empirical information that have initially been flooded throughout the network. Braided multipath relaxes the requirement of the node-disjointness of the complete paths. Instead, it aims to find a small number of alter-

nate paths between the sink and the intended sensor node that are partially node-disjoint with the primary path. For example, a path that differs from the primary path by one node could be an alternate path. The path finding algorithm makes use of two *path enforcement* messages and depends on some localized techniques to construct braids at each node along the primary path. The procedure can be briefly described as follows. The sink initiates by sending out a *primary path reinforcement* message to its most preferred neighbor, say *A* (as illustrated in Fig. 3(b)). In addition, the sink sends an *alternate path reinforcement* to its next preferred neighbor, say *B*. An intermediate node, say *C*, once receiving a primary path reinforcement, forwards the primary path reinforcement to its most preferred neighbor, say *D*. Therefore, the path travelled by the primary path reinforcement forms the primary path to the intended sensor node. Besides forwarding the primary path reinforcement, each node on the primary path, say *C*, also initiates an alternate path reinforcement to its next most preferred neighbor (in this example, *E*). Once a node that is not on the primary path receives an alternate path reinforcement, it forwards it to its most preferred neighbor. If the node that receives the alternate path reinforcement is on the primary path, it simply stops the the propagation of the alternate path reinforcement. By this means, an alternate path reinforcement initiated at a node on the primary path provides an alternate path that routes around the next node on the primary path but tends to rejoin the primary path later. The multipath structure formed by this technique looks like a braid (see Fig. 3(b)).

3.2 Disjoint Paths

In many multipath routing applications, disjoint paths are more attractive due to the independence of the paths. A number of multipath routing algorithms/protocols have been proposed in order to find disjoint paths in MANETs/WSNs.

There are two types of disjoint paths: *edge-disjoint* and *node-disjoint* (or *vertex-disjoint*). Despite the source and destination, node-disjoint paths have no node in common, while edge-disjoint paths do not share common edges. Clearly, node-disjoint paths are also edge-disjoint paths. From the reliability perspective, both the nodes and the wireless links are error-prone. The node failure could be caused by the physical node failure (e.g., physical damage or depletion of the battery) or the heavy congestion at the node which causes packet drop due to buffer overflow. The link failure could be caused by the breakage of the link due to the node's moving out of the transmission range, the media access contention, the multiuser interference, or any interference which causes the radio signal not being correctly decoded at the intended receiver. Both node-disjoint paths and edge-disjoint paths help in terms of reliability. For some other applications, such as the security consideration which aims to deal with compromised node problem, node-disjoint paths

are desired.

3.2.1 Edge-disjoint Paths

Diversity injection is one technique that has been proposed to find multiple disjoint paths between a single source-destination pair for on-demand routing protocols [35]. Typically, in a route discovery procedure of an on-demand routing protocol, route query (e.g., RREQ) messages are broadcasted by every node in the network. An intermediate node only responses to the first received RREQ and simply discards the duplicated ones that come later. Diversity injection technique tries to reclaim those dropped information contained in the duplicate RREQs by recording in a temporary query cache the accumulated route information contained in all the received RREQs. Since RREQ is only forwarded once at each node, each RREQ received at a node travels a different path. By claiming the path information contained at each RREQ, a node learns a diversified route information back to the source. Then during the route reply phase, when the node receives a route reply packet, it checks its temporary query cache and selects a different reverse path for the reply packet. The selection of the reverse path favors the less "heard" path so that the route reply packets bring back more diversified path information to the source.

Split Multipath Routing (SMR) is another on-demand routing protocol aiming to build maximally disjoint multiple routes [28]. It is source routing based. Notice that the propagation of the RREQs essentially builds a tree structure rooted at the source. The dropping of the duplicate RREQs tends to have one RREQ dominating the path finding process and when multiple route replies follow the reverse paths back to the source, they tends to converge when getting closer to the source. SMR modified the propagation rule of RREQs as follows - Instead of dropping every duplicate RREQs, intermediate nodes forward a duplicate packet if that RREQ packet comes from a different link other than the link from which the first RREQ is received, and whose hop count is not larger than that of the first received RREQ. SMR disables route reply from intermediate nodes. Only the destination sends out route replies. The destination replies to the first received RREQ as it is the minimum delay paths. It then waits for a certain duration of time to receive more RREQs and learns all possible routes and select the routes that is maximally disjoint to the route that is already replied. The destination sends one reply to each selected path.

Diversity injection technique and split multipath routing accumulate the path information while propagating the route request messages. The disjointness of the paths as well as the loop-freedom of each path is therefore not difficult to maintain. *Ad-hoc On-demand Multipath Distance Vector Routing* (AOMDV) is an on-

demand multipath routing protocol which is based on distance vector routing and the multiple paths are computed distributively and independently at each hop [32]. AOMDV again makes use of the information in the duplicate RREQ messages. AOMDV propagates the RREQ messages the same way as the basic AODV - only the first received RREQ is further rebroadcasted. For the duplicate RREQs, instead of simply ignoring them, AOMDV examines the path information contained in the message for potential alternate reverse path which preserves loop-freedom and link-disjointness among other paths back to the source. For each new alternate path found, the intermediate node generates a RREP message and sends it back to the source along the reverse path if it knows a forward path that has not been used in any previous RREPs for this RREQ. The destination node replies to every RREQ it receives. To ensure the loop-freedom, AOMDV uses the *destination sequence numbers* the same way as AODV to indicate the freshness of the routes. In addition, AOMDV uses the notion of *advertised hop count* to maintain multiple paths for the same sequence number. The advertised hop count is set to the hop count of the longest path available at the time when a node first advertises a path for the destination. It is reset on each new sequence number and remains unchanged until the sequence number changes. A node forms an alternate path through a neighbor only if that neighbor has a smaller advertised hop count than the node itself. Another criteria for loop-freedom guarantee is that, besides the next hop information, the route table contains the last hop information for each path. Paths have distinct first (i.e., next) hops as well as distinct last hops are disjoint.

3.2.2 Node-disjoint Paths

Node-disjoint paths are of particular interest in many application scenarios because the independence and resilience they provide. A number of node-disjoint path finding algorithms have been proposed in the literature.

AODV-Multipath (AODVM) is one multipath routing protocol that aims to find node-disjoint paths [46]. It is based on AODV. The propagation of RREQs follows the same rule as the basic AODV except that the intermediate nodes are disallowed to send route replies back to the source. Although not further propagated, duplicate RREQs received at an intermediate node are processed for possible alternate path back to the source. Every node maintains a *RREQ table* which keeps track of all the neighbors from which a RREQ is received and the corresponding cost (hop count) back to the source. When a RREQ reaches the destination, a RREP message is generated and sent back to the last hop node from which the destination receives the RREQ. The RREP packet contains an additional field *last_hop_ID* to indicate the last hop node (i.e., the neighbor of the destination). The RREP message may not follow the exact reverse path. In stead, an intermediate node determines which next

hop node the RREP should be sent to based on the information saved in the RREQ table. When an intermediate node receives a RREP, it finds from its RREQ table a shortest path back to the source and send the RREP to the corresponding next hop node. In AODVM, in order to ensure that a node does not participate in multiple paths, when a node overhears its neighbor's transmission of a RREP message, the node deletes the entry corresponding to that neighbor in its RREQ table so that it won't attempt to use that neighbor for another RREP. If an intermediate node when receiving a RREP message cannot forward it further (i.e., its RREQ table is already empty), it generates a RDER (Route Discovery Error) message and sends it back to the node from which it receives the RREP. The neighbor, upon receiving the RDER message will try to forward the RREP to a different neighbor. The number of RDERs that a particulary RREP can experience is limited so that unnecessary endless attempts can be avoided.

A node-disjoint multipath routing protocol has been described in [13] for WSNs. As we have mentioned in section 3.1, with the basic directed diffusion scheme, a node is able to determine locally its most preferred neighbor in the direction of the intended sensor node. With a primary path reinforcement message, the sink is able to find the primary path to the node. In the proposed algorithm, once the primary path is settled, the sink sends an alternate path reinforcement in order to find a node-disjoint path. Different from the partially disjoint paths algorithm (i.e., braided multipath routing), where each node on the primary path initiates an alternate path reinforcement, in order to find node-disjoint paths, only the sink initiates the alternate path reinforcement to its next preferred neighbor. That neighbor further propagates the alternate path reinforcement to its most preferred neighbor in the direction of the intended sensor node. If the node that receives the alternate path reinforcement happens to be already on the primary path, it sends a *negative reinforcement* back to the previous node, the previous node then tries its next preferred neighbor; otherwise the node continues the propagation of the alternate path reinforcement to its most preferred neighbor and so on. This mechanism allow each alternate path reinforcement sent by the sink to find a node-disjoint path between the sink and the intended sensor node. The mechanism can be extended to construct multiple node disjoint paths by sending out multiple alternate path reinforcement messages, each separated from the next by a small delay.

The multipath routing protocols proposed for our SPREAD scheme are also node-disjoint and will be presented in the following section.

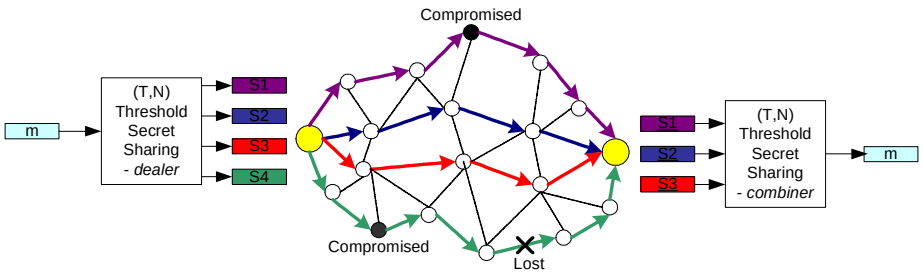


Figure 4: Basic idea of SPREAD

4 The SPREAD Scheme

4.1 SPREAD Overview

The idea of SPREAD was first proposed in [22] and then was studied as a complementary mechanism to enhance the secure data delivery service in a MANET in [24] and for secure and reliable data collection service in WSNs in [26]. The basic idea and operation of SPREAD is illustrated in Fig. 4. A secret message m is transformed into multiple shares, $S_1, S_2, \cdots$, by secret sharing scheme, and then delivered to the destination via multiple independent paths. Due to the salient features of the secret sharing and the distributed fashion of the multipath delivery, the SPREAD has been shown to be more resistent to node compromise/failure problem, namely, even a small number of paths/nodes/shares are compromised/lost, the message as a whole is not compromised/lost.

A number of coding schemes can be used to split the traffic for multipath routing in order to enhance reliability. Examples include the well-known Reed-Solomon codes, the diversity coding, the multiple description coding, etc. In our SPREAD scheme, we used the threshold secret sharing scheme for its add-on security property. A (T, N) threshold secret sharing scheme could transform a secret into N pieces, called *shares* or *shadows*. The nice property of the N shares is that form any less than T shares one cannot learn anything about the secret, while with an effective algorithm, one can reconstruct the system secret from any T out of N shares. The generation of the shares is very simple - by evaluating a polynomial of degree $(T-1)$

$$f(x) = (a_0 + a_1 x + \cdots + a_{T-1} x^{T-1}) \bmod p$$

at point $x = i$ to obtain the $i-$th share:

$$S_i = f(i)$$

where $a_0, a_1, a_2, \ldots, a_{T-1}$ are secret bits while p is a large prime number greater than any of the coefficients and can be made public.

According to the fundamental theorem of algebra, T values of a polynomial of degree $(T-1)$ can completely determine the polynomial (i.e., all its coefficients), while any fewer values cannot determine the polynomial (at least computationally difficult). Thus, any T shares can reconstruct the original secret bits, but any fewer shares cannot. Efficient $(O(T \log^2 T))$ algorithms have been developed for polynomial evaluation and interpolation [7]. Moreover, depending on the number of paths available, the (T, N) value in our SPREAD will not be large. Even the straightforward quadratic algorithms are fast enough for practical implementation.

A challenging job in any multipath routing approach is the efficient and effective multipath routing protocols. In [24] we proposed a multipath finding technique to find multiple node disjoint paths between a single source-destination pair. In fact, most of the current multipath routing protocols fall into this category. In response to the communication pattern in a WSN, we also proposed a novel N-to-1 multipath discovery protocol [26]. Instead of finding multiple paths between a specific source and a specific destination, the N-to-1 multipath discovery protocol takes advantage of the flooding in a typical route discovery procedure and is able to find multiple node-disjoint paths from every sensor nodes to the common destination (i.e., the sink node) simultaneously in one route discovery. In the rest of the section, we introduce the two routing protocols for SPREAD.

4.2 End-to-end Multipath Routing

Most of the proposed multipath routing protocols are on-demand and they work by broadcasting the route request messages throughout the network and then gathering the replies from the destination following slightly different rules (see section 3). Although those routing protocols are able to find multiple node-disjoint paths, the path set found directly through them are short in terms of number of paths and might not be optimal for a particular application as the path selection is usually based on the hop count or propagation delay, not necessary the desired property such as the security in our SPREAD scheme. We proposed a different approach in [24] to find multiple node-disjoint paths. Our approach has two major components. One is the "link cache" organization we studied in [23]. The link cache can take advantage of many multipath routing techniques, such as diversity injection, split multipath routing, etc., as mentioned in the previous section, to help to collect diversified path information. The other component is the multipath finding algorithm that is used to find maximal number of node-disjoint paths with the desired property. Once the paths are selected, our multipath routing protocol depends on source routing mechanism to route the packets along the designated multiple paths.

In DSR and other DSR-like on demand routing protocols, the route replies back to the source contain the complete node list from the source to the destination. By caching each of these paths separately, a "path cache" organization can be formed. This type of cache organization has been widely used in the proposed protocols. In stead of using the returned paths directly, we adopted a "link cache" organization in [23] where each path returned to the source is decomposed into individual links and represented in a unified graph data structure. We also proposed an adaptive link lifetime prediction and stale link removal scheme to work with the link cache. A link cache organization provides the source node a partial view of the network topology, similar to a link-state type of routing protocol. Using such a link cache organization allows us to further optimize the multipath selection based on other cost metrics. Typically by reorganizing the links, more disjoint paths can be found and paths can be selected according to some desired property, such as security or energy consumption, besides propagation delay or hop count. In addition, with a link cache, although we rely on an underlying routing protocol to provide a node with a partial view of network topology, the optimization of the path set can be done solely based on the discovered partial network topology, which is independent of the underlying routing protocols.

For the optimal selection of paths, we proposed a security related link cost function such that paths can be selected according to their security properties (i.e. the probability that the path might be compromised). Assume that a node, say, n_i, is compromised with probability q_i (a number determined from certain measurements, say, from an intrusion detection device). Assume that the overhearing does not resulting in message compromise (e.g., by link-layer encryption or directional antenna), then the probability that a path from a source s to a destination t, consisting of intermediate nodes, $n_1, n_2, \ldots, n_l$, is compromised, is given by

$$p = 1 - (1 - q_1)(1 - q_2) \cdots (1 - q_l).$$

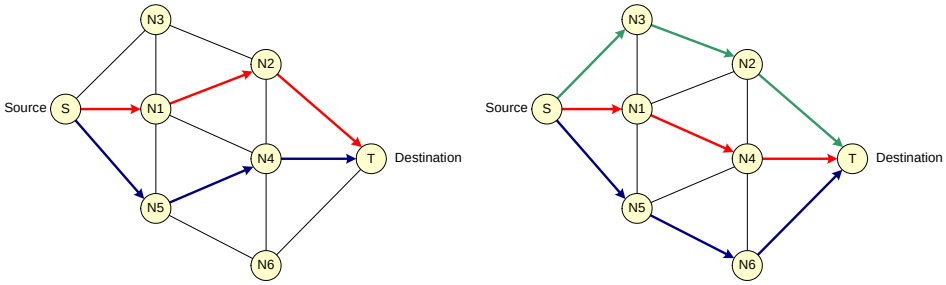
We define the link cost between n_i and n_j as

$$c_{ij} = -\log \sqrt{(1 - q_i)(1 - q_j)},$$

then we have the path cost

$$\sum c_{ij} = -\log(1 - q_1)(1 - q_2) \cdots (1 - q_l) = -\log(1 - p),$$

hence, minimizing the path cost with link metrics c_{ij} is equivalent to minimizing the path compromise probability p. If we use this link metrics, then we can find multiple paths with certain bound for path compromise probability.



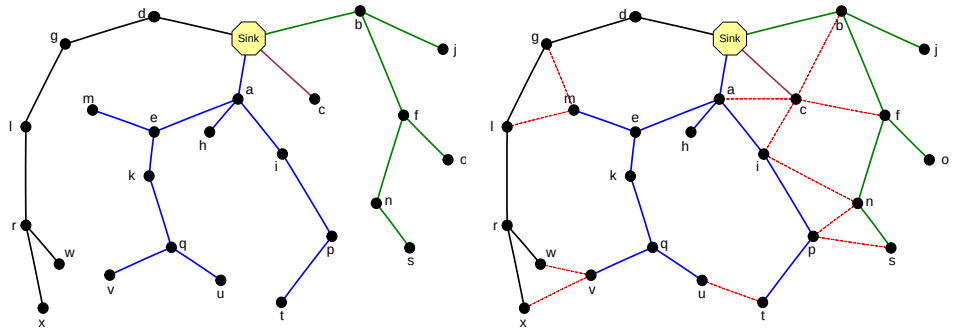
(a) 2 node-disjoint paths (b) 3 node-disjoint paths

Figure 5: Maximal Path Finding Algorithm

The next component of our end-to-end SPREAD routing protocol is a maximal node disjoint path finding algorithm to discover maximal number of secure paths. The maximal path finding algorithm is an iterative procedure. The most secure path is found first and added to the path set. In each iteration, the number of paths in the set is augmented by one. Details of the algorithm can be find in [24]. Basically each time a new path is added to the selected path set, a graph transformation is performed, which involves a vertex splitting of the nodes on the selected paths (except the source and destination node). Then, the modified Dijsktra algorithm [4] is executed to find the most secure path in the transformed graph. Then, the split nodes are transformed back to the original one, any interlacing edges are erased, and the remaining edges are grouped to form the new path set. This approach has the advantage that it can find maximal node-disjoint paths, which is solely determined by the topology. The order of the selected path does not affect the final number of paths found. As an example shown in Fig. 5, after finding the first two node-disjoint paths, the algorithm is able to regroup the links and form a path set consisting of 3 paths instead of 2.

4.3 N-to-1 Multipath Routing

While end-to-end connections are the most common communication pattern in many networks, a typical task of a WSN is the data collection where the base station broadcasts the request for the data of interest and every sensor node (or nodes that have the data of interest) sends its readings back to the base station. Therefore, a routing protocol that is able to efficiently disseminate information from the sink node to the many sensor nodes and find paths from each of the many sensor nodes back to the common sink becomes more desirable in a WSN. For this purpose, Berkeley's TinyOS sensor platform utilizes a flooding-based beaconing protocol [20]. The base station periodically broadcasts a route update. Each sensor



(a) Spanning tree created by flooding (b) A simple multipath extension of flooding

Figure 6: N-to-1 Multipath Routing

node when receiving the update for the first time rebroadcasts the update and marks the node from which it receives the update as its parent. The algorithm continues recursively till every node in the network has rebroadcasted the update once and finds its parent. What follows is that all the nodes forward the packets it received or generated to its parent until the packets reach the base station.

As illustrated in Fig. 6(a), the beaconing protocol essentially constructs a breadth first spanning tree rooted at a base station. It finds every sensor node a single path back to the base station efficiently, however reliability and also security suffer from the single path routing. The failure of a single node or link will disrupt the data flow from the node itself and all its children. Similarly, the compromise of a single node will cause the information leakage from the node and all its children. In [26] we extend our SPREAD idea into the data collection service in WSNs and present a multipath discovery algorithm that is able to find multiple node-disjoint paths from every sensor node to the base station efficiently. Then each sensor node can follow the SPREAD idea - splitting the data into multiple shares and routing them to the sink node using the multiple node-disjoint paths.

The proposed multipath discovery algorithm consists of two phases. The mechanism used in phase one, termed *branch aware flooding*, takes advantage of the simple flooding technique. Without introducing additional routing messages, the mechanism is able to find a certain number of node-disjoint paths, depending on the density of the network topology.

The general idea of the branch aware flooding is as follows. A simple flooding such as the beaconing protocol essentially constructs a breadth first spanning tree rooted at a base station. The route update initialized at the base station is first propagated to the immediate neighbors of the base station, e.g., nodes a, b, c, and d as shown in Fig. 6(b). Then from each of the immediate neighbors, it will be

further propagated and each forms a branch of the tree. The number of branches the tree has depends on the number of immediate neighbors the base station has (e.g., 4 branches in the example where different branches are distinguished by different colors). The maximum number of node-disjoint paths from any node to the base station is thus bounded by the number of the immediate neighbors of the base station. Notice that while each node has a *primary* (in most cases also the shortest) path to the base station by following its tree links up, a link between two nodes that belong to two different branches provides each node an alternate disjoint path to the base station through the other. For example, as shown in Fig. 6(b), while node w has the primary path $(w - r - l - g - d - Sink)$ back to the base station, it learns another alternate path $(w - v - q - k - e - a - Sink)$ from node v which is not in the same branch as w when w overhears v's broadcast.

Since each node needs to broadcast the route update message once anyway, the branch aware flooding is able to find a certain number of node-disjoint paths without any additional routing messages. However the limitation of this method is that it only finds disjoint paths at the nodes where there are direct links to other branches. In the same example, notice that if w further propagates the disjoint paths it learned to its parent or siblings/cousins (but not necessary the children), its parent or siblings/cousins might learn a new disjoint path as well. For example, node r has the primary path $(r - l - g - d - Sink)$. If it hears a disjoint path $(w - v - q - k - e - a - Sink)$ from w and it does not yet know a path through branch a, it learns a new disjoint path $(r - w - v - q - k - e - a - Sink)$. Therefore, in order to maximize the number of disjoint paths each node may have, a second phase of the multipath routing algorithm is designed which is to further propagate the disjoint paths found in the first phase. The tradeoff of the second phase is that it finds more disjoint paths with additional routing messages. Details of the N-to-1 multipath routing protocol can be found in [26].

4.4 SPREAD Summary

Multipath routing has been a promising technique in MANEETs and WSNs to deal with the unreliable data communications. It is also a feasible technique due to the dense deployment of nodes in the MANETs/WSNs. SPREAD is an innovative scheme which combines the multipath routing and the secret sharing techniques to address both reliability and security issues. The end-to-end SPREAD scheme adopts the concurrent multiapth routing and provides more secure data transmission when messages are transmitted across the insecure network. A certain amount of redundancy can be added without affecting security through the optimal share allocations onto each selected path (see [24] for details on optimal share allocation schemes). The N-to-1 SPREAD scheme distinguishes from all previous work in

that the mulitpath discovery protocol is receiver-initiated (in contrast to the common source-initiated route discovery) and the protocol is efficient in that it finds multipath from every sensor node to the base station simultaneously in one route discovery procedure, which fits the special communication pattern (i.e., multiple senders to a single receiver) in the WSN very well. For the data delivery, the N-to-1 SPREAD adopts concurrent multipath routing which spread traffic onto multiple disjoint paths simultaneously between the sensor node and the sink. In addition, taking advantage of the multiple paths available at each node, it also adopts the per-hop multipath packet salvaging technique which uses the multipath alternately and helps to improve the reliability of each packet delivery/path significantly. By the combination of the two, the overall scheme improves both security and reliability with none or very little information redundancy.

A few remarks are necessary here. First, the SPREAD scheme considers the security and reliability when messages are transmitted across the network, assuming the source and destination are trusted. It improves the security and reliability of the end-to-end message delivery in the sense that it is resilient to a certain number of compromised/faulty nodes but it does not improve the security of each individual node. Secondly, the SPREAD scheme cannot address the security, i.e., confidentiality, alone, it only statistically enhances such service. For example, it is still possible for adversaries to compromise all the shares, e.g. by collusion. Finally, the SPREAD can be made adaptive in the sense that the source node could make final decision whether a message is delivered at certain time instance according to the security level and the availability of multiple paths. Moreover, the chosen set of multiple paths may be changed from time to time to avoid any potential capture of those multiple shares by adversaries.

5 Conclusion

Multipath routing has been a promising technique in MANETs and WSNs. It has been shown through both theoretical analysis and simulation results that multipath routing provides many performance benefits, including the improved fault tolerance, security, and reliability, improved routing efficiency and reduced routing overhead, more balanced traffic load and energy consumption, reduced end-to-end latency and aggregated network bandwidth, etc. Significant research efforts have been made and are continuously being made in developing multipath routing protocols and multipath packet forwarding techniques in order to achieve the above-mentioned performance gains effectively and efficiently. Nevertheless, many issues that directly related to the application of the multipath routing remain untouched, such as the integration of the multipath routing into the current single path rout-

ing paradigm, the synchronization of the packets among the multiple paths, and the interfaces of the multipath routing protocols to other layers of protocol in the network protocol stack, etc.

Due to the space limitations, we are only able to introduce the basic concept of multipath routing, highlight the fundamental techniques used to find the multiple paths, and outline the essential idea what and why it can help in the performance. For detailed algorithms/protocols as well as the performance evaluations, interested readers are referred to respective publications.

References

[1] E. Ayanoglu, I. Chih-Lin, R. D. Gitlin, J. E. Mazo, "Diversity coding for transparent self-healing and fault-tolerant communication networks", *IEEE Transaction on Communications*, 41(11):1677-1686, November 1993.

[2] A. F. Akyildiz, W. Su, Y. Sankarasubramainiam, E. Cayirci, "A survey on sensor networks", *IEEE Communications Magazine*, August 2002

[3] A. Banerjea, "On the Use of Dispersity Routing for Fault Tolerant Real-time Channels", *European Transactions on Telecommunications*, 8(4):393-407, July/August 1997

[4] R. Bhandari, Survivable Networks - Algorithms for diverse routing, *Kluwer Academic Publisher*, 1999.

[5] J. Broch, D. Maltz, D. Johnson, Y-C. Hu, J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocol", *The 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98)*, pp. 85-97, Dallas, TX, October 1998

[6] J. Chen, "New Approaches to Routing for Large-Scale Data Networks", Ph.D Dissertation, Rice University, 1999

[7] T. Cormen, C. Leiserson, R. Rivest, *Introduction to algorithms*, MIT Press, 1990.

[8] H. Chan, A. Perrig, D. Song, "Randon key predistribution schemes for sensor networks", *IEEE Symposium on Security and Privacy (SP'03)*, Oakland, CA, May 2003

[9] I. Cidon, R. Rom, Y. Shavitt, "Analysis of multi-path routing", *IEEE/ACM Transactions on Networking*, 7(6):885-896, Dec 1999

[10] J. Chang, L. Tassiulas, "Energy conserving routing in wireless ad hoc networks", *Proceedings INFOCOM 2000*, Mar 2000.

[11] S.K. Das, A. Mukherjee, et al, "An adaptive framework for QoS routing through multiple paths in ad hoc wireless newtorks", *J. Parallel Distributed Computing*, 63(2003)141-153

[12] S. De, C. Qiao, H. Wu, "Meshed multipath routing: an efficient strategy in sensor networks", *IEEE Wireless Communications and Networking Conference (WCNC'03)*, New Orleans, LA, Mar 2003

[13] D. Ganesan, R. Govindan, S. Shenker, D. Estrin, "Highly-resilient, energy-efficient multipath routing in wireless sensor networks", *Mobile Computing and Communication Review*, 5(4):10-24, Oct 2001.

[14] E. Gustafsson, G. Karlsson, "A literature survey on traffic dispersion", *IEEE Networks*, 11(2):28-36, March/April 1997.

[15] N. Gogate, S. S. Panwar, "Supporting video/image applications in a mobile multihop radio environment using route diversity", *IEEE International Conference on Communications (ICC'99)*, Vancouver, Canada, June 1999.

[16] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, "Directed diffusion for wireless sensor networks", *IEEE/ACM Transactions on Networking*, 11(1):2-16, Feb 2003.

[17] D. B. Johnson, D. A. Maltz, Y-C. Hu, J. G. Jetcheva, "The dynamic source routing protocol for mobile ad hoc networks", IETF Internet Draft, draft-ietf-manet-dsr-06.txt, Nov 2001

[18] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, M. Degermark, "Scenario-based performance analysis of routing protocols for mobile ad hoc networks", *The 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*, pp.195-206, Seattle, WA, August 1999

[19] S. Jain, Yi Lv, S. R. Das, "Exploiting path diversity in the link layer in wireless ad hoc networks," Technical Report, SUNY at Stony Brook, CS department, WINGS Lab, July 2003.

[20] C. Karlof, D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures", *Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, 1(2-3):293-315, September 2003

[21] J. Kong, P. Zerfos, H. Luo, S. Lu and L. Zhang, "Providing robust and ubiquitous security support for manet," *Proceedings of the 9th IEEE International Conference on Network Protocols(ICNP)*, pp. 251 -260, 2001.

[22] W. Lou, Y. Fang, "A multipath routing approach for secure data delivery", *IEEE Military Communications Conference (MILCOM 2001)*, Mclean, VA, USA, Oct 2001

[23] W. Lou, Y. Fang, "Predictive caching strategy for on-demand routing protocols in ad hoc networks", *Wireless Networks*, vol.8, issue 6, pp.671-679, Nov 2002

[24] W. Lou, W. Liu, Y. Fang, "SPREAD: Enhancing data confidentiality in mobile ad hoc networks", *IEEE INFOCOM 2004*, HongKong, China, March 2004

[25] W. Liu, Y. Zhang, W. Lou, Y. Fang, "Scalable and Robust Data Dissemination in Wireless Sensor Networks", *IEEE GLOBECOM 2004*, Dallas, TX, Dec 2004

[26] W. Lou, Y. Zhang, W. Liu, Y. Fang, "A multipath protocol for secure and reliable data collection in wireless sensor networks", technical report, ECE department, Worcester Polytechnic Institute, June 2004

[27] S.-J. Lee, M. Gerla, "AODV-BR: backup routing in ad hoc networks", *IEEE Wireless Communications and Networking Conference (WCNC'00)*, Sep 2000

[28] S.-J. Lee, M. Gerla, "Split multipath routing with maximally disjoint paths in ad hoc networks", *International Conference on Communications (ICC'01)*, Helsinki, Finland, June 2001.

[29] N. F. Maxemchuk, "Dispersity routing", *International Conference on Communications (ICC '75)*, pp.41.10-41.13, San Francisco, CA, June 1975.

[30] N. F. Maxemchuk, "Dispersity routing in high speed networks", *Computer Networks and ISDN Systems*, 25(6):645-661, 1993.

[31] N. F. Maxemchuk, "Dispersity routing on ATM networks", *IEEE INFOCOM'93*, vol.1, pp.347-57, San Francisco, CA, Mar 1993.

[32] M. K. Marina, S. R. Das, "On-demand multipath distance vector routing in ad hoc networks", *9th International Conference on Network Protocols*, Riverside, CA, November, 2001

[33] A. Nasipuri, R. Castaneda, S. R. Das, "Performance of multipath routing for on-demand protocols in mobile ad hoc networks", *Mobile Networks and Applications*, 6(4):339-349, 2001

[34] V. D. Park, M. S. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks", *IEEE INFOCOM'97*, pp. 1405-1413, Kobe, Japan, April 1997

[35] M. R. Pearlman, Z. J. Haas, "Improving the performance of query-based routing protocols through diversity injection", *IEEE Wireless Communications and Networking Conference (WCNC'99)*, New Orleans, LA, September 1999.

[36] M.R. Pearlman, Z.J. Haas, P. Sholander, S. S. Tabrizi, "On the impact of alternate path routing for load balancing in mobile ad hoc networks", *The ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC'00)*, Boston, MA, August 2000.

[37] P. Papadimitratos, Z.J. Haas, E. G. Sirer, "Path set selection in mobile ad hoc networks", *The ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'2002)*, EPFL Lausanne, Switzerland, June 2002.

[38] E. M. Royer, C-K Toh, "A review of current routing protocols for ad hoc mobile wireless networks", *IEEE Personal Communications*, 6(2):46-55, April 1999

[39] D. Sidhu, S. Abdallah, R. Nair, "Congestion Control in High Speed Networks via Alternative Path Routing", *Journal of High Speed Networks*, 2(2):129-144, 1992.

[40] N. Taft-Plotkin, B. Bellur, R. Ogier, "Quality-of-Service Routing Using Maximally Disjoint Paths", *1999 Seventh International Workshop on Quality of Service*, London, UK, June 1999

[41] A. Tsirigos, Z. J. Haas, "Multipath routing in the presence of frequent topological changes", *IEEE Communication Magazine*, 39(11):132-138, November 2001

[42] A. Shamir, "How to Share a Secret," *Communications of the ACM*, 22(11):612-613, November 1979.

[43] G. J. Simmons, "An Introduction to Shared Secret and/or Shared Control Schemes and The Application," *Contemporary Cryptology: The Science of Information Integrity*, IEEE Press, pp.441-497, 1992.

[44] S. Singh, M. Woo, C.S. Raghavendra, "Power aware routing in mobile ad hoc networks", *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '98)*.

[45] K. Wu, J. Harms, "Performance study of a multipath routing method for wireless mobile ad hoc networks", *9th international symposium on modeling, analysis and simulation of computer and telecommunication system (MASCOTS'01)*, Cincinnati, Ohio, August 2001.

[46] Z. Ye, S. V. Krishnamurthy, S. K. Tripathi, "A framework for reliable routing in mobile ad hoc networks", *IEEE INFOCOM 2003*, Sanfrancisco CA, Mar 2003

[47] J. Yang, S. Papavassiliou, "Improving network security by multipath traffic dispersion", *IEEE Military Communications Conference (Milcom'01)*, McLean, VA, October 2001

[48] L. Zhou, Z. J. Haas, "Securing ad hoc networks", *IEEE Network Magazine*, 13(6):24-30, November/December 1999.

[49] S. Zhu, S. Xu, S. Setia, S. Jajodia, "Establishing pairwise keys for secure communication in ad hoc networks: A probabilistic approach", *11th IEEE International Conference on Network Protocols (ICNP'03)*, Atlanta, GA, November 2003