Hybrid Routing: The Pursuit of an Adaptable and Scalable Routing Framework for Ad Hoc Networks

Prince Samar, Marc R. Pearlman, and Zygmunt J. Haas
School of Electrical and Computer Engineering
Cornell University
323 Frank Rhodes Hall
Ithaca, NY 14853
{samar, pearlman, haas}@ece.cornell.edu
http://wnl.ece.cornell.edu

Abstract

Advances in ad hoc network research have opened the door to an assortment of promising military and commercial applications for ad hoc networks. However, because each application has unique characteristics (such as traffic behavior, device capabilities, mobility patterns, operating environments, etc), routing in such a versatile environment is a challenging task and numerous protocols have been developed to address it. While many protocols excel for certain types of ad hoc networks, it is clear that a single basic protocol cannot perform well over the entire space of ad hoc networks. To conform to any arbitrary ad hoc network, the basic protocols designed for the edges of the ad hoc network design space need to be integrated into a tunable framework.

The Zone Routing framework demonstrates how multi-scoping can provide the basis for a hybrid routing protocol framework. Zone Routing proactively maintains routing information for a local neighborhood called the routing zone (local scope), while reactively acquiring routes to destinations beyond the routing zone. In this paper, we review the Zone Routing concept and propose Zone Routing with independently sized routing zones capability. With this capability, each of the nodes in the network can adaptively configure its own optimal zone radius in a distributed fashion. We show that the performance of Zone Routing is significantly improved by the ability to provide fine-tuned adaptation to local and temporal variations in network characteristics.

I. Ad Hoc Networks Overview

An ad hoc network is a self-organizing wireless network made up of mobile nodes and requiring no fixed infrastructure. The limitations on power consumption imposed by portable wireless radios result in a node transmission range that is typically small relative to the span of the network. To provide communication throughout the entire network, nodes are designed to serve as routers. The result is a distributed multi-hop network with a time-varying topology.

Because ad hoc networks do not rely on existing infrastructure and are self-organizing, they can be rapidly deployed to provide robust communication in a variety of hostile

This work has been supported in part by the DoD Multidisciplinary University Research Initiative (MURI) program administered by the Office of Naval Research under the Grant number N00014-00-1-0564 and by the National Science Foundation grant numbers ANI-9980521 and ANI-0081357.

environments. This makes ad hoc networks very appropriate for providing tactical communication for the military, law enforcement and emergency response efforts. Ad hoc networks can also play a role in civilian forums, such as the electronic classroom, convention centers, and construction sites. With such a broad scope of applications, it is not difficult to envision ad hoc networks operating over a wide range of coverage areas, node densities, and node velocities.

This potentially wide range of ad hoc network operating configurations poses a challenge for developing efficient routing protocols. On one hand, the effectiveness of a routing protocol increases as network topology information becomes more detailed and up-to-date. On the other hand, in an ad hoc network, the topology may change quite often, requiring large and frequent exchanges of control information among the network nodes. This is in contradiction with the fact that all updates in the wireless communication environment travel over the air and are, thus, costly in resources.

II. Brief Survey of Basic Ad Hoc Routing Protocols

Existing routing protocols can be classified either as proactive or as reactive. Proactive protocols attempt to continuously evaluate the routes within the network, so that when a packet needs to be forwarded, the route is already known and can be immediately used. Early applications of proactive routing schemes for ad hoc networks were Distance Vector protocols based on the Distributed Bellman-Ford (DBF) algorithm [2]. Modifications to the basic DBF algorithm (i.e., [4], [7], and [27]) were proposed to address inherent problems of convergence and excessive traffic (both of which can be quite severe problems in ad hoc networks, where bandwidth is scarce and topologies are often very dynamic). The convergence problem has also been addressed by the application of Link State protocols such as TBRPF [1], STAR [8], ALP [9], and GSR [3] to the ad hoc environment. In general, Link State protocols converge faster than Distance Vector protocols, but may lead to a lot of control traffic. Motivation to both improve protocol convergence and to reduce control traffic has led to the development of proactive path finding algorithms, which combine the features of the Distance Vector and Link State approaches. Realizations of the path finding algorithms, like the Wireless Routing Protocol (WRP) ([19] and [20]), are able to eliminate the "counting-to-infinity" problem and to reduce the occurrence of temporary loops, often with less control traffic than traditional Distance Vector schemes.

In contrast, reactive protocols[1] invoke a route determination procedure on an on-demand basis. The reactive route discovery is usually based on a query-reply exchange, where the route query uses some flooding-based process to reach the desired destination. In the case of the Temporally Ordered Routing Algorithm (TORA) [21], the route replies are also flooded, in a controlled manner, distributing routing information in the form of directed acyclic graphs (DAGs) rooted at the destination. In contrast, the Dynamic Source Routing (DSR) [16] and Ad hoc On Demand Distance Vector (AODV) [28] protocols unicast the route reply back to the querying source, along a path constructed during the route query phase. In the case of DSR, the routing information is accumulated in the query packet, and the complete sequence of nodes is returned to the source (to be used for source routing of the actual user data). AODV, on the other hand, distributes the discovered route in the form of next-hop information stored at each node in the route. The on-demand discovery

[1] *Reactive protocols* are also referred to as *on-demand protocols*.

of routes can result in much less traffic than the standard, proactive Distance Vector or Link State schemes, especially when innovative route maintenance schemes are employed. However, the reliance on flooding of the reactive schemes may still lead to considerable control traffic in the highly versatile ad hoc networking environment. Moreover, due to the large increase in control traffic at the times of route discovery, the delay of the route-discovery process in reactive protocols can be significant.

III. Multi-scope Routing

All else being equal, the value of information decreases with respect to the distance from the information source. For example, a node cannot compute routes without knowing any of its neighbors. On the other hand, a node may make near-optimal forwarding decisions, even in spite of outdated or missing distant state information.

This relationship between information value and distance is extremely valuable for routing protocol design. Simply distributing the same information, at the same rate to all nodes in the network does not provide the most "bang for the buck." There is more value in providing nearby nodes with more frequent update and/or detailed information, at the expense of keeping more distant nodes less informed.

To some extent, most basic protocols exhibit some degree of multi-scope behavior. Many proactive routing protocols monitor the status of neighbor connectivity through neighbor broadcast HELLO beacons, which occur at a faster rate than the global link state (or distance vector) advertisements. In many reactive routing protocols, route discovery is based on querying on a global scale, whereas subsequent route repair utilizes local querying, constrained by a time-to-live (TTL) packet hop counter

The high quality local route information provided by multi-scope routing can be used to provide an assortment of new and enhanced services. By identifying overlaps in local connectivity, broadcast message can be distributed to all nodes more efficiently (e.g., OLSR's multipoint relay [6]). Moreover, local exchange of route information can be further exploited to provide a *bordercast* query distribution service, in which only a subset of the network's nodes need to be queried. Such a service can be applied to global route discovery, name-address translation, and general database lookups. In the case of global reactive protocols, once a route has been discovered, changes in local connectivity can be quickly identified, allowing for either proactive route repair or proactive route shortening [23]. Local multi-hop feedback of link layer acknowledgments can be used to discover and reliably use unidirectional links [24]. Intelligent node participation/sleep-mode algorithms can use local route information to determine if a node's absence would compromise network connectivity.

Perhaps the most familiar examples of multi-scope routing are the various flavors of hierarchical routing [15] [5] [26]. In basic clustered routing, nodes are aggregated into subnets. Each node knows the topology of its subnet through a local proactive protocol. On a global scale, each subnet is represented by a clusterhead, which knows the connectivity to other subnets' clusterheads, but not the details of the other subnets' topologies. Nodes are located in the hierarchy through relative addressing that associates nodes with clusterheads. This two level example can be easily extended by grouping clusterheads into intermediate level subnets, thus creating a deeper hierarchy.

A variation on the clustered routing is landmark routing (e.g. LANMAR [10]). As in the previous example, nodes are organized into local subnets and assume hierarchical addresses. However, the clusterheads are replaced with globally visible landmarks. A global distance vector routing protocol is used to provide all nodes with routes to the landmarks. The role of the landmark is to identify the general location of the associated subnet. Data packets are forwarded toward the landmark, until they reach the subnet, at which point the subnet nodes can forward the packet directly to the destination.

Another hierarchical routing approach is based on the concept of a core. In these schemes, local topology is proactively monitored for the purpose of selecting a set of core nodes, such that every node has at least one core node neighbor. The purpose of the core nodes is to determine routes on behalf of the nodes that they cover. This is generally accomplished through global route discovery that is carried through the core. Although the route discovery occurs in the core, the core nodes apply knowledge of their local topology to construct routes that do not necessarily pass through the core. In addition to this basic operation, the CEDAR protocol [29] introduces an interesting local scoping behavior by advertising higher quality links (e.g. high capacity links) over greater distances. In the Dynamic Virtual Backbone scheme [17], which employs a similar concept of the core, the route queries are restricted within the virtual backbone.

Specialized node roles and regional node addressing help hierarchical routing protocols to scale with network size, especially when there is a structure in the underlying network connectivity (for example, group mobility) to be exploited [18]. However, as network behavior becomes less coordinated, the overhead of the hierarchy maintenance (e.g. clusterhead election, node re-addressing) becomes a limiting factor for scaling. In addition, hierarchical routing may introduce uneven resource utilization, traffic hotspots, and in some cases, sub-optimal routing.

It is also possible to provide multi-scope routing without the limitations and overhead of hierarchy management. For example, in FSR [25] a node's link state is distributed over various distances (scopes), with longer distance updates occurring at lower frequencies. This provides each node with a fresh view of the surrounding topology, but a more dated view of farther network regions. The less accurate distant views effectively serve as landmarks, getting data forwarded in the right general direction. As the data approaches the destination, the path to the destination becomes more accurate and the forwarding more refined.

Additional benefits of multi-scope routing can be realized when larger scope protocols are able to exploit the information provided by a smaller scope. Two protocols that exhibit this kind of scope integration are OLSR and Zone Routing. In OLSR [6], an extended neighbor discovery provides each node with the topology of its surrounding two hops. This local information is used to provide an efficient global link state broadcast, based on multipoint relay. Multipoint relay identifies a "minimal" subset of neighbors needed to relay a message, such that all nodes two hops away will receive the message. In the case of Zone Routing, a proactive routing protocol is used to provide each node with a view of its surrounding "routing zone" topology. This local information enables a efficient query distribution service (bordercasting), which is used by a global reactive route discovery protocol. (Zone Routing is described in more detail later in this paper). For OLSR and Zone Routing, global protocol efficiency increases with the size of the local "zone." The cost of local vs global scope can be traded-off, and ultimately optimized, through the adjustment of a single parameter – the zone radius.

IV. Protocol Hybridization

The diverse applications of ad hoc network pose a challenge for a single protocol that operates efficiently across a wide range of operational conditions and network configurations. Each of the purely proactive or purely reactive protocols described above performs well in a limited region of this range. For example, reactive routing protocols are well suited for networks where the "call to mobility" ratio is relatively low. Proactive routing protocols, on the other hand, are well suited for networks where this ratio is relatively high. Figure 1 shows the ad hoc network design space with node mobility and call rate as the two dimensions, and the approximate regions where each of these two kinds of protocols performs well. The performance of either class of protocols degrades, when they are applied to regions of ad hoc network space between the two extremes.

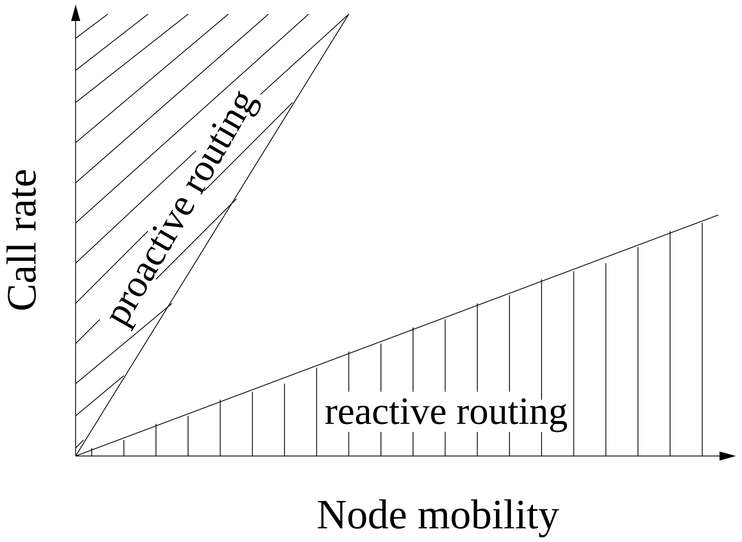


Fig. 1. The ad hoc network design space and the applicability of proactive vs. reactive routing.

Given multiple protocols, each suited for a different region of the ad hoc network design space, it makes sense to capitalize on each protocol's strengths by combining them into a single framework (i.e., hybridization). In the most basic hybrid framework, one of the protocols would be selected based on its suitability for the specific network's characteristics. Although not an elegant solution, such a framework would perform as well as the best suited protocol for any scenario, and outperform either protocol over the entire ad hoc network design space. However, by not using both protocols together, this approach fails to capitalize on the potential synergy that would make the framework perform as well or better than either protocol *for any given scenario.*

A more promising approach for protocol hybridization is to have the base protocols operate simultaneously, but with different "scopes." For the case of a two-protocol framework, protocol A would operate locally, while the operation of protocol B would be global. The key to this framework is that the local information acquired by protocol A is used by protocol B to operate in a more efficient manner. This framework can be tuned to network behavior simply by adjusting the size of the protocol A's scope. In one extreme configuration, the scope of protocol A is reduced to nothing, leaving protocol B to run by itself. As the scope of protocol A is increased, the information provided to protocol B increases, thereby decreasing protocol B's overhead. At the other extreme, protocol A is made global, eliminating the load of protocol B altogether. So, at either extreme, the framework defaults to the operation of an individual protocol. In the wide range of intermediate configurations, the framework performs better than either protocol on its own.

V. Framework Tuning

The motivation behind hybrid routing and multi-scope routing is to provide a framework that can be configured to match the network's operational condition and configuration. Therefore, an integral component of the framework is a tuning mechanism. In particular, the three basic ingredients for tuning are a means for measuring relevant network characteristics, a mapping of these measurements to a framework configuration, and a scheme to update the configurations of affected nodes.

The most basic approach to tuning is to determine the network characteristics and proper configuration offline, prior to the network deployment. Typically, the configuration would be determined through network simulation and subsequent parameter optimizations. The nodes are loaded with the proper configuration and then activated. When it is not possible to pre-configure all nodes individually, a small number of nodes may be configured, and this configuration can be shared with other nodes as part of an automatic configuration procedure.

The main advantages of pre-configuration are that it requires limited network intelligence, low real-time processing overhead, and ensures stable and consistent configuration. However, for many applications, pre-configuration is not an option. Pre-configuration requires a central configuration authority, which may not exist for distributed applications. In addition, the network characteristics may not be known apriori, or may vary over time, preventing the offline analysis and reducing the effectiveness of the static configuration.

Ad hoc networks naturally lend themselves to dynamic reconfiguration. Through the course of normal operation, nodes directly measure (or infer) local network characteristics. Each node may use its own local measurements for independent self-configuration. Alternatively, the measurements could be relayed to a central configuration node or shared with surrounding nodes for a distributed configuration approach.

At first glance, centralized dynamic reconfiguration may appear to prevent inconsistent configuration, as is the case for centralized static configuration. However, the multi-hop nature of ad hoc networks makes it impossible to reliably perform tightly synchronized configuration updates for all nodes. This means that, for some period of time, the network could be in an inconsistent state. As this also affects distributed and independent reconfiguration, it is necessary that a dynamically tunable routing framework be able to deal with, and potentially exploit, non-uniform node configurations. The way in which a routing framework supports non-uniform configuration depends on its particular design. We will later see in the sequel how non-uniform configuration is supported in the Zone Routing framework.

With support for non-uniform configuration, reconfiguration decisions, and the associated measurement/control traffic, can be kept local (or eliminated altogether in the case of independent reconfiguration), making the tuning mechanism scalable. Furthermore, the framework can be fine-tuned to adapt to regional changes or even to nodal behavior, rather than broadly tracking average network behavior. This can lead to significant performance improvements, especially in the case of networks where node behavior has regional dependencies.

VI. The Zone Routing Framework

Protocol hybridization, multi-scope operation, and dynamic reconfiguration, the key features of adaptable and scalable routing, form the basis of the Zone Routing framework. At a local level, a proactive routing protocol provides a detailed and fresh view of each node's surrounding local topology (routing zone). The knowledge of local topology is used to support services such as proactive route maintenance, unidirectional link discovery, and guided message distribution. One particular message distribution service, called bordercasting, directs queries throughout the network across overlapping routing zones. Bordercasting is used in place of traditional broadcasting to improve the efficiency of a global reactive routing protocol.

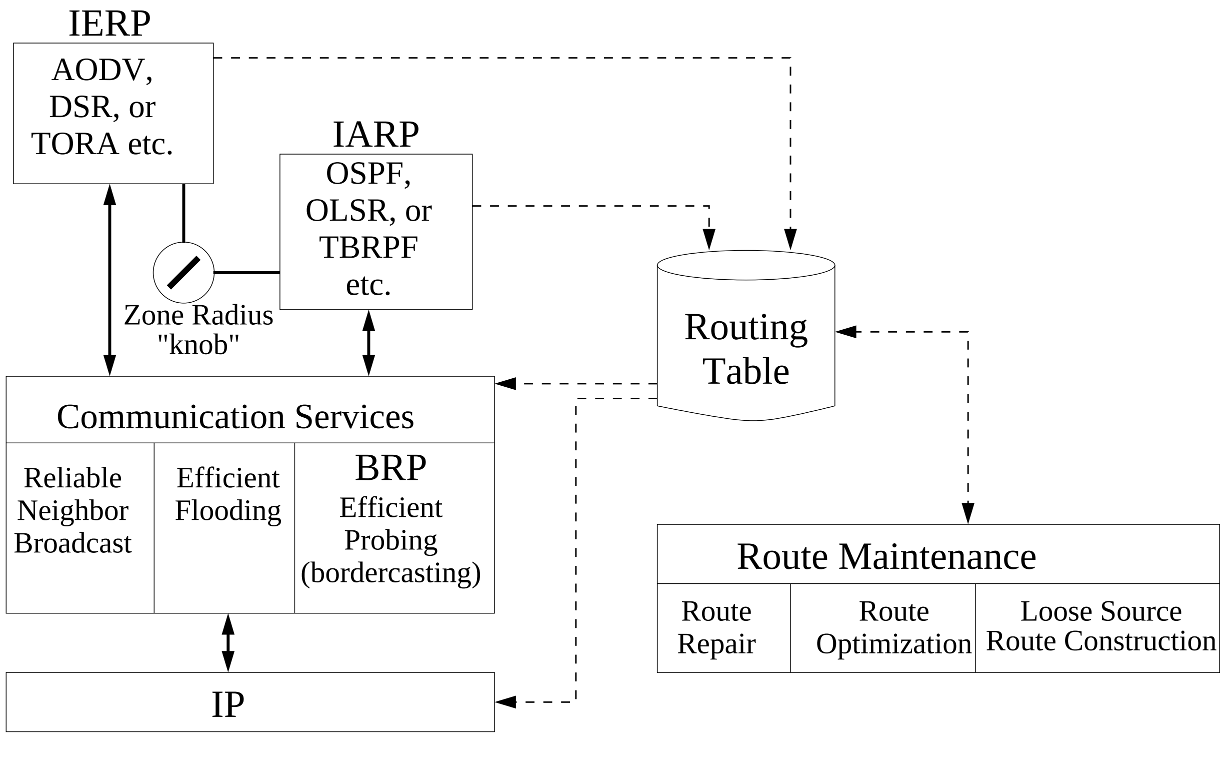


Fig. 2. Architecture of the Zone Routing Framework.

The benefits provided by routing zones, compared with the overhead of proactively tracking routing zone topology, determine the optimal framework configuration. As network conditions change, the framework can be dynamically reconfigured through adjustment of each node's routing zone

In the following sections, we describe the routing and dynamic configuration components of the Zone Routing framework in more detail.

A. Local Proactive (Intrazone) Routing

In Zone Routing, the Intrazone Routing Protocol (IARP) proactively maintains routes to destinations within a local neighborhood, which we refer to as a routing zone. More precisely, a node's routing zone is defined as a collection of nodes whose minimum distance in hops from the node in question is no greater than a parameter referred to as the zone radius. Note that each node maintains its own routing zone. An important consequence, as we shall see, is that the routing zones of neighboring nodes overlap.

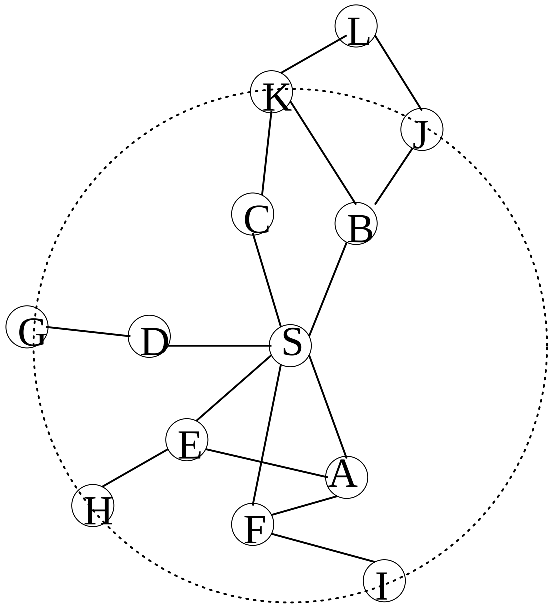


Fig. 3. A Routing Zone of radius 2 hops.

Figure 3 illustrates the routing zone concept with a routing zone of radius 2 hops. This particular routing zone belongs to node S, which we refer to as the central node of the routing zone. Nodes A through K are members of S's routing zone. Node L, however, is three hops away from S, and is therefore outside of S's routing zone. An important subset of the routing zone nodes is the collection of nodes whose minimum distance to the central node is exactly equal to the zone radius. These nodes are aptly named *peripheral nodes*. In our example, nodes G-K are peripheral nodes of node S. We typically illustrate a routing zone as a circle centered around the central node. However, one should keep in mind that the zone is not a description of physical distance, but rather nodal connectivity (hops).

The construction of a routing zone requires a node to know first who its neighbors are. A neighbor is defined as a node with whom direct (point-to-point) communication can be established and is, thus, one hop away. Identification of a node's neighbors may be provided directly by the media access control (MAC) protocols, as in the case of polling-based protocols. In other cases, neighbor discovery may be implemented through a separate Neighbor Discovery Protocol (NDP). Such a protocol typically operates through the periodic broadcasting of "hello" beacons. The reception (or quality of reception) of a "hello" beacon can be used to indicate the status of a connection to the beaconing neighbor.

Neighbor discovery information is used as a basis for the IARP. IARP can be derived from globally proactive link state routing protocols that provide a complete view of network connectivity (for example, the Shortest Path First OSPF). The base protocol needs to be modified to ensure that the scope of the route updates is restricted to the radius of the node's routing zone [13]. In this paper, IARP is based on a simple, timer-based, link state protocol. To track the topology of R hop routing zones, each node periodically broadcasts its link state for a depth of R hops (controlled by a time-to-live (TTL) field in the update message).

B. Bordercast-Based Global Reactive (Interzone) Routing

Route discovery in the Zone Routing framework is distinguished from standard broadcast-based route discovery through a message distribution service known as *bordercasting* [14]. Rather than blindly broadcasting a route query from a neighbor to a neighbor, bordercasting allows the query to be directed outward, toward regions of the network (specifically, toward peripheral nodes) that have not yet been "covered" by the query. (A covered node is one that belongs to the routing zone of a node that has received a route query). The query control mechanisms reduce route query traffic by directing query messages outward from the query source and away from covered routing zones, as illustrated in figure 4.

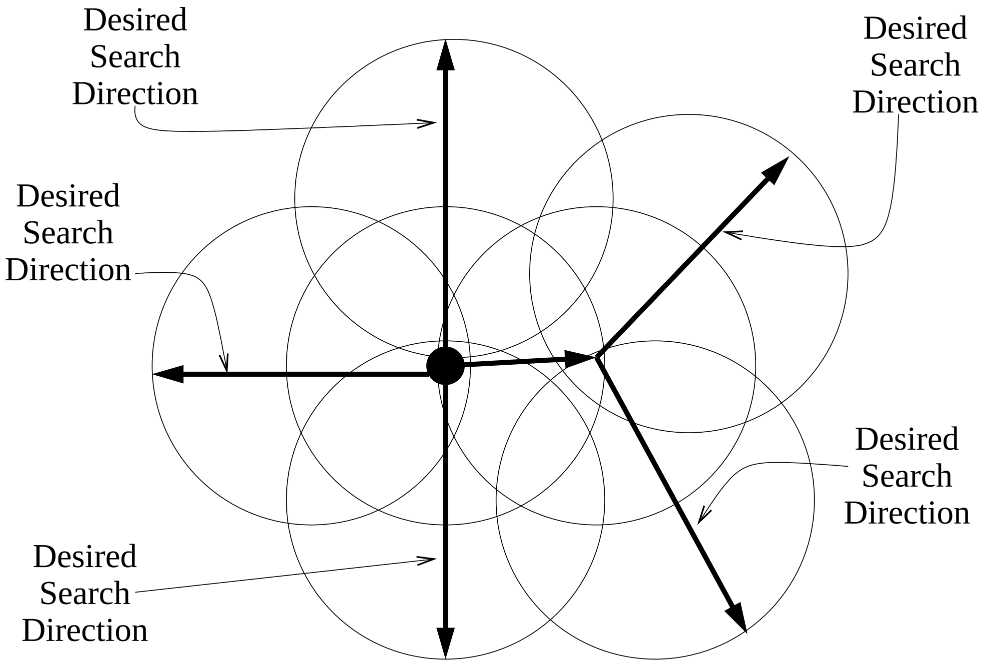


Fig. 4. Guiding the search in desirable directions.

A node can determine local query coverage by noting the addresses of neighboring nodes that have forwarded the query. In the case of multiple channel networks, a node can only detect query packets that have been directly forwarded to it. For single channel networks, a node may be able to detect any query packet forwarded within the node's radio range (i.e., through eavesdropping in promiscuous reception mode). When a node identifies a query forwarding neighbor, all known members of that neighbor's routing zone (i.e., those members which belong to both the node's and neighbor's routing zones) are marked as covered.

When a node is called upon to relay a bordercast message, it again uses its routing zone topology to construct a bordercast tree, that is rooted at itself and spans its uncovered peripheral nodes. The message is then forwarded to those neighbors in the bordercast tree. By virtue of the fact that this node has forwarded the query, all of its routing zone members are marked as covered. Therefore, a bordercasting node will not forward a query more than once.

Query detection can be enhanced by introducing a random delay prior to construction of the bordercast tree. During this time, the waiting node benefits from the opportunity to detect the added query coverage from other bordercasting neighbors. This, in turn, promotes a more thorough pruning of the bordercast tree. Increasing the average delay can significantly improve performance, up to a point. Once the bordercast delays are sufficiently spread out, further increases in delay have only a negligible impact on query efficiency.

The use of random delays does not necessarily result in extra route discovery delay. Many route discovery protocols use random pre-transmission jitter to dilute the "instantaneous" channel load of neighboring query retransmissions. This forwarding jitter may be scheduled any time between query packet reception and query packet retransmission, including just prior to bordercast tree construction.

Given the implementation of an underlying bordercast service, the operation of Zone Routing's global reactive Interzone Routing Protocol (IERP) is quite similar to standard route discovery protocols. An IERP route discovery is initiated when no route is locally available to the destination of an outgoing data packet. The source generates a route query packet, which is uniquely identified by a combination of the source node's address and request number. The query is then relayed to a subset of neighbors as determined by the bordercast algorithm. Upon receipt of a route query packet, a node records its ID in the route query packet. The sequence of recorded node IDs specifies an accumulated route from the source to the current node. If a valid route for the destination is not known (i.e., the destination is not in the node's routing zone and an active route does not appear in the node's route cache) then the node bordercasts the query. This process continues until the query reaches a node that has a valid route to the destination or until the query reaches the destination itself. In that case, a route reply is sent back to the source, along the path specified by reversing the accumulated route. The operation of the IERP is sufficiently general, so that many existing reactive protocols can be used as an IERP with minimal modification [12].

VII. Motivation for Independent Zones

Intuition, confirmed by simulation results of Zone Routing, suggests that high mobility and/or low call rates favor smaller zone radius. And vice versa, low mobility and/or high call rates favor larger zone radius. Now consider a network where different areas of the network have different mobility and call rate patterns. Due to these differences, it may turn out that the different sections may have different *optimal* zone-radii. This motivates the development of the Zone Routing framework with independent zones capability, such that different nodes are possibly assigned different zone radii. It is quite likely that such a framework would perform better that the single zone size case, as it can be fine-tuned to the local conditions of the network. Furthermore, if the network characteristics change over time, such a framework can easily and quickly adapt to the changing conditions of the network.

According to the description of Zone Routing in the previous section, every node participating in network routing should have the same value of the zone-radius. This means that before the network becomes operational, all the nodes in the network should come to a consensus on the optimal value of the zone-radius by some extraneous means. Also, any node joining the network later or undergoing rebooting should be able to infer the correct value of zone-radius that the rest of the network is operating at.

Many applications of ad hoc networks require that the network be formed and be operational quickly and nodes be free to join and leave the system at their own will, without the need for any external configuration. In such networks, the constraint of having uniform zone radius for all the nodes may not be desirable. Having independent zones capability within the Zone Routing framework would allow nodes to dynamically, distributedly, and automatically configure their optimal zone-radii making the framework truly flexible.[2]

All these points motivate the development of Zone Routing with the capability to have independent routing zones. Such a framework would help concoct a hybrid mix of proactive and reactive components, which is just right for the specific characteristics and operational conditions of the network. The proportion of proactive and reactive components in this hybrid mix can easily be changed over time and location by changing a single parameter – the zone radius of each node. We expect that such a framework would not only reduce the routing overhead, but would be responsive to the needs of the network traffic as well.

VIII. IZR Introduction

In the Independent Zone Routing (IZR) framework, different nodes may have different sized “routing zones.” What does it mean for the nodes to have independent routing zones and how does such a routing protocol operate? Before exploring these issues, we begin by re-defining some terms in the IZR context.

- *Routing Zone or Receive Zone:* The *neighborhood* around each node about which a node proactively maintains routing information is called its routing zone. A node maintains this information by receiving proactive updates from these nodes, hence this zone is also called its receive zone. This *neighborhood* consists of the set of all nodes, whose minimum distance, in hops, from the node is not more than the *zone radius*, R.
- *Send Zone:* All the nodes which require proactive updates from the node in question in order to maintain their intra-zone routing information belong to the node's send zone. A node is expected to broadcast proactive updates to the members of its send zone.
- *Peripheral Nodes:* The farthest members of a node's routing zone, whose minimum distance from the node is R hops are called its peripheral nodes.

We have seen that in the case of equally sized routing zones, a node broadcasts proactive routing information to all the members of its zone and also receives the same from each one of them. Thus, the send zone of a node is the same as its receive zone, when all nodes have equal zone-radius. However, when the nodes in the network are allowed to have independent routing zones, this may not be the case.

[2] We will see later how a node distributedly determines the optimal value of its zone-radius based on its measurements of the routing control traffic.

In IZR, the routing zone or the receive zone is also *regular* in shape – that is, it can be represented by a circle of radius proportional to the zone-radius of the node. All nodes with lesser number of hops from the node lie inside this circle and the peripheral nodes lie on the circle. In contrast, in IZR, the send zones may not have such a regular shape. The members of the send zone of a particular node S consists of all nodes of which S is a routing zone member (so that they expect to receive a routing update from S). With possibly different routing zone sizes for different nodes, S may be a member of routing zones of nodes which may possibly form an irregularly-shaped area. It is to be noted that the send zone may not even be a connected (contiguous) area.

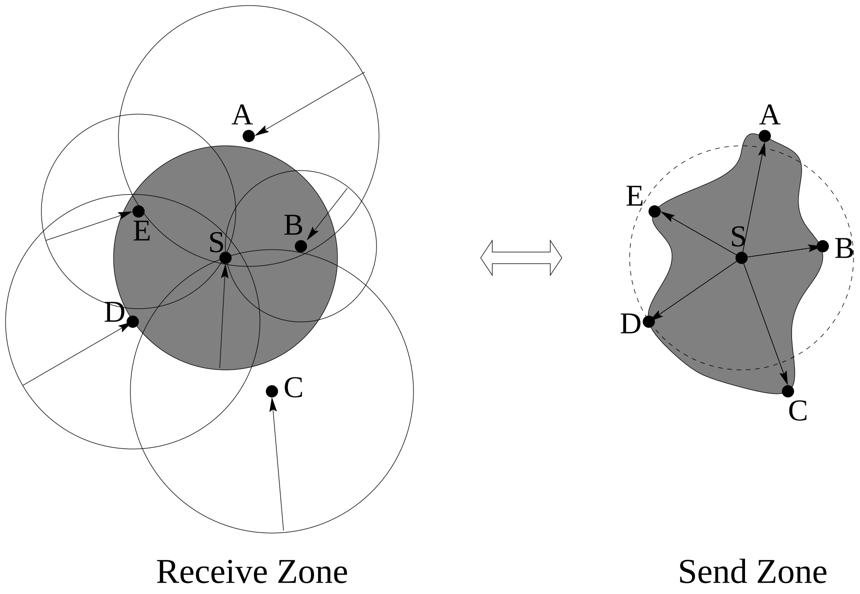


Fig. 5. The receive zone is regular in shape but the send zone may not be.

Figure 5 shows the routing or receive zones of nodes S, A, B, C, D, and E, which are regular in shape. As S is a routing zone member of A, B, C, D, and E, they belong to its send zone, which is irregular in shape.

IX. IZR Details

The basic operation of IZR is similar to Zone Routing as discussed before. If a source node has a packet to send to a destination node which is not a member of its routing zone, it bordercasts a route query packet. However, due to the presence of unequal routing zones in the network, a somewhat different bordercasting scheme is used. As unequal routing zones imply that the send zone of a node may be irregular in shape, the Intrazone Routing Protocol (IARP) has to be modified in order to distribute the proactive updates in such a send zone. Below, we discuss the changes needed in IARP and BRP in order to accommodate independently sized routing zones in the network.

A. *Intrazone Routing Protocol (IARP)*

Each node maintains proactive routing information about the members of its routing or receive zone. For this to happen, each node needs to broadcast its proactive updates to the members of its send zone. As the send zone may be irregular in terms of the distance in hops to the "boundary" nodes, a node first needs to infer the size and shape of its send zone.

Consider a scenario where each node broadcasts "zone building packets" to all the members of its *routing zone*. As the routing zones are regular in shape, this can easily be done by setting the time-to-live (TTL) field of the update packet equal to the zone-radius R. The value in the TTL field is decremented by one, each time the packet travels one hop. If the TTL value reaches zero, the packet is dropped, else it is rebroadcasted. In Figure 5, nodes A, B, C, D, E, and S broadcast their zone building packets to the members of their routing zones (marked by a circle around each of them). Thus, each node will receive a zone building packet from all those nodes to whose routing zone it belongs. In particular, S would receive A, B, C, D, and E's zone building packets as it is a member of each of their routing zones. Note that all these nodes whose zone building packets are received by a node will belong to that node's send zone – A, B, C, D, and E belong to S's send zone, as in Figure 5.

Based on the above discussion, in the general case of independent zone radius, a node can find out the size and extent of its send zone. The following scheme is used by the nodes to distribute the proactive updates in their send zones. Along with the zone-radius field and the dynamic time-to-live TTL field, an update packet also has a field which contains the initial value of TTL at the source, the TTL_0 field. The source node sets the value of the TTL and TTL_0 fields equal to the distance in hops to the farthest member of its send zone.

Initializing the TTL field to the above value make the updates reachable to all the members of a node's send zone. For example, in Figure 6, B sets the TTL (and TTL_0) field equal to the distance in hops to one of the farthest members of its send zone (node C, E or G). However, this may lead to some extra overhead, due to the updates being broadcasted in the area marked by the horizontal lines in the figure (which lies outside B's send zone). In order to reduce this overhead, each of the peripheral nodes of B maintains information about members of B's send zone which lie further away from B than itself. That is, A maintains information about C, D about E, and F about G. B's other peripheral nodes, H, J, and K do not have any such nodes. Hence, when A, D, and F receive B's proactive updates, they send it towards C, E, and G respectively. When H, J, and K receive B's updates, they do not forward the update packets, thus reducing the extra overhead.

This information to reduce the overhead is maintained as follows. A node A maintains a list of all nodes for whom it serves as a peripheral node. For each node B in this list, A maintains another list called the *expecting_nodes_list*, which consists of all nodes C, whose "zone building packets" are received by A such that the value of TTL in the packet is not less than the zone radius of B. (This implies that B lies in C's routing zone, or equivalently, C lies in B's send zone.)

Now a peripheral node A of a node B does not forward a proactive update packet originated at B, if A has no nodes in the *expecting_nodes_list* for node B. This reduces unnecessary traffic going beyond the peripheral nodes, if there are no nodes in that region which have B in its routing zone. Note that all these conditions can be checked by using the TTL, TTL_0, and the zone radius values available in the update packets or the zone building packets.

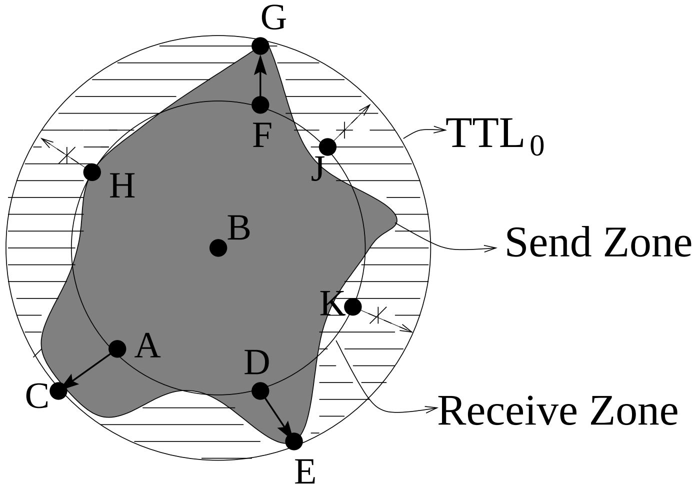


Fig. 6. The irregular send zone of node B, which is marked by the grayed area.

Using the above scheme, each node in the network broadcasts proactive update packets by initializing the values of TTL and TTL_0 as above. Propagation of unnecessary update packets may be terminated by the peripheral nodes, if no nodes beyond them are *expecting* these packets, as found by examining the maintained lists.

B. Bordercast Resolution Protocol (BRP)

With independently sized routing zones in the network, it is possible that some of the nodes in the bordercast tree of the source node have a routing zone which is small, so that it lies completely within the source node's routing zone. Such nodes may not be able to reconstruct the source node's bordercast tree and may not be able to correctly judge who to forward the packets to. In order to deal with situations like these, a different bordercasting mechanism is used.

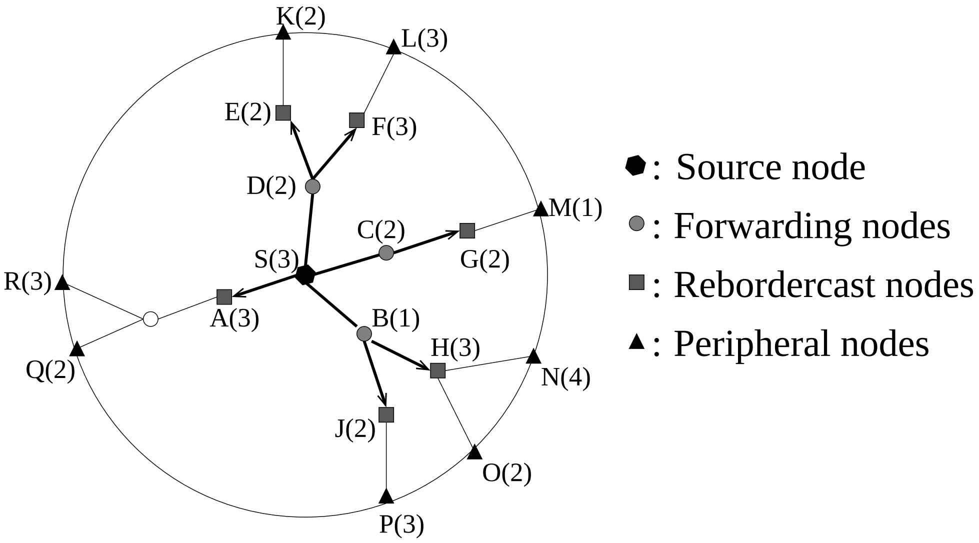


Fig. 7. The bordercast tree of the source node S. Numbers in parentheses next to the node labels indicate the zone radii of the nodes.

Figure 7 shows the bordercast tree of the source node S, which has a zone radius of 3. Nodes A, B, C, and D are the bordercast tree neighbors of S. The zone radius of A is 3 and its zone extends beyond S's routing zone. However, as the zone radii of B, C, and D are small (1, 2, and 2, respectively), their routing zones lie completely inside S's routing zone. So, S examines the nodes which are two hops from it in the bordercast tree downstream from B, C, and D. It finds that the routing zones of E, F, G, H, and J include newer regions outside its own routing zone, and so they get selected. S then sends the route query packet to A, E, F, G, H, and J (via forwarding through one-hop neighbors, if needed), which could, in turn, query unexplored regions of the network.

Formally, we define the following two kinds of nodes, which a bordercasting node identifies after constructing the bordercast tree to its *uncovered* peripheral nodes.

• *Rebordercast Node:* The node closest to the source node on the bordercast path from the source node to a peripheral node, such that its routing zone extends beyond the source node's routing zone, is called a rebordercast node of the source node corresponding to that peripheral node. For example, in Figure 7, H is a rebordercast node corresponding to O and N, J is a rebodercast node corresponding to P etc.

• *Forwarding Node:* Nodes lying on the bordercast-path between the source node and a rebordercast node belong to the set of forwarding nodes corresponding to that rebordercast node. For example, B is a forwarding node corresponding to J and H, while A does not have any forwarding node. Note that it is possible that this set may be empty.[3]

The following bordercasting mechanism is used by the nodes in order to guide a route query "outwards", towards unexplored regions of the network.

1. Source node S constructs the bordercast tree to *uncovered* peripheral nodes.
2. S chooses rebordercast nodes corresponding to each of its *uncovered* peripheral nodes.
3. S then sends the query packet to each of these rebordercast nodes via the forwarding nodes, if any.
4. When the rebordercast nodes receive the query packet, they become the bordercast nodes and go back to step 1.

For query control, a node marks certain members of its zone as *covered* and tries to steer the query away from such nodes. The following rules are used by a node for identifying such covered regions of its routing zone:

• A *forwarding node* marks all the members of its zone as covered.

• A *rebordercast node* marks

– the nodes lying in the intersection of its zone with the zone of the bordercasting node as covered, if the bordercasting node is a member of its zone.[4]

– the nodes lying in the intersection of its zone with the zone of the last forwarding node as covered, if the bordercasting node does not lie in its zone.[5]

The above mechanism ensures that the query always gets bordercasted by nodes whose routing zones covers newer, unexplored regions.

[3] The routing zone of a forwarding node does not extend beyond that of the source node, otherwise it itself would be a rebordercast node.

[4] As the bordercasting node is a member of the rebordercasting node's routing zone, the rebordercasting node knows the bordercasting node's position relative to the other members of its routing zone and thus can infer the intersection of their routing zones.

[5] The last forwarding node will be a member of the rebordercasting node's routing zone and thus it has the required information to mark the intersection of their routing zones as covered.

C. Zone Radius Determination Algorithm

In order to determine the optimal zone radius of each node in the network independently, the algorithm for zone radius determination should depend only on the local measurements made at the node. A hybrid of *Min Searching* and *Traffic Adaptive Estimation* schemes described in [22] is used to determine the optimal zone radius of each node dynamically and independently.

The Min Searching scheme involves iteratively searching for the minima of the routing control traffic curve by incrementally increasing or decreasing the routing zone radius of a node by one hop. During each estimation interval, the amount of routing traffic is measured. If the amount of routing traffic in the current estimation interval is less than that in the previous interval, the zone radius is further incremented/decremented in the same direction. Else, the direction of zone radius change is reversed. The process continues until a minimum is detected, as shown in Figure 8. The Min Searching scheme converges to the local minima, provided that the network characteristics do not change substantially during the search and that the estimation interval is long enough to provide a good measurement of the routing control traffic.

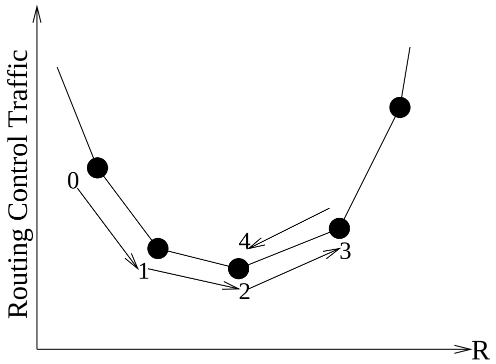


Fig. 8. Min Searching to determine the minima of the routing control traffic curve. The algorithm starts at t=0 and converges to the optimal value at t=4.

The optimal zone radius of a node lies in a region where the total routing control traffic is neither predominantly reactive nor predominantly proactive. The Adaptive Traffic Estimation scheme exploits the fact that close to the minimum the amount of these two traffic types is relatively equal. The scheme involves iteratively changing the routing zone radius based on the relative proportions of reactive and proactive components in the total routing overhead generated in each estimation interval. Let $\Gamma(R)$ be the ratio of reactive (IERP) traffic to proactive (IARP) traffic at zone radius R during a certain estimation interval. Adjustments to the zone radius is made by comparing this ratio with a predetermined threshold, Γ_{thres}. If $\Gamma(R) > \Gamma_{thres}$, increase the zone radius; if $\Gamma(R) < \Gamma_{thres}$, decrease the zone radius. However, possibly changing the zone radius after each estimation interval could lead to too frequent adaptation of the zone radius and to possible network instability. Hence, a triggering mechanism is introduced by a hysteresis term, so that if $\Gamma(R) > \Gamma_{thres} \cdot H$, increase the zone radius; if $\Gamma(R) < \Gamma_{thres}/H$, decrease the zone radius. In this scheme, the decision to change the zone radius is based on the measurements made in the current estimation interval only. This is desirable as the scheme can better track the changes in the network and the dependence on the need for correlation between successive intervals is reduced.

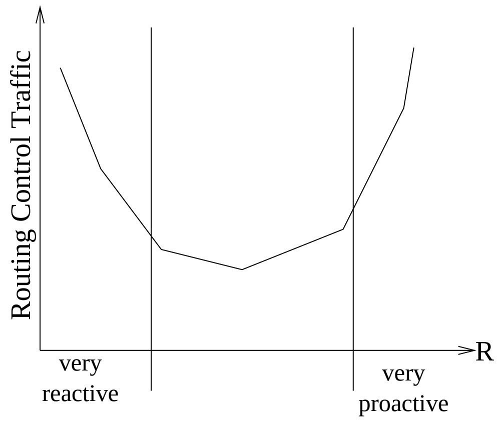


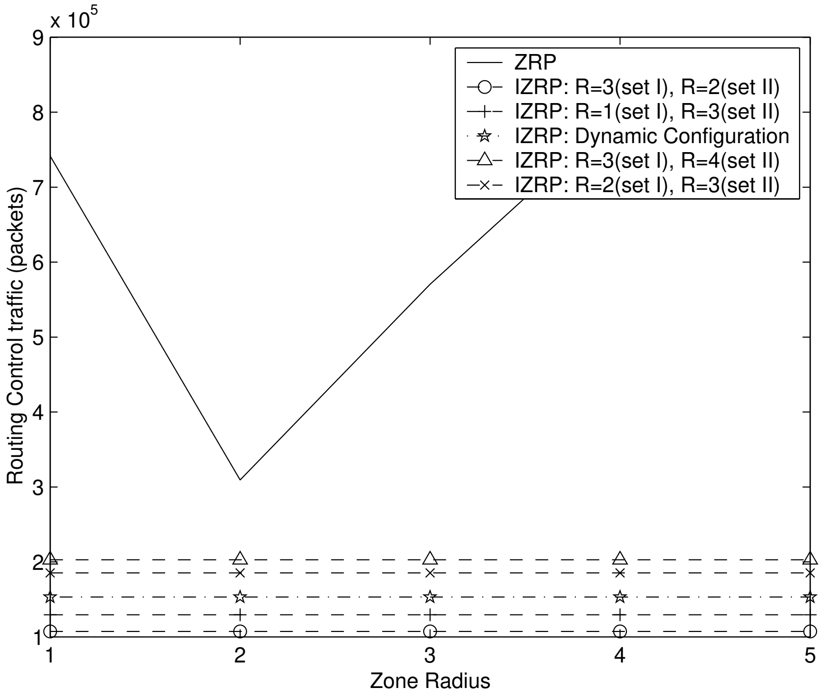
Fig. 9. Location of optimal zone radius is the region where the routing control traffic is neither very reactive or very proactive.

A hybrid zone radius determination scheme is used in IZR, where control can switch between the Min Searching and the Traffic Adaptive Estimation schemes. Initially the Min Searching scheme is under control. The Min Searching scheme provides useful statistics, which could be used to improve the estimates of Γ_{thres} and H for the Traffic Adaptive Estimation scheme [22]. Once the minimum of the control traffic curve is reached, the control switches to the Traffic Adaptive Estimation scheme. Also, whenever the zone radius R reaches one, the Min Searching scheme assumes control again [22]. This is because the Traffic Adaptive Estimation scheme could be misled by the highly reactive traffic at zone radius of one ($\Gamma(1) = \infty$).

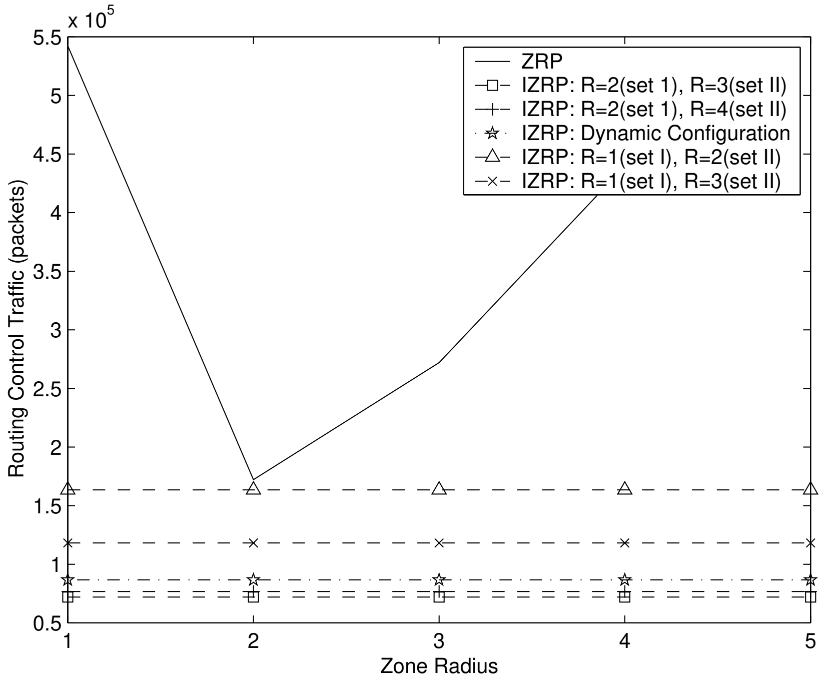
X. Performance Evaluation

The OPNET simulation environment was used to simulate the Independent Zone Routing framework. Link-state based IARP, described in [13] was used as the proactive component and a source-route based IERP described in [12] was used as the reactive component. The network consists of 50 nodes, spread randomly in an area of 1000×1000 $meter^2$. The transmission radius of the nodes is set at 225 $meters$. A node moves at a constant speed v and is assigned an initial direction θ, which is uniformly distributed between 0 and 2π. When a node reaches an edge of a simulation region, it is reflected back into the network. A node's session with a randomly chosen destination consists of sending an average of 25 packets. The interarrival time between sessions are exponentially distributed. As different simulation runs were performed for different zone radius settings, the network behavior was made to remain exactly the same; i.e., the nodes move in exactly the same path and start sessions with exactly the same nodes at the very same instants.

Figure 10 (a) shows the amount of routing control traffic generated during a simulation duration of 180 seconds. The scenario consists of 50 nodes, half of which (Set I) move at a constant speed (v) of 14 m/sec and have a mean session interarrival delay (MSID) of 1 sec. The other half (Set II) move at a speed of 1 m/sec and have a mean session interarrival delay of 0.1 sec. From the plot, it can be seen that Independent Zone Routing with dynamic radius configuration leads to about 50% reduction in routing control traffic



(a)



(b)

Fig. 10. Total routing control traffic generated in the network consisting of 50 nodes divided into two sets (a)Set I: $v = 14\ m/sec$, MSID = $1\ sec$ and Set II: $v = 1\ m/sec$, MSID = $0.1\ sec$; (b)Set I: $v = 7\ m/sec$, MSID = $1\ sec$ and Set II: $v = 1\ m/sec$, MSID = $0.1\ sec$

as compared to regular Zone Routing. The plot also shows the amount of routing control traffic for IZR with fixed but different zone radius (R) assignments for the two sets of nodes. The reduction in control traffic for this case reinforces our belief that different zone radius may be preferable for nodes with different characteristics.

The curves in Figure 10 (b) show the amount of routing control traffic for a scenario consisting of 50 nodes. Half of these nodes (set I) move at a speed of 7 m/sec and have a mean session interarrival delay of 1 sec. The rest (set II) move at a speed of 1 m/sec and have a mean session interarrival delay of 0.1 sec. The plot shows the reduction in routing control traffic as compared to Zone Routing, when IZR with dynamic zone radius configuration or IZR with fixed but different zone radii for the two sets of nodes are used.

In both Figures 10 (a) and (b), the values for IZR with fixed but different zone radii are the representative combinations of the zone radii for the two sets of nodes. The routing control traffic generated for IZR with dynamic zone radius configuration is between the different values for IZR with fixed but different zone radii combinations. The reason for this behavior is that the dynamic zone radius configuration algorithm tires to estimate the state of the network based on the traffic measurements in an estimation interval. Making the estimation interval small makes the traffic estimates during that interval inaccurate, whereas making it large makes the algorithm slow in adaptively tracking the changing conditions of the network. Thus choosing a value of the estimation interval implies a tradeoff between the accuracy of the zone radius determination scheme and the speed with which it adapts to the changing network conditions.

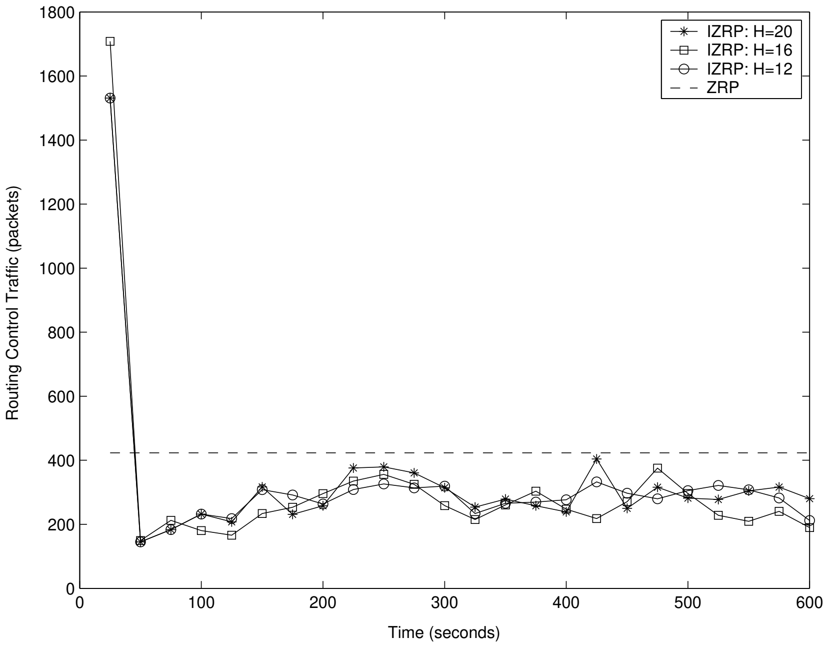


Fig. 11. The variation in routing control traffic for IZR with dynamic zone radius configuration. The points correspond to routing overhead generated over a window of last 25 seconds.

Figure 11 shows how the routing control traffic generated in the network varies as time progresses. The points on the curves represent the amount of control traffic generated over a window of last 25 seconds. Initially, the amount of traffic generated is high, as the network is operating at sub-optimal routing zone values. Soon, the zone radius determination algorithm is able to find optimal values of zone radii for the nodes and the routing overhead decreases to below the level for regular Zone Routing. The routing control traffic values for the case of IZR with dynamic zone radius configuration in Figures 10 (a) and (b) do not include this initial overhead of the scheme during its stabilization . Γ_{thres} has been set equal to 1 (corresponding to equal proactive and reactive components) for these curves. The curves for different hysteresis (H) values show that the dependence on H is not very strong.

Thus, we have demonstrated that IZR enhances the Zone Routing framework by enabling each node in the network to independently and adaptively configure its optimal zone radius. Furthermore, it can lead to reduction in routing control traffic as well, as observed from the simulation results. IZR enables setting the zone radius of each of the nodes to its optimal value over time and space. This can improve the efficiency and increase the scalability of the routing protocol.

XI. Conclusions

Hybridization, multi-scope operation, and local tuning form the basis for scalable, adaptable routing, as demonstrated by the Zone Routing framework. Zone Routing provides a flexible solution to the challenge of discovering and maintaining routes in a wide variety of ad hoc networking environments. The proportions of proactive and reactive components can be changed depending on the network conditions. With independent zones capability, Zone Routing can be fine tuned to the local conditions of the network. Each of the nodes in the network can dynamically, distributedly, and automatically configure its zone radii to the temporally and locally optimal value. This configuration is done at each node by analyzing just the local route control traffic, making the tuning mechanism itself scalable. All these factors lead to significant performance improvements and increase the scalability and robustness of the routing protocol.

Possible future directions consist of extending the hybrid routing framework into additional dimensions, for example balancing the tradeoffs between bandwidth efficiency and local processing/storage requirements. Hybrid frameworks for other layer 3+ global operations, like security and database management, can also be developed.

References

[1] Bellur, B., and Ogier, R.G., "A Reliable, Efficient Topology Broadcast Protocol for Dynamic Networks," *IEEE INFOCOM,* March 1999.
[2] Bertsekas, D., and Gallager, R., *Data Networks*, Second Edition, Prentice Hall, Inc., 1992.
[3] Chen, T.-W., and Gerla, M., "Global State Routing: A New Routing Scheme for Ad-hoc Wireless Networks," *IEEE ICC'98*, Atlanta, GA, Jun. 1998.
[4] Cheng, C., Reley, R., Kumar, S.P.R., and Garcia-Luna-Aceves, J.J., "A Loop-Free Extended Bellman-Ford Routing Protocol without Bouncing Effect," *ACM Computer Communications Review*, vol. 19, no. 4, 1989, pp. 224-236.
[5] Chiang, C.-C., "Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel," *IEEE SICON'97*, Apr.1997.
[6] Clausen, T., Jacquet, P., Laouiti, A., Minet, P., Muhlethaler, P., Qayyum, A., Viennot, L., "Optimized Link State Routing Protocol (OLSR)," *IETF MANET,* Internet Draft, October 2001.
[7] Garcia-Luna-Aceves, J.J., "Loop-Free Routing Using Diffusing Computations," *IEEE/ACM Transactions on Networking*, vol. 1, no. 1, February 1993, pp. 130-141.
[8] Garcia-Luna-Aceves, J.J., and Spohn, M., "Efficient Routing in Packet-Radio Networks Using Link-State Information," *IEEE WCNC 99*, August 1999.

[9] Garcia-Luna-Aceves, J.J., and Spohn, M., "Scalable Link-State Internet Routing," *IEEE International Conference on Network Protocols (ICNP 98)*, Austin, Texas, October 14-16, 1998.

[10] Gerla, M., Hong,X., Pei, G., "Landmark Routing for Large Ad Hoc Wireless Networks," *IEEE GLOBECOM 2000*, San Francisco, CA, Nov. 2000.

[11] Haas, Z.J., and Pearlman, M.R., "The Performance of Query Control Schemes for the Zone Routing Protocol," *ACM/IEEE Transactions on Networking*, vol. 9, no. 4, pp. 427-438, August 2001.

[12] Haas, Z.J., Pearlman, M.R., and Samar, P., "Interzone Routing Protocol (IERP)," *IETF MANET,* Internet Draft, June 2001.

[13] Haas, Z.J., Pearlman, M.R., and Samar, P., "Intrazone Routing Protocol (IARP)," *IETF MANET,* Internet Draft, June 2001.

[14] Haas, Z.J., Pearlman, M.R., and Samar, P., "Bordercast Resolution Protocol (BRP)," *IETF MANET,* Internet Draft, June 2001.

[15] Iwata, A., Chiang, C.-C., Pei, G., Gerla, M., and Chen, T.-W., "Scalable Routing Strategies for Ad Hoc Wireless Networks," *IEEE JSAC, special issue on Ad-Hoc Networks*, vol. 17, no.8, August 1999.

[16] Johnson, D.B., and Maltz, D.A., "Dynamic Source Routing in Ad hoc Wireless Networking," in *Mobile Computing*, T. Imielinski and H. Korth, editors, Kluwer Academic Publishing, 1996.

[17] Liang, B., and Haas, Z.J., "Hybrid Routing in Ad Hoc Networks with a Dynamic Virtual Backbone", to appear in *Transactions on Wireless Communications.*

[18] McDonald, A.B., and Znati, T., "Predicting Node Proximity in Ad-Hoc Networks: A Least Overhead Adaptive Model for electing Stable Routes," *MobiHoc 2000*, August 4th, Boston.

[19] Murthy, S. , and Garcia-Luna-Aceves, J.J., "A Routing Protocol for Packet Radio Networks," Proc. of ACM Mobile Computing and Networking Conference, *MOBICOM'95*, Nov. 14-15, 1995.

[20] Murthy, S., and Garcia-Luna-Aceves, J.J., "An Efficient Routing Protocol for Wireless Networks," *MONET*, vol.1, no.2, pp.183-197, October 1996.

[21] Park, V.D., and Corson, M.S., "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," *IEEE INFOCOM '97*, Kobe, Japan, 1997.

[22] Pearlman, M.R., and Haas, Z.J., "Determining the Optimal Configuration of for the Zone Routing Protocol," *IEEE JSAC, special issue on Ad-Hoc Networks*, vol. 17, no.8, August 1999.

[23] Pearlman, M.R., Haas, Z.J. and S.I. Mir, "Using Routing Zones to Support Route Maintenance in Ad Hoc Networks," *IEEE WCNC 2000*, Chicago, IL, Sept. 2000.

[24] Pearlman, M.R., Haas, Z.J. and B.P. Manvell, "Discovering and Reliably Communicating over Unidirectional Links in Ad Hoc Networks," *IEEE WCNC 2000*, Chicago, IL, Sept. 2000.

[25] Pei, G., Gerla, M., and Chen, T.-W., "Fisheye State Routing: A Routing Scheme for Ad Hoc Wireless Networks," *ICC 2000*, New Orleans, LA, Jun. 2000.

[26] Pei, G., Gerla, M., Hong, X., and Chiang, C.-C., "A Wireless Hierarchical Routing Protocol with Group Mobility," *IEEE WCNC'99*, New Orleans, LA, Sep. 1999.

[27] Perkins, C.E., and Bhagwat, P., "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *ACM SIGCOMM*, vol.24, no.4, Oct. 1994, pp.234-244.

[28] Perkins, C.E. and Royer, E.M., "Ad Hoc On-Demand Distance Vector Routing," *IEEE WMCSA'99*, New Orleans, LA, Feb. 1999.

[29] Sivakumar, P., Sinha, R., Bharghavan, V., "CEDAR: A Core-Extraction Distributed Routing Algorithm," *IEEE JSAC, special issue on Ad-Hoc Networks*, vol. 17, no.8, August 1999.