Abstract

With today's state-of-the-art technology, miniature sensors and actuators can be manufactured and integrated with electronics onto extremely small footprint devices. Such devices could be embedded into different platforms, creating a highly controllable, supportive, and cooperative environment. Applications of this technology can easily be envisioned in nearly every aspect of our life: in the workplace, at home, in a supermarket, in a department store, in a car, on the battlefield, and so on. Of course, to realize this vision, a network is necessary that enables communication among these devices. Since the main characteristics of these devices are that their power supply is extremely restricted and the span of the network is limited, conventional algorithms and protocols need to be adapted to this "micro" networking environment. This position article addresses some key issues in the creation of such a micronetwork.

A Communication Infrastructure for Smart Environments: A Position Article

ZYGMUNT J. HAAS, CORNELL UNIVERSITY

Technological advances in the area of miniaturization now allow us to produce extremely small devices with volumes on the order of tens of cubic millimeters. Moreover, it is anticipated that further reduction in size will be possible in the near future [1, 2]. Small footprint devices manufactured with micro-electromechanical systems (MEMS) technology can accommodate a variety of sensing functions such as temperature, pressure, or acceleration, and actuation functions such as rotation, linear displacement, or torque. These mechanical operations can be coupled with electronic functions, which may include onboard communications means. Inclusion of communications would allow these miniature devices not only to exchange and share information among the platforms on which these devices are mounted (e.g., people, animals, rotating machinery parts), but to form a network that implements distributed execution of algorithms. Algorithms executed in such a network interact with and within the environment in which the network is embedded, allowing environment-dependent processing and information sharing. In other words, with inclusion of even a limited amount of memory and processing power on these microdevices, an ad hoc environment can be devised in which the participating entities may both contribute and receive services and information. The far-fetched vision is that of *intelligent powder*, in which these miniature and lightweight devices can be deployed instantaneously and operate autonomously in a concerted manner to carry out certain actions.

The microsensors/actuators, to which we refer in this article, possess the raw ability to transmit a signal between two relatively close devices. To create a network, one must provide the nodes with the following basic functions: a medium access control, an addressing scheme, and a routing algorithm. Of course, these are the traditional issues in the design of a communication network. What is new here is that the traditional algorithms and protocols may not be applicable to the problem at hand, mainly because of the extremely limited computing, storage, and communication resources available in the microsensor nodes. Extremely low battery power and a limited amount of nonvolatile memory lead to a requirement for very low-complexity algorithms and protocols. In other words, the name of the game is to design a system with protocols as simple as possible requiring the least communication among the network nodes. Additional characteristics of our network of miniature devices are the very large number of network nodes (in the thousands) and the ability of these nodes to move relative to the rest of the network. However, probably the most challenging of all is the requirement that the network be deployable without preplanning and continue to operate as the carrying platform moves among different environments. Thus, another design attribute is that the protocols need to be versatile enough to adapt to both different and differing operational conditions of the network.

Interestingly, the problem of creating a network and supporting connectivity in this micronetworking environment is related to a problem that has been the subject of extensive research in the past 30 years: *multihop* networks; for example, the early 1970s effort of the Defense Advanced Research Project Agency (DARPA) on packet radio network technology and the more recent GLOMO program [3]. Under the name *ad hoc networks*, research on this type of communications has recently received renewed interest. Issues related to ad hoc networks are being extensively addressed in the Internet Engineering Task Force (IETF) MANET Working Group [4]. Because of the similarities between ad hoc and microsensor networks, MANET protocols could be adapted with some changes to create such a *smart communication environment*.

This article concentrates mainly on the issue of routing in a network that will support communication among a large collection of small-sized, possibly mobile nodes, operating in an environment with continually changing conditions. Because of its very small coverage area, we term such a network a *micronetwork*.

What Is an Ad Hoc Network?

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 nodal transmission range that is typically

This work has been sponsored in part by ONR contract no. N00014-00-1-0564 and AFRL contract no. F30602-97-C-0133.

1070-9916/00/$10.00 © 2000 IEEE

small relative to the span of the network. To provide communication throughout the entire network, nodes are designed to serve as relays if needed. The result is a distributed multihop 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 communications in a variety of hostile environments. This makes ad hoc networks very appropriate for providing tactical communication for military, law enforcement, and emergency response efforts. Ad hoc networks can also play a role in civilian fora such as the electronic classroom, convention centers, and construction sites. Finally, ad hoc networks could also be used to create communications among distributed collections of devices, such as sensors or actuators. With such a broad scope of applications, it is not difficult to envision ad hoc networks operating over a wide range of coverage areas, nodal densities, and nodal 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 data among the network nodes. This contradicts the fact that all updates in a wireless communications environment travel over the air and are costly in resources, especially with energy-limited devices, as in our micronetworking environment.

Existing routing protocols can be classified as either *proactive* or *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 used immediately. In contrast, reactive protocols invoke a route determination procedure on demand. Reactive route discovery is usually based on a query-reply exchange, where the route query is flooded through the network to reach the desired destination. The advantage of proactive schemes is that route information is readily available when needed, resulting in little delay prior to data transmission. In contrast, reactive schemes may produce significant delay in order to determine a route when routing information is needed but not available.

Routing schemes, whether proactive or reactive, require some exchange of control traffic. This overhead can be quite large in ad hoc networks if the topology frequently changes. Reactive protocols produce a large amount of traffic by effectively flooding the entire network with route queries. The combination of excessive control traffic and long route query response time rules out pure reactive routing protocols for real-time communications applications. Pure proactive schemes are likewise inappropriate for ad hoc networks, since they continuously use a large portion of the network capacity to keep the routing information current. Proactive protocols tend to distribute topological changes widely in the network, even though the creation/destruction of a new link at one end of the network may not be a significant piece of information at the other end of the network. Furthermore, since ad hoc network nodes may move quite fast, and changes may be more frequent than route requests, most of this maintained routing information is never used! This results in unacceptable waste of network capacity. In our approach, which we term the *Zone Routing Protocol*, we create a hybrid proactive-reactive scheme in which the route discovery process is both fast and efficient in the amount of control traffic.

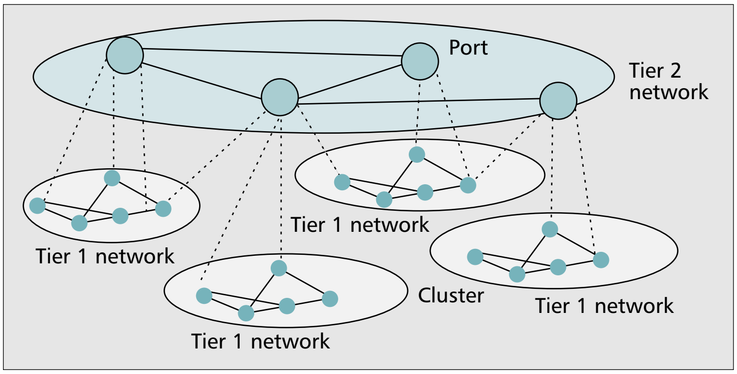


Figure 1. *A two-tier hierarchical network architecture.*

The Sensor Network Architecture

The overall network architecture discussed here is that of a two-tiered network, as shown in Fig. 1. The lower tier, tier 1, is composed of clusters of sensor devices, which can communicate among themselves in a peer-to-peer manner. In addition, the sensor devices can communicate with larger transceivers, which we call *ports*. The main function of a port is to provide an interface between a "cloud" of miniature sensor devices — the tier 1 network — and an external network, or among clouds of sensor networks. That is, a port interconnects two tier 1 networks if they cannot be directly connected due to transmission power limitations. A tier 1 node may not be able to communicate with any port, or may be in communication with a single or multiple port. Ports communicate among themselves by creating the tier 2 network, which can interface to an external network, with some of the ports serving as gateways.

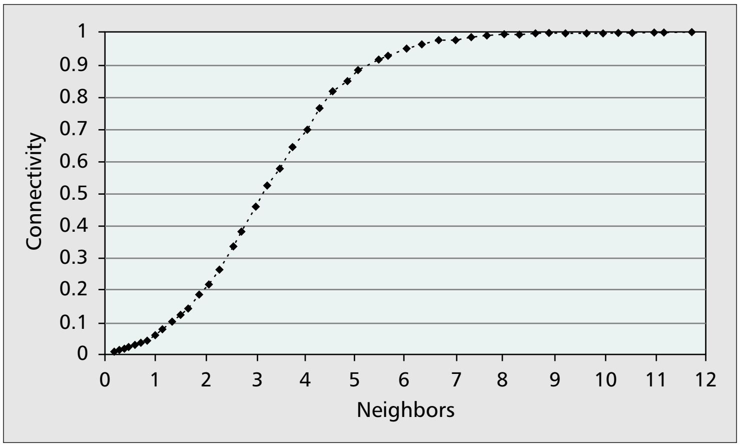
Because it is assumed that tier 2 nodes are not battery power limited, the tier 2 network uses "conventional" wireless transmitter/receiver design. In contrast, tier 1 networks are transmission power restricted. Furthermore, communication among tier 1 nodes may also be based on passive communication, such as reflection of a modulated radio frequency (RF) carrier. Thus, the tier 1 nodes communicate through multihop routing. Consequently, the issues related to the design of the network protocols in the two tiers are quite different. Here, we focus on the more challenging problem of communication within the tier 1 network; the first issue we discuss is the nodal addressing scheme.

Hardware-Based Addressing

Traditional addressing is done by assigning an address to every network node from a *logical* address space. However, logical addressing requires a fair amount of processing. An alternative is to use *physical addressing*, which could be implemented by using some characteristic of the transmitted signal that will allow selection of the intended receiver. As one example, we have considered associating each node with a particular offset from the central frequency of the transmitted signal; this offset identifies the node. As it is possible to build very sharp miniaturized filters [5, 6], it is possible to create a large address space of thousands of nodes. Thus, to communicate between adjacent nodes, the source device needs to tune to the particular offset of its neighbor (i.e., transmit to a node i); the transmitter tunes to frequency f_i, where

$$f_i = f_c - \left(\left\lfloor {}^{n}\!/_{2} \right\rfloor - i \right) \cdot f_o; \qquad 0 \le i \le n,$$

f_c is the central frequency, f_o is the frequency offset between adjacent addresses (which depends on the selectivity of the RF filter), and n is the size of the address space. Of course,



■ **Figure 2.** *Connectivity vs. number of neighbors.*

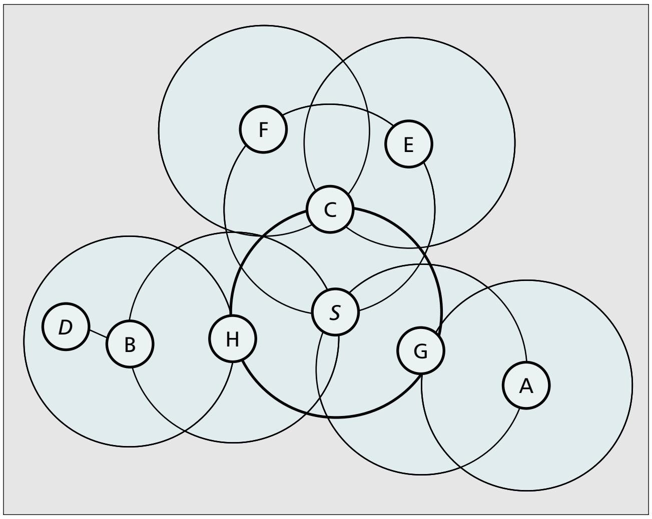
this is just one example of physical addressing, and there are many other possibilities as well.

Once the address scheme is in place, the next step in creating a network is for each node to learn about the existence of the other nodes around it.

On-Demand Neighbor Discovery

In wireless packet data networks, neighbor discovery is usually done by some sort of a *beaconing* process, in which the nodes listen to periodically transmitted signals from their neighbors. Such an operation is rather wasteful of transmission power. A number of alternative solutions to the periodic beaconing process are possible. For instance, one approach would be to perform *on-demand* neighbor discovery by a simple query, which is initiated by a node that needs to learn who its neighbors are. Of course, the design of such a neighbor discovery process has to be tightly coupled with the addressing scheme used.

Another issue that needs to be addressed in this context is the number of neighbors a node is designed to see. The number of neighbors is controlled by the transmission power of the nodes. On one hand, a large number of neighbors is advantageous, since there is a smaller chance of losing network connectivity (i.e., creating a network partition). However, many neighbors also lead to larger transmission power of a node because of, for example, the neighbor discover beaconing process. Additionally, it also results in a larger number of transmissions, and thus more transmission power. Furthermore, it also reduces the network throughput due to more contention among the accessing nodes. Our study of the on-demand neighbor discovery protocol suggests that five to eight neighbors correspond to 90, 95, 98, and nearly 100 percent connectivity, respectively (Fig. 2). The actual choice of the number of neighbors is dependent on factors such as the transmission power capabilities of the sensor devices, the addressing and medium access control schemes used, and the required richness of routes between source and destination devices.



■ **Figure 3.** *An example of ZRP operation.*

Once the communication between adjacent nodes is implemented, an end-to-end network routing scheme is required, since the distance between the two communicating nodes is typically much larger than the transmission range of a single node. We propose to use a simple routing scheme, the Zone Routing Protocol (ZRP), which allows discovery of routes within the network of a large number of microsensor nodes with little communication overhead and minimal nodal storage requirements.

The Zone Routing Protocol: A Short Description

The wired Internet uses routing protocols based on topological broadcast, such as Open Shortest Path First (OSPF) [7]. These protocols are not suitable for the micronetworking environment due to the relatively large bandwidth required for topological update messages.

Routing in multihop packet radio networks was based in the past on shortest-path routing algorithms [8] such as the Distributed Bellman-Ford (DBF) algorithm [9]. These algorithms suffer from slow convergence rates (the "counting to infinity" problem). Besides, DBF-like algorithms incur large update message penalties. New multihop routing protocols were proposed [10–15]; however, synchronization requirements, large storage, and extra processing overhead are common in these protocols, which may make them inappropriate for the micronetworking environment.

In contrast, our routing protocol [16] incurs very low overhead in route determination. It requires a minimal amount of routing information to be maintained in each node, and the cost in wireless resources for maintaining routing information of inactive routes is very small. The protocol identifies multiple routes to the destination (increasing reliability and performance), with no looping problems. However, the most appealing feature of the protocol is that its behavior is adaptive, based on the mobility and communication patterns of the microsensor devices. ZRP limits the propagation of such information to the neighborhood of the change only, thus limiting the cost of topological updates. This feature is of particular interest in a large network such as the microsensor network.

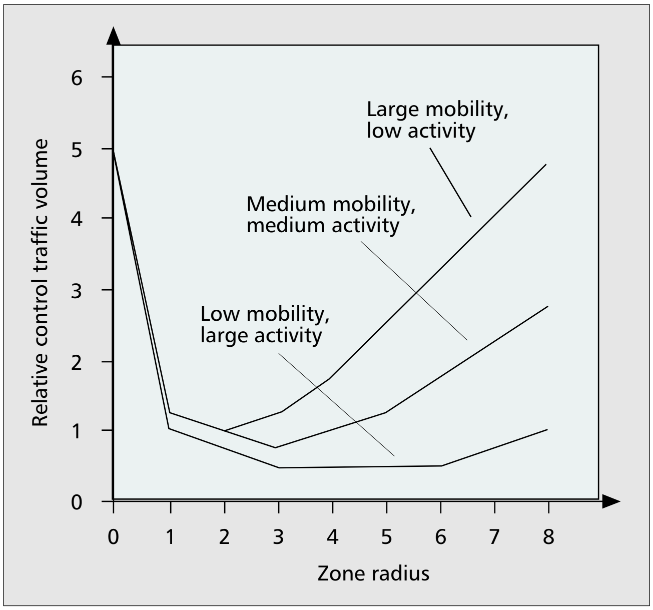
ZRP is based on the notion of a *routing zone*, which is defined for each node and includes the nodes whose distances in hops are at most some predefined number, referred to here as the *zone radius*. Each node is required to know the topology of the network within its routing zone only. Thus, nodes are updated about topological changes only within their routing zone. Therefore, in spite of the fact that a network can be quite large, the updates are only locally propagated. Since for a radius greater than one the routing zones heavily overlap, the routing tends to be extremely robust. The routes within the network are specified as a sequence of nodes, separated by approximately the zone radius. We illustrate the operation of the *route discovery* procedure by an example in Fig. 3. The source sensor S attempts to send a packet to the destination sensor node D. To find a route within the network, S first checks whether D is within its routing zone.[1] If so, S knows the route to node D. Otherwise, S sends a query to all the nodes on the periphery of its zone[2] (i.e., to nodes C, G, and H). Now, in turn, each of these nodes, after verifying that D is not in its routing microzone, forwards the query to its peripheral nodes. In particular, H sends the query to B, which recognizes D as being in its routing zone and responds to the query, indicating the forwarding path S-H-B-D. The mechanism by which B learns about the forwarding path is the simple route accumulation protocol, by which each node that forwards the query appends its identity to the query message.

Because the routing zones heavily overlap, the route query will be forwarded to many network nodes multiple times. In fact, it is very possible that the query will be forwarded to all the network nodes, effectively flooding the network. To prevent this, the query termination and query forwarding strategies used in traditional flooding algorithms need to be properly extended for use in a routing zone architecture.

In order to understand the need for a query control mechanism in ZRP, it is important to stress one of the key features of the routing zone: a node's response to a route query contains information about that node's entire routing zone. From this perspective, excess route query traffic can be regarded as a result of overlapping query threads (i.e., overlapping queried routing zones). Thus, the design objective of query control mechanisms should be to reduce the amount of route query traffic by steering threads outward from the source's routing zone and away from each other. This problem is addressed in ZRP from two different perspectives: thread overlap detection/termination and thread overlap prevention. We omit further discussion of these important mechanisms due to space limitations and refer the reader to [16, 17].

The main advantage of ZRP is the fact that the number of flood messages to discover a route is significantly reduced from those of the other reactive-type protocols. This is because the route discovery queries leap between the peripheral zone nodes in quantum of the zone radius. An important aspect of zone routing is that it discovers multiple routes to the destination. Finally, the route discovery process can be made even more efficient in resources at the expense of longer latency. This could be done by sequentially, rather than simultaneously, querying the peripheral zone nodes, either one by one or in groups. Thus, there is a trade-off between the cost and latency of the route discovery procedure.

The performance of ZRP has been evaluated as a function of the network parameters, such as the size of the zone radius, the density of the microsensors, the transmission radius of the microsensors, migration patterns of the microsensors, and the size and shape of the micronetwork coverage. A sample of ZRP performance is shown in Fig. 4. The three representative curves in the figure show the relative volume of the control traffic as a function of the size of the zone radius. For a highly mobile network with low traffic activity, a smaller zone radius results in minimal control traffic, and vice versa. For extremely high mobility, the network degenerates to flooding (zone radius of 1); for a stationary network, the whole network becomes a single zone (large zone radius). The figure clearly demonstrates the drastic reduction in the amount of control traffic from those in the purely proactive (large zone radius) and the purely reactive approaches (zone radius of 1). We omit additional performance evaluation results due to space limitations; those could be found in [16, 17].



■ **Figure 4**. *The overhead of ZRP control traffic.*

Processing vs. Communication Costs

After the network has been established, the nodes are in position to communicate. One can imagine a wide spectrum of applications being implemented on the proposed microsensor network. In particular, some applications may require processing at the sensor nodes. Since the processing capability of the nodes is quite limited, a natural possibility is to offload some of the processing from the network nodes to be processed on the ports, or even in the backbone fixed network. However, shipping of processing out of the sensor nodes that collect the information to other network components requires transmission, which is also limited, in both network capacity and energy expenditure.

Therefore, there exists a trade-off between the distribution of processing on network nodes vs. other network components. As another aspect of this trade-off, the information collected by the sensor nodes themselves can be locally processed (e.g., compressed), with less information transmitted out of the sensor nodes. Again, we see that there is a trade-off between local processing and off-node processing with transmission.

Another trade-off to consider is due to the limited storage in network nodes. Processing of some algorithms may require availability of storage. Even the routing protocol itself needs some limited storage for execution. The question is what is the minimal amount of storage needed to perform some network operations.

Thus, there is a three-dimensional space one may try to optimize: computation vs. storage vs. transmission. The challenge is to determine the permissible boundaries in this three-dimensional space, given the device and circuit technology with which the sensor nodes are designed. As the processing, storage, and transmission capabilities of network nodes increase, more complex network control protocols can be deployed. The challenge is to optimize the performance of the network protocols (e.g., network capacity and delay) subject to the amount of the above three resources in the miniaturized network nodes.

Yet another aspect worth mentioning is the trade-off between the size of a device and its computation power, storage capability, and transmission capacity. In particular, if one sizes the device to reduce its dimensions by some factor, how

[1] *Recall that each node knows all the nodes and routings within its routing zone.*

[2] *Nodes that are one zone radius away.*

does this affect the above three parameters? Although such results are very much technology-dependent, we believe they can be extrapolated over a spectrum of technologies.

Summary and Concluding Remarks

In this position article, we have introduced the vision of micronetworking: creating a network of miniature devices (sensors and actuators) which could possibly be manufactured using MEMS technology. These devices include, in addition to limited communications capabilities, also restricted memory and computing resources. Thus, the challenge is to design a set of protocols that, on one hand, would allow sufficient coupling between the devices, but, on the other hand, be extremely efficient in the use of computing and communication. In particular, we have introduced the notion of physical addressing, on-demand neighbor discovery, and the Zone Routing Protocol, which, working together, may constitute a basic set of schemes to implement the smart communication environment of the future. Our Wireless Networking Laboratory at Cornell University has been pursuing these issues in collaboration with other research groups at Cornell University.

References

[1] T. Akin, K. Najafi, and R. Bradley, "A Wireless Implantable Multichannel Digital Neural Recording Systems for a Micromachined Sieve Electrode," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 1, 1998.

[2] G. Asada *et al.*, "Wireless Integrated Network Sensors: Low Power Systems on a Chip," *Proc. 1998 Euro. Solid State Circuits Conf.*

[3] http://www.darpa.mil/ito/research/glomo/index.html

[4] http://www.cs-ipv6.lancs.ac.uk/ipv6/documents/standards/general-comms/ietf/manet/

[5] D. J. Young *et al.*, "A Low-Noise RF VCO Using On-Chip High-Q 3-D Coil Inductor and Micromachined Variable Capacitor," *Solid-State Sensor and Actuator Wksp.*, Hilton Head, SC, 1998.

[6] D. R. Pehlke *et al.*, "Extremely Hight-Q Tunable Inductor for Si-Based RF Integrated Circuit Applications," *IEDM'97 Int'l. Elect. Dev. Mtg. Tech. Digest*, 1997.

[7] J. Moy, "OSPF Version 2", RFC 1583, Mar. 1994.

[8] B.M. Leiner, D. L. Nielson, and F.A. Tobagi, "Issues in Packet Radio Network Design," *Proc. IEEE*, vol. 75, Jan. 1987, pp. 6–20.

[9] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed., Prentice Hall, 1992.

[10] C. Cheng *et al.*, "A Loop-Free Extended Bellman-Ford Routing Protocol without Bouncing Effect," *ACM Comp. Commun. Rev.*, vol. 19, no. 4, 1989, pp. 224–36.

[11] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad-Hoc Wireless Networking," *Mobile Computing*, T. Imielinski and H. Korth, Eds., Kluwer, 1996.

[12] S. Murthy and J. J. Garcia-Luna-Aceves, "A Routing Protocol for Packet Radio Networks," *Proc.ACM MOBICOM '95*, Nov. 14–15, 1995.

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

[14] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *ACM SIGCOMM*, vol. 24, no. 4, Oct. 1994, pp. 234–44.

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

[16] M. R. Pearlman and Z. J. Haas, "Determining the Optimal Configuration for the Zone Routing Protocol," *IEEE JSAC*, Special Issue on Ad-Hoc Networks, vol. 17, no. 8, Aug. 1999.

[17] Z. J. Haas and M. R. Pearlman, "The Performance of Query Control Schemes for the Zone Routing Protocol," *Ad-Hoc Networks*, C. Perkins, Ed., Addison Wesley, 2000.

Biography

ZYGMUNT J. HAAS [SM] (haas@ee.cornell.edu) received his B.Sc. and M.Sc. in electrical engineering in 1979 and 1985. In 1988, he earned his Ph.D. from Stanford University and subsequently joined AT&T Bell Laboratories in the Network Research Department. There he pursued research on wireless communications, mobility management, fast protocols, optical networks, and optical switching. From September 1994 to July 1995 he worked for the AT&T Wireless Center of Excellence, where he investigated various aspects of wireless and mobile networking, concentrating on TCP/IP networks. In August 1995 he joined the faculty of the School of Electrical and Computer Engineering at Cornell University. He is an author of numerous technical papers and holds 12 patents in the fields of high-speed networking, wireless networks, and optical switching. He has organized several workshops, delivered tutorials at major IEEE and ACM conferences, and serves as editor of several journals. He has been a guest editor of three *IEEE JSAC* issues (Gigabit Networks, Mobile Computing Networks, and Ad-Hoc Networks). He is a voting member of ACM and vice chair of the IEEE Technical Committee on Personal Communications. His interests include mobile and wireless communication and networks, personal communication service, and high-speed communication and protocols. His URL is http://www.ee.cornell.edu/~haas/wnl.html