IGF: A State-Free Robust Communication Protocol for Wireless Sensor Networks

Technical Report CS-2003-11

Brian Blum, Tian He, Sang Son, Jack Stankovic
Dept. of Computer Science
University of Virginia
Charlottesville, VA 22904
{bblum, tianhe, son, stankovic}@cs.virginia.edu

Abstract

Wireless Sensor Networks (WSNs) are being designed to solve a gamut of interesting real-world problems. Limitations on available energy and bandwidth, message loss, high rates of node failure, and communication restrictions pose challenging requirements for these systems. Beyond these inherent limitations, both the possibility of node mobility and energy conserving protocols that power down nodes introduce additional complexity to routing protocols that depend on up to date routing or neighborhood tables. Such ***state-based*** *protocols suffer excessive delay or message loss, as system dynamics require expensive upkeep of these tables. Utilizing characteristics of high node density and location awareness, we introduce IGF, a location-aware routing protocol that is robust and works without knowledge of the existence of neighboring nodes (state-free). We compare our work against established routing protocols to demonstrate the efficacy of our solution when nodes are mobile or periodically sleep to conserve energy. We show that IGF far outperforms these protocols, in some cases delivering close to 100% of the packets transmitted while alternate solutions fail to even find a path between a source and destination. Specifically, we show that our protocol demonstrates a vast improvement over prior work using metrics of delivery ratio, control overhead, and end-to-end delay.*

1 Introduction

Wireless Sensor Networks (WSNs) continue to expand the state of the art in distributed computing. Proposed for ad hoc deployment in remote or inhospitable terrain, systems are designed to abide unpredictable conditions for extensive periods of time. These WSNs – whose applications may include military surveillance, vehicle tracking, disaster relief, search and rescue, and smart environments – require energy conserving features to ensure system longevity. Miniscule devices transported by the elements (e.g. wind, water, or earth tremors) or devices annexed to troops also make dynamic topologies possible. When nodes are mobile, the constant migration of nodes into and out of the communication ranges of one another make routing state upkeep difficult. In addition, state upkeep is crucial to performance in systems that allow nodes to transition into and out sleep states to conserve energy. These state transitions therefore render state-based routing protocols for WSNs impractical, and create the demand for a solution that can deliver end-to-end traffic without knowledge of potentially invalid or dynamic routing and neighborhood tables.

State-free routing protocols are needed that do not suffer these dire affects of transitory neighborhood or system state. We define state-free as *having no dependence on knowledge of the network topology or the presence/absence of any other node, including the state of that node, at a particular time.* This characteristic of being state-free is salutary to sensor networks, as it induces fault tolerance and makes protocols robust to topology shifts or node state transitions. Further, a state-free solution can accomplish this without the bandwidth-consuming messages required in state-based solutions for routing and neighbor table upkeep. With the goals of introducing a protocol that is impervious to system-wide state transitions caused by mobility or induced node quiescence, manifesting properties of fault tolerance and robust performance, eliminating the latency and overhead of state upkeep, and minimizing message loss, we propose a novel location based routing protocol that is altogether state-free. We call our protocol **Implicit Geographic Forwarding (IGF),** as it makes non-deterministic routing decisions, implicitly allowing opportune receiving nodes to determine a packet's next-hop at transmission time. Specifically, IGF works without prior knowledge of any other node in the network[1], using an integrated Network/MAC solution to identify the best forwarding candidate at the instant a packet is sent. Aside from providing robust message delivery, increasing system stability, and reducing control overhead, we utilize the energy remaining at a node during the candidate election process to ensure nodes do not shoulder vastly unequal portions of the workload and die out sporadically.

To demonstrate IGF's superior performance, we compare our research with both classical and state of the art solutions in simulation using end-to-end delivery ratio, control overhead (which correlates to energy consumption), and communication delay as our metrics. While demonstrating equivalent or better performance under more "traditional" tests, we verify IGF's superior qualities when considering mobile or energy conserving systems. In these more likely scenarios, we show that IGF is capable of delivering close to 100% of simulated end-to-end traffic when other protocols fail to find a path from the source to the destination. We note that our protocol prevails when nodes migrate or transition into and out of energy conserving sleep states for the following reasons:

- We reduce end-to-end communication delay and decrease protocol overhead by eliminating neighborhood table and system state upkeep messages.
- We allow awakened or arriving nodes to actively participate in routing without delay.
- We allow node migration or protocol induced energy-conserving state transitions to occur irrespective of routing, saving the overhead of notifying neighbors or performing system maintenance operations.
- We make routing more robust to message loss, allowing any opportune node to handle message forwarding instead of tasking a specific node to perform this duty.
- We distribute the workload amongst nodes through energy-aware algorithms, ensuring energy depletion occurs in a uniform manner.
- We save memory by disregarding neighborhood and routing tables. This frees limited RAM for alternate uses (e.g. caching, message buffering, and aggregation).

In the remainder of this paper we discuss our protocol and provide an in depth comparison and analysis of our research. We address relevant work in Section 2. We then describe the details of IGF in Section 3. Section 4 presents our experiments and provides an analysis of the data collected. Finally we conclude in Section 5.

2 Related Work

In this section we discuss prior research in distributed computing that addresses issues of system state, robustness, routing, mobility, and energy conservation. We address protocols at all layers of the communication stack (transport, network, and MAC), to discuss both contrasting techniques and influential properties that remain pertinent to the design of IGF. In addition we look at proposed applications to better understand the system requirements that IGF supports.

Above the network layer, various protocols have been introduced to reduce message loss through reliable communication. RMST [26] and PSFQ [31], both proposed as reliable transport layer solutions, attempt to provide such reliability at a minimal cost. RMST acts as a filter in Directed Diffusion[9]; tracking fragments so that receiver initiated requests can be instantiated when individual pieces of an application payload get lost. PSFQ caches packets along the path to the sender, initiating fragment recovery as required, starting with its local neighborhood. While RMST and PSFQ have proven effective when dealing with interference and packet collisions, their robust and reliable features are not appropriate for handling network layer routing problems that result when nodes migrate or transition into and out of inactive (i.e. sleep) states as they introduce significant end-to-end delay and communication overhead.

Tolerance to node inactivity or failure has been addressed in [5][6][18][19][30]. These protocols establish groups and hierarchies that introduce redundancy through aggregation and consensus. Architectures such as [3] and [37] also mask individual node responsibilities through concepts such as loose group semantics, providing a robust and fault tolerant solution for distributed sensor networks where the state of individual nodes is unreliable. In each case, these algorithms introduce robust and fault tolerant

[1] We expect routing to locations, not individual sensing devices.

properties that mask node failure or sleep schedules above the network layer. Because these higher layer protocols do not directly deal with problems inherent to routing, we consider them complementary to our work.

At the network layer, classical routing protocols [10][20][21] have been deemed inappropriate for WSNs that must contend with issues of mobility, node failure, message loss, and bandwidth limitations. LAR [15] extends the ideas proposed by AODV [21] and DSR [10], utilizing location information to limit the scope of route requests. While LAR significantly reduces routing overhead by only propagating queries to relevant portions of the network, it still must maintain or establish an explicit path between every source and destination prior to transmitting a packet.

Geographic Forwarding [12] uses greedy forwarding to forward packets to the neighboring node that is closest to the final destination. GF began the trend in pure location based routing and has since seen extensions for handling voids [13], supporting real-time traffic [16], and coping with congestion [8]. An important contribution of GF based solutions was the removed requirement that a protocol maintain a global view of the network (i.e. end-to-end routing tables). GF therefore reduces communication overhead by eliminating its dependence on network wide state information. However, GF based solutions still depend on up to date local neighborhood tables, requiring some communication overhead to maintain and suffering latency and message loss when a node's neighborhood state changes between updates.

One alternate routing solution, designed to eliminate costly routing tables, is the Directed Diffusion protocol [9]. This work is built around interested parties (e.g., a Base Station) diffusing *interests* throughout the network. As these interests propagate, reverse paths are established, along which sources send relevant data back to the querying sink. Although Directed Diffusion reduces a node's state by eliminating end-to-end routing tables, it still relies on neighborhood tables, like GF, requiring extensive upkeep and suffering packet loss when node migration or neighbor state transitions temporarily invalidate the path to a destination.

More recent work to ensure robust data delivery [7][38] by simultaneously sending packets along multiple paths to a destination comes at the expense of significant increases in communication overhead and does not eliminate the need to establish and maintain neighbor tables locally.

The desire to extend a system's lifetime, through energy conservation and traffic dispersion, are both features and motivations of our research. With the ratio of energy consumption in idle:receiving:transmitting states being estimated at 1:1.05:1.4 [27] or 1:2:2.5 [14], depending on the hardware, transitioning into a sleep state provides the only significant means of conserving energy. Protocols focusing on powering down the radio for energy conservation have thus been proposed at all levels of the communication stack.

At the MAC layer, 802.11 power saving mode [40], PAMAS [24], S-MAC [36], and Piconet [2] have been proposed to temporarily power down the radio and conserve energy. As previously stated, these protocols' power saving features may have adverse effects when coupled with state-based routing protocols because they periodically transition nodes to sleep, introducing latency or routing failure as nodes wait for a next-hop neighbor to become active. Although our work is developed on top of a simplified 802.11 DCF MAC [40], we note that the use of additional power saving mechanisms add complexity in the form of state transitions, making our non-deterministic algorithm even more appropriate.

At the network layer, energy-conserving protocols, applicable to WSNs, include an extension to Directed Diffusion [7], [22], and SPEED [8]. While working to power down the radio or disperse traffic for distributed power consumption throughout the network, these protocols create significant communication overhead and latency when nodes freely transition into and out of dormant states. In addition, several protocols (SPAN [4], GAF [34], AFECA [33], PEAS [39]) work in conjunction with routing to rotate communication responsibilities, powering down nodes that are not needed to form a connected communication backbone. SPAN and GAF introduce additional communication overhead as nodes notify one another of state transitions. Likewise, AFECA and PEAS power down nodes that are not required for communication, failing to distribute the system workload and thereby limiting applications that may desire higher node densities for aggregation or consensus.

We note that many of these existing protocols are designed around rotating communication responsibilities and may be appropriately reconsidered concomitant with our work. However, in their current state, their inadequacies stem from the additional cost, in communication, to notify neighbors of transitions in state, and their inability to distribute energy consumption throughout the network. While the drawback of the former is obvious, uniform system energy consumption is desired to keep as many nodes alive as possible for the lifetime of the system. This uniform system depreciation is desirable, as upper layer protocols (e.g. tracking, group management) often make distributed decisions based on the participation of some threshold number of nodes. If rotation were to occur that reduced the network to a minimum routing backbone and then replaced dying nodes as appropriate, achieving sufficient node density for cumulative decision-making (e.g. consensus or aggregation) may become impossible.

Classical MAC protocols lack the ability to cope with mobility, individual node failure, or cyclical state transitions. Prior solutions [11][40] are all invoked with a pre-specified next-hop, returning control back to the routing component when the specified receiver is unavailable or fails to respond. To more efficiently handle such failure, IGF couples the routing and MAC components into a single integrated Network/MAC protocol. As we later explain in more depth, our solution utilizes non-deterministic routing, where the next-hop receiver is determined during MAC layer handshaking. We note, however, that our solution is not dependent on MAC contention schemes and could be implemented with small modifications to several existing protocols. We therefore introduce a brief history of MAC layer contention, not for comparison, but as a complementary introduction and potential optimization of our work. By this we mean that, while our protocol, as implemented, directly builds off of the more traditional 802.11 DCF contention scheme, we simply present our work in this context as one feasible, albeit chosen, alternative. This chosen MAC contention scheme is therefore important, but by no means immutable.

The basic CSMA/CD MAC, as proposed for sensor networks, provides carrier sensing prior to message transmission. Issues arise such as hidden terminal interfering with current transmissions resulting in frequent collisions. The 802.11 protocols, which build off of the underlying MACA handshake, are designed to address this problem. After performing Carrier Sensing for a Distributed Inter Frame Spacing (DIFS) time to detect an idle channel, the basic 802.11 DCF uses Request to Send (RTS), Clear to Send (CTS), Data, Acknowledgment (ACK) handshaking when transmitting a Unicast packet. NAV timers are used to monitor the expected channel occupancy time and a Backoff Window is used to handle contention. While the basic 802.11 DCF provides an algorithm for contention that can be used in wireless ad hoc or sensor networks, we note that more advanced protocols [32][35], developed in response to specific system demands and limitations, have been proposed. Because the focus of our work is a novel idea in non-deterministic routing, we build IGF off of the basic 802.11 DCF protocol, as implemented in GloMoSim, noting that our scheme could be implemented in accordance with alternate schemes.

As a final consideration of relevant work, it is important to identify the systems and applications that may run on top of WSNs, to better understand any requirements imposed on lower layer protocols. While WSNs have been proposed for applications in consumer based Smart Environments [25], we foresee the main use of these systems to be the environmental monitoring of remote or inhospitable terrain. This includes applications proposed to monitor nature [17], assist during disaster relief and search and rescue operations, or to be deployed in enemy territory for target tracking and surveillance [3][23]. Such systems, aside from requiring general communication and coordination services, will likely differ in the required granularity of application level services. For search and rescue workers using a WSN during an earthquake, high levels of node density will be desirable to get the most accurate location of a victim possible. Alternately, in tracking applications, some accuracy will likely be sacrificed to ensure system longevity. In these and other scenarios, having a protocol that works independent of application requirements is desirable. In addition, both mobile and immobile WSNs will benefit from a protocol that maximizes throughput and minimizes communication overhead, despite the existence of state transitions that are unpredictable at the network layer.

3 Protocol Description

IGF is a combined Routing/MAC protocol that assumes nodes have knowledge of their location (and optionally their remaining energy) to make non-deterministic forwarding decisions when routing point-to-point traffic. In this section we give a complete description of our baseline protocol, as implemented and tested in our wireless simulator. Specifically we highlight the state-free features of our work, showing how IGF provides:

- Robust performance when nodes migrate or transition into and out of sleep states
- The immediate utilization of recently awoken or newly arriving nodes
- Shorter end-to-end latency compared to schemes that must update system state prior to sending
- The elimination communication overhead to maintain system state
- Distance and energy aware forwarding

- Distribution of the workload
- The decoupling of routing from energy conserving protocols

3.1 IGF

IGF is conceived to provide robust and effective communication under transient conditions that state based routing solutions cannot handle well. While working on a WSN demo that "merged" several novel protocols developed at various institutions, we realized that node dormancy periods, introduced somewhat haphazardly to conserve energy and distribute participation amongst nodes, introduces a complex dynamic into the system. As nodes transition into and out of sleep states, participation in the network becomes probabilistic at any given point in time. Although most sensor network protocols have been developed with robust characteristics in mind, the level of fault tolerance is usually designed to adapt to occasional node failure. Thinking more about these state transitions, we extended this problem to include mobility, as node migration also introduces complexity when maintaining neighborhood tables. We, therefore, see protocols that rely on particular nodes to perform a function (e.g. route a message onward to a destination) as being problematic in this context. Previous state-based solutions (DSR, LAR, GF, and virtually every other WSN routing protocol we've come across) fit this category.

The basic idea we introduce in this work is to non-deterministically route packets by allowing next-hop candidate nodes to "compete" in the forwarding process. Removing the exclusivity of assigning a next-hop candidate permits orthogonal energy conserving protocols to function irrespective of routing protocol requirements. This feature eliminates costly communication that would otherwise be required to maintain neighborhood state information for routing. In addition to removing undo restrictions, our protocol enhances the decision making process by incorporating **increased distance toward the destination** (IDTD) and **energy remaining** (ER) metrics into the route selection process.

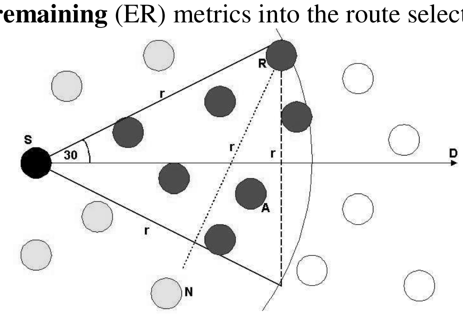


Figure 1: Forwarding Area for Source S

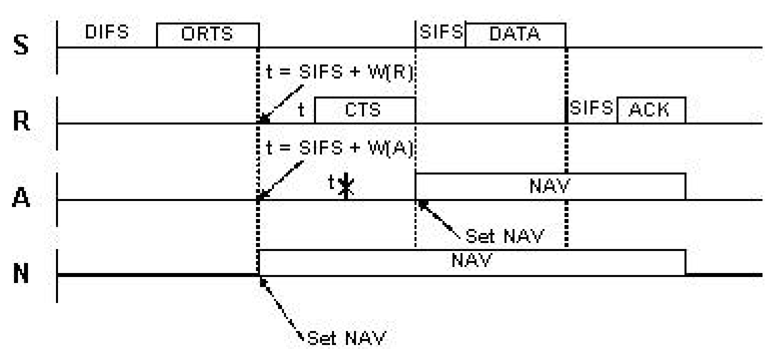


Figure 2: IGF Handshake

To help present our protocol, Figure 1 depicts an example topology where source node S is transmitting a message toward destination D. While many nodes are candidate nodes – a term we define in more depth later – we highlight two nodes, R and A, to represent the chosen next-hop and an alternate "competing" node, respectively. In addition, node N represents a node within communication range of S that is not a candidate node. The communication handshake for this topology is provided in Figure 2. The following protocol description makes reference to these figures.

With slight modifications to the 802.11 DCF MAC protocol from which our work is derived, the IGF handshake begins when the sender S's NAV timer is zero and it carrier senses an idle channel for DIFS time. Having validated that the channel is free, S sends, via broadcast, what we deem an ***Open RTS*** (ORTS). While all nodes within communication radius of S receive and process this request to send, only neighboring nodes (R and A), within the confined **forwarding area** (discussed in more depth below), set a ***CTS_Response*** timer (t) that defines an appropriate amount of time that must pass before responding to the received ORTS message. The value of this timer depends on the increase in distance toward the destination, the energy remaining at the potential receiver, and an additional random value. A formula for this calculation follows in Section 3.1.2. All nodes receiving the ORTS that are outside this forwarding area (N) set their NAV timer in accordance with 802.11 semantics. We call the nodes in this forwarding area, **candidate nodes.** While all candidate nodes set their CTS_Response timer, only a single node, specifically the node that assigns the shortest time value (R), will respond to the ORTS. To prevent multiple responses, candidate nodes hearing a CTS response will cancel their timers and set their appropriate NAV timers. In addition, the Sender S, having already received a valid CTS, will no longer acknowledge further CTS messages heard in response to the now antiquated ORTS. Having isolated a communication channel with a specific receiver, the Sender responds to the received CTS and communication continues, also in accordance with 802.11 DCF semantics (CTS->DATA->ACK).

3.1.1 Forwarding Candidates

As previously specified, neighboring nodes receiving an ORTS must determine whether or not they are a valid forwarding candidate for two reasons. First, to ensure that a message is propagated on a progressive path toward the ultimate destination; and second, to ensure that every node within the forwarding area is capable of hearing one another, to prevent interference between candidate nodes. While the first "reason" is intuitive, the second deserves some justification. Specifically, we ensure that all candidate nodes responding to a ORTS are within communication range of one another, to ensure that no node responds with a CTS after the handshaking process has begun. Referring back to Figure 1, if node N had also set a response timer, its inability to receive messages from R would prevent N from knowing that a response to S's ORTS has already been transmitted. If N were to respond during R's CTS transmission, a collision would occur at Sender S.

As depicted in this diagram, we choose candidate nodes that reside within a 30-degree angle of the line connecting the Sender and final Destination. Using Sender, final Destination, and the Receiver's own location, obtained using GPS or some distributed localization protocol, we use simple trigonometry to calculate this angle. In our work we assume that a high enough node density exists to ensure that there is at least one candidate node within the forwarding area between a Sender and the final Destination.

As stated before, neighboring nodes that receive an ORTS, but are not within the defined forwarding area, simply set their NAV timer to reflect the duration of communication. This prevents later collisions that result from hidden terminal affects. Since an ORTS message does not contain an absolute duration of transmission (the waiting time of the CTS responder is unknown at this time), NAV vectors are set at the minimum time required for communication. Subsequent packets (CTS, DATA, ACK) then allow this timer to be updated reflecting a more accurate transmission duration.

3.1.2 Setting Response Wait Times

Having determined that it is within the forwarding area of communication, a node sets its *CTS_Response* timer according to the IDTD and ER parameters, plus an additional random delay. IDTD measures the increase in distance toward the destination that a message will take if the Receiver is appropriated the responsibility of communication. Less time is contributed to the *CTS_Response* timer for a greater IDTD in accordance with equation 1, with no additional time is added when the distance closer to the destination is equivelant to the presumed radio radius (i.e. maximum possible distance). ER measures the energy remaining in a node as a fraction of the maximum energy provided upon deployment. In our work we assume that a node is cognizant of its remaining energy and deployment energy, and we additionally assume that all nodes are initially deployed with the same maximum energy. Like IDTD, less time is contributed to the *CTS_Response* timer for a greater ER value. This makes nodes with more available energy more likely to participate in communication. Finally, to further disperse the system workload, randomization is included in the *CTS_Response* time.

IDTD = Increase in Distance Toward the Destination ER = 1 – Energy Consumed by Receiver / Maximum Energy of Mote Rand: Random number between 0 and 1 W_D, W_E, W_R: Weight of Distance, Energy, and Random Parameters R: Radio Range of Motes SIFS: Short Inter Frame Spacing (From 802.11) M: Maximum Additional Wait Time (Note! M+SIFS < DIFS) $D_{Parameter} = C_D * IDTD / R$ $E_{Parameter} = C_E * ER$
$F = 1 - ((D_{Parameter} + E_{Parameter} + Rand) / (W_D + W_E + W_R))$
CTS_Response = SIFS + M * F

Equation 1: Calculating Response Wait Times

When setting *CTS*_Response, we must not allow the timer to extend beyond DIFS to ensure that other nodes waiting to transmit will not interject with an ORTS packet that will cancel current CTS timers. Such behavior could severely impede throughput (for a discussion of this issue see [29]).

In equation 1, we allow for the tuning of each parameter using weights. Additionally, we note that while our current parameters for calculating *CTS_Response* time include only distance and energy remaining, alternate parameters used in this calculation could include the probability of packet loss, processor load, or any other pertinent, influential, and available information.

3.1.3 PseudoCode

Now that we have described the basic protocol, we provide the following pseudo-code to clarify the communication process.

Sender		***Receiver***
// Sender has a packet to transmit Carrier Sense until (Channel is idle for DIFS AND NAV == 0)		
Send ORTS Set ORTS_Wait Timer	→	Receive ORTS If *InForwardingArea* (Source, Receiver, Destination) CTS_Response = *CalculateWait*(energy, distance) else set NAV = Expected_Transmission_Duration
If (ORTS_Wait Timer Expires) Backoff and Re-transmit else // CTS Received before Timeout		If CTS Overheard // sent by another candidate node Set NAV = Expected_Transmission_Duration Cancel CTS_Response Timer else // CTS_Response Timer Expires
Receive CTS	←	Send CTS

// Communication established
// Proceed in accordance with 802.11
// semantics		

3.2 Optimizations

While IGF performs extremely well under conditions of high node density, the absence of a candidate node will result in communication failure. To remedy this problem, IGF can optionally implement a **forwarding area shift**. Similar to GPSR's implementation to routing around voids, this works when, having sent a message down to our modified MAC protocol, the MAC layer notifies our Network layer of single hop failure. Realizing that such failure could be the result of an absent or unavailable candidate node, the Network can then retransmit the message, requesting a shift in the forwarding area to "search" for an available receiver. Subsequent shifts, when necessary, will allow IGF to utilize the communication area outside of the initial forwarding area. In addition to avoiding sparse areas using forwarding area shifts, techniques such as backpressure [8] can be used to avoid network voids.

4 Experiments and Analysis

4.1 Simulation Overview

Parameters	**Setting**
Radio Communication Radius	40 meters
Bandwidth	200 kbps
Packet Size	32 byte CBR Payload
Terrain	150X150 m^2
Nodes	100

Table 1: System Parameters

To assess the performance of our protocol, we implement IGF in GloMoSim [1], a wireless simulator for sensor, ad hoc, and mobile networks. GloMoSim models the communication architecture from physical layer bit transmissions, including signal interference and attenuation patterns, through application layer traffic loads. To make our work as close in proximity as possible to existing hardware proposed for use in WSN environments [35][41], we set our system parameters as shown in Table 1.

One-to-One	
Flows (node → node):	50→59
Initial Flow Rate	4 packet/second 32 byte CBR payload
Total packet sent	100
Step Change Between Experiments	4 packet/second
Number of Changes	10

Many-to-One	
Flows (node → node):	10→59, 50→59, 80→59
Initial Flow Rate	1 packet/second 32 byte CBR payload
Total packet sent	300
Step Change Between Experiments	2 packet/second
Number of Changes	10

Many-to-Many	
Flows (node → node):	10→49, 50→49, 80→49, 10→69, 50→69, 80→69
Initial Flow Rate	1 packet/second 32 byte CBR payload
Total packet sent	600
Step Change Between Experiments	1 packet/second
Number Changes	10

Table 2: One-to-One, Many-to-One, and Many-to-Many CBR Traffic Flows

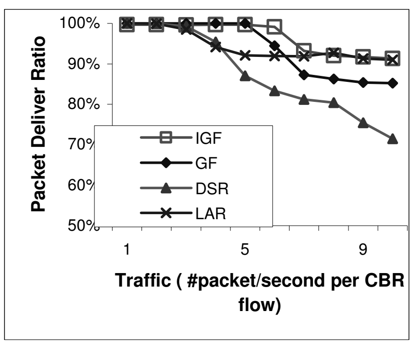
For our experiments we configure a terrain of 150X150 square meters with 100 motes distributed uniformly throughout the terrain. Nodes are numbered left to right, top to bottom, in randomly distributed rows of 10. Experiments are set up to test One-to-One, Many-to-One, and Many-to-Many constant bit rate (CBR) traffic flows. These flows are chosen to mimic periodic point-to-point communication expected in such systems. The details of these flows, including sending and receiving nodes, flow rates, the number of packets sent, and the state space of exploration, are provided in Table 2. For each configuration we average 60 runs with different random seeds to ensure adequate confidence of our results. Almost all of the results presented are within 3% to 10% of the mean for GF and IGF, and 8% to 20% for LAR and DSR.

When determining the CTS_Response timer weights (Section 3.1.2) we studied various settings to choose the best configuration for the node density and communication radius chosen. Because our implementation of sleep cycles does not coincide with GloMoSim's energy model, we set the contributing weight of our energy parameter to 0. However traffic dispersion still exists due to the influence of our random parameter. We set the contributing weight of our distance and random parameters to 2 and 1 respectively. We intend to further study the affect of the energy parameter on system lifetime and workload distribution in future work.

We compare IGF over the modified 802.11 protocol previously described, with GF/802.11, LAR/802.11, and DSR/802.11. We chose these to compare IGF against a well-established classical routing protocol (DSR), a protocol optimized for mobility (LAR), and what has become the standard location based solution for WSNs (GF). We consider these protocols under three system scenarios. The first, which we call the **traditional** test, rivals these routing protocols in the less practical case, where nodes are not mobile and energy conservation is not considered. The second flavor of experiments introduces **mobility** into our assessment. Finally, the third evaluates these protocols in the presence of energy conserving **sleep** cycles. For each experiment we collect data on the packet *Delivery Ratio* (number of packets received / number of packets sent), average end-to-end communication *Delay* (average time in network of received packets) in ms, and overall communication *Overhead* (total number of packets sent at the Radio layer).

4.2 Traditional System Test

We consider the first set of experiments a baseline for comparison. This traditional scenario shows that IGF performs as well or better than GF, DSR, and LAR, even when mobility and energy conserving sleep cycles are not considered. In these tests the aforementioned flows (Table 2) are simulated, increasing the rate of transmission until sufficient congestion is seen. Because the One-to-One, Many-to-One, and Many-to-Many configurations all produce similar results, we only show graphs for the more complex and interesting Many-to-Many flows.



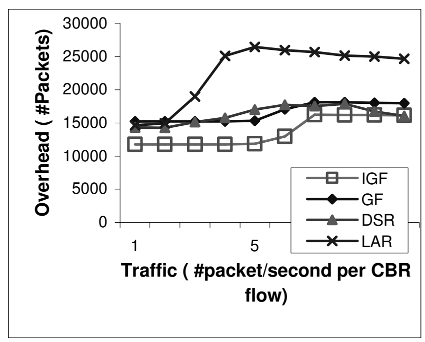


Figure 3A: Packet Delivery Ratio **Figure 3B: Communication Overhead**

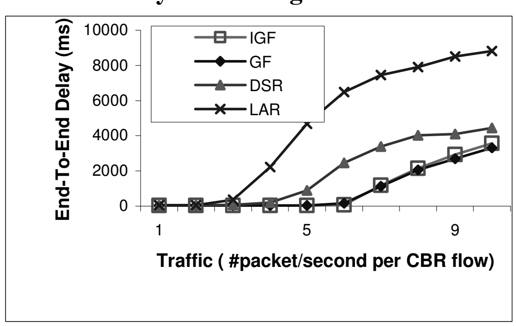


Figure 3C: End-to-End Communication Delay

From Figure 3A, we see that GF and IGF have comparable delivery ratios under light loads, while LAR and DSR lose packets early as these protocols quickly congest the network by sending route discovery packets. When traffic flow rates increase enough to adequately congest the network in GF and IGF (6+ packets/second per CBR flow), performance in GF quickly degrades along limited intersecting routes, suffering additional congestion caused by neighbor table update beacons. While DSR, LAR, and IGF all distribute traffic loads along multiple routes for each Sender/Receiver pair, only LAR and IGF perform well under heavy loads as DSR sees significant message loss due to flooded route discovery packets. LAR uses location information to keep the effects of route discovery to a minimum allowing it to maintain delivery ratios comparable to IGF. However, LAR's frequent transmission of route discovery packets toward the destination, coupled with the latency incurred awaiting the route discovery response, lead to significantly more Overhead and longer Delay when compared with IGF. This phenomenon can be observed in Figures 3B and 3C, as the graph of LAR spikes around 4 packets/second per CBR flow. Figure 3B demonstrates IGF's savings in communication Overhead at lower traffic loads, as IGF does not require beaconing (GF) or route discovery packets (DSR, LSR) required in these other protocols. As traffic loads increase, congestion increases the number of MAC layer collisions in IGF and GF, resulting in retransmission attempts that add to Overhead. For DSR, the Overhead actually diminishes because packet loss and the failure of route discovery packets to make it back to the source lead to fewer transmission attempts as messages are dropped early. Figure 3C demonstrates how local routing decisions introduce less end-to-end delay when compared to routing protocols that require complete paths between a source and destination a priori. In GF and IGF we see significantly lower end-to-end Delay beyond 4 packets/second per CBR flow because DSR and LAR suffer latency awaiting the return of route discovery packets. This effect becomes less apparent in DSR under heavy traffic because DSR's low delivery ratio leads to fewer packets contributing to this metric. Finally under heavy traffic, we see slightly longer delay in IGF over GF due to the additional receiver wait time used when choosing a next hop. We justify this slight increase in delay later in our analysis when we consider IGF's performance in the mobility and energy conservation experiments.

4.3 Mobile System Test

To test performance in the presence of mobility, we allow nodes to migrate during the simulation according to a waypoint model with a one second pause time between moves. Traffic is set at a rate of 1 packet/second per CBR flow to prevent congestion and therefore isolate the effects of mobility in our analysis. We model speeds up to 18 meters per second (~40 mph) to evaluate performance at reasonable walking and vehicular speeds (for motes attached to soldiers or piggybacked on tanks or military vehicles). In these experiments source nodes are mobile; however, we anchor destination nodes due to the location-based nature of WSN routing. Again, the consistent performance between configurations allows us to only show results for Many-to-Many flows.

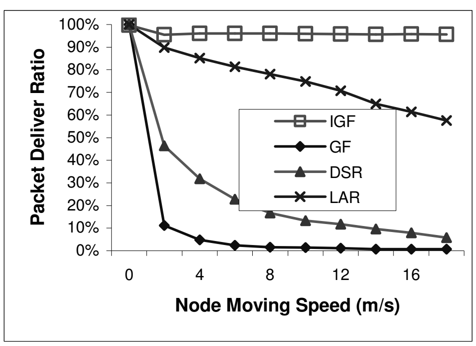


Figure 4A: Packet Delivery Ratio

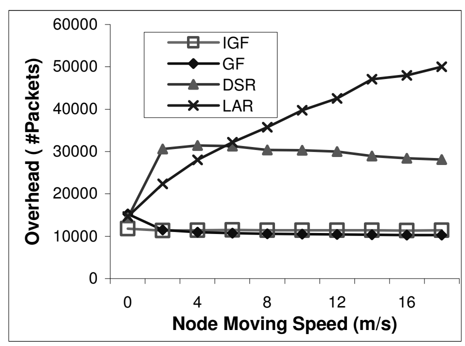


Figure 4B: Communication Overhead

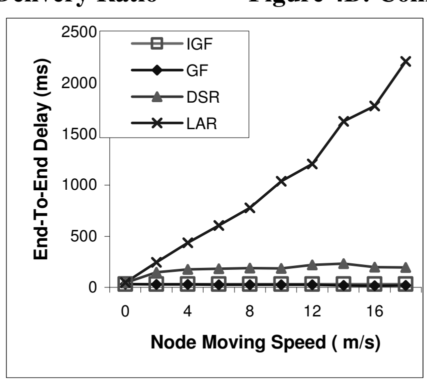


Figure 4C: End-to-End Communication Delay

From Figure 4, we see that when nodes do not move (0 m/s), no messages are lost and the lowest Delay and Overhead are incurred due to minimal congestion. As we introduce mobility, increasingly affecting the validity of neighborhood and routing state with increased node speeds, we see the delivery ratios in GF, DSR, and LAR drop off quickly while IGF continues to perform at close to optimum. For DSR and LAR, performance degrades as node migration invalidates pre-established routes. LAR's milder degradation, seen in Figure 4A, results from location controlled flooding of route discovery messages that reestablish routes despite mobility. We do note, however, that while LAR achieves Packet Delivery ratios above DSR and GF, it does so at the expense of end-to-end Delay and Overhead (Figure 4B and C), suffered as a result of route discovery. In contrast, DSR's uncontrolled flooding during route discovery congests the network quickly, preventing DSR from ascertaining routes to the destination. The Overhead for DSR therefore increases slowly because fewer transmissions are attempted as a result of message loss.

As an addendum to explain why GF performs poorly, we note from Figure 4A that GF's Delivery Ratio quickly drops to zero at relatively low node speeds. This happens because the chosen next-hop mobile node is located at the outer reach of the sender's communication radius. Because nodes are equally likely to move in any direction, there is approximately a 50% chance that the designated receiver will have moved away from the sender, moving it out of communication range since the last received beacon. Over multi-hops the chance of failure grows exponentially. To optimize GF, we increase the rate at which nodes beacon to combat this effect, keeping the neighborhood state as up to date as possible. However, because beacons consume bandwidth, their cost offsets their savings, arriving at similar results in all of the configurations we tested.

4.4 Energy Conserving System Test

We test IGF, GF, DSR, and LAR in the simulated presence of orthogonal energy conserving protocols by randomly transitioning nodes into and out of sleep states at various **Toggle Periods**[2] (ranging from 300 milliseconds to 31 seconds). For the same reasons discussed in our mobility experiments, we set the flow rate to 1 packet per second. Under these conditions we note that transitioning nodes to sleep significantly affects node density as higher **Sleep Percentages**[3] reduce the number of nodes participating in routing at any point in time. Across experiments we vary the Toggle Period and Sleep Percentage under the system configurations previously discussed. We then look at the packet Delivery Ratio to assess performance.

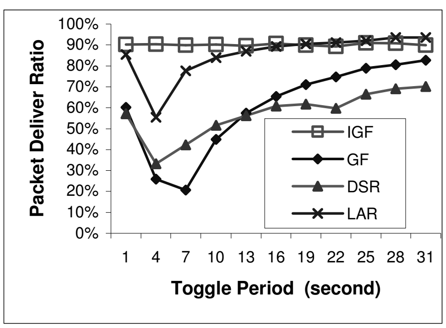


Figure 5A: Packet Delivery Ratio

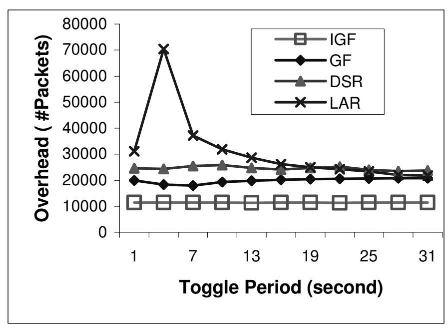


Figure 5B: Communication Overhead

Figure 5A shows results for Many-to-Many flows where the Sleep Percentage is set at 33% for varied Toggle Periods. We note that the One-to-One and Many-to-One flows at varied Sleep Percentages produce similar results. From this Figure we can see that IGF outperforms all other protocols when the Toggle Period is below 18 seconds. In this range, the first data point (300 ms) shows that all protocols do relatively well as transmissions that initially fail at the MAC layer often succeed during later MAC layer retransmissions. This occurs because the toggle period is low enough for nodes that were previously asleep to wake up between retransmission attempts. The immediate drop off in the graph, seen after this first data point, occurs because the increase in Toggle Period makes receiver nodes less likely to wake up during the MAC layer retransmission process. A node then requires the network layer to handle transmission failures

---

[2] Toggle Period is the time interval between subsequent transitions into a sleep state.
[3] The percentage of the Toggle Period in which the node is in the sleep state.

by initiating route discovery (DSR, LAR) or by electing an alternate neighbor to act as a next-hop (GF). When this occurs, GF performs worse than DSR and LAR, as the recently returned route discovery packet will traverse nodes that are currently awake and therefore able to act as routers. For GF, the choice of a next-hop is independent of the current state of the sender's neighbors, resulting in the possibility that the next chosen receiver will also be sleeping when the transmission is retried. DSR's performance is worse than LAR because of the bandwidth consumed during the less efficient route discovery process. Finally, we see IGF perform significantly better than the other protocols, at times showing more than a 350% improvement in packets delivered when compared to GF. We attribute this superior performance to IGF's state-free characteristic, utilizing whatever neighbors are currently awake en route to the destination.

While IGF still outperforms DSR and GF above a 18 second Toggle Period, LAR achieves higher Delivery Ratios with increases in Toggle Period, as route discovery results become valid for longer periods of time. LAR eventually outperforms IGF because it possesses the ability to deliver packets when any path exists that is within the route request zone. However, this increase in Delivery Ratio comes at the expense of significant communication Overhead (Figure 5B) and Delay (graph not shown). IGF's 10% loss in Figure 5A occurs when a node's forwarding area does not contain an available candidate node. A solution to this problem, not yet implemented, was discussed in Section 3.2. This optimization should increase IGF's delivery ratio to almost 100%, assuming that the network is not partitioned.

For systems in which the application may require specific node participation (in the case of a leader) or control over the node density of an area at a given time, we consider Toggle Periods above 20 seconds impractical. We explain this with an example. In a vehicular tracking application where nodes transition to sleep until an event of interest is detected in the environment, the application may want all available nodes to contribute to target identification or tracking. Such an application will therefore require fast toggling to ensure that nodes can be awoken when they are required to participate. Vehicles moving up to 25 meters per second (~56 mph) in a network where motes communicate up to 30 meters requires nodes to wake up and check if they are required for tracking after at most 1 second. For slower objects (5 meters per second) and larger communication radii (100 meters) we still require nodes to wake up within a 20 second interval to ensure they will be available for tracking, consensus, or estimation. Similar justification can be made for alternate systems including systems applicable to disaster relief, environmental monitoring, tracking, or in smart environments.

To supplement our analysis when varying Toggle Periods, we assess routing performance varying the Sleep Percentage for the extremely volatile case where the Toggle Period is set to 100ms. This not only allows us to compare our work under varied Sleep Percentage times, but allows us to stress test our protocol under exceptionally arduous system settings.

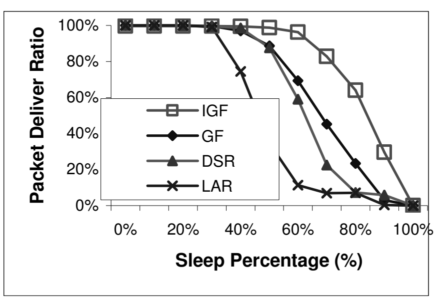


Figure 6A: Packet Delivery Ratio

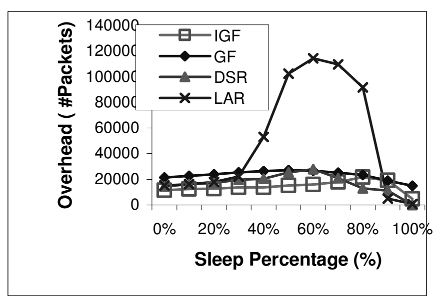


Figure 6B: Communication Overhead

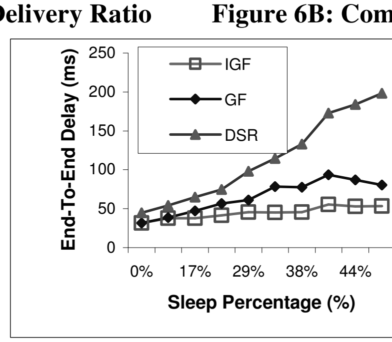


Figure 6C: End-to-End Communication Delay

Figures 6 A, B, and C all demonstrate IGF's optimal performance over varied Sleep Percentages. We note that for Figure 6C, LAR has been removed as its end-to-end Delay climbs off the chart, making it difficult to observe differences between IGF, GF, and DSR. In addition, we only show performance up to a Sleep Percentage value of 50% for this graph, as DSR escalates rapidly beyond this point, also obscuring the graph.

We conclude from these graphs that IGF delivers the highest percentage of packets under all Sleep Percentage settings, while incurring the smallest end-to-end Delay and the lowest transmission Overhead. The marginally higher Overhead of IGF at the 80%+ level, from Figure 6B, is a result of other protocols dropping packets early, thereby only incurring overhead for the small number of packets that actually get transmitted from source to destination. The drastic drop in communication Overhead (Figure 6B) seen in LAR also is attributed to this drop in the Packet Delivery Ratio.

4.5 Experimental Conclusions

In our experiments we assess the performance of IGF when compared to DSR, LAR, and GF for One-to-One, Many-to-One, and Many-to-Many traffic flows. We use Delivery Ratio, transmission Overhead, and end-to-end Delay as our metrics for comparison. For the traditional experiment, we vary the network workload demonstrating that IGF performs as well or better than GF, DSR, and LAR at varied levels of congestion. We then run similar tests under a light workload to assess IGF's abilities when considering mobile nodes. We find that while mobility barely detriments IGF's performance, it severely influences other state-based routing protocols as neighborhood and routing tables require constant repair. Finally, we consider sensor network systems where nodes transition into dormant states to conserve energy. Under varied Toggle Periods we demonstrate that IGF outperforms other solutions by as much as 350% when nodes transition into and out of sleep states at periods less than 30 seconds. We do not consider Toggle Periods above this threshold as we deem them in conflict with possible application protocols that may request the participation of nodes at any time. We also demonstrate IGF's superior performance under a stressed Toggle Period (100 ms). We show that while varying Sleep Percentages, IGF continues to drastically outperform all of the other protocols considered. While we only test IGF in traditional, mobile, and energy conserving experiments, we expect similar results would follow for alternate scenarios where message loss, irregular signal propagation, or other causes of missed or ignored messages make the assigned next-hop receiver invalid or unavailable for routing.

5 Conclusions and Future Work

In WSNs, where nodes may migrate and cycle in and out of sleep states, the maintenance of routing or neighborhood tables is costly. Route discovery messages and beaconing used for such maintenance contribute to network congestion and communication delay, wasting precious energy, causing increased message loss, and introducing end-to-end transmission latency. Additionally, the time delay between actual system state transitions (node migration or transitions to sleep) and network protocol state updates (updating routing and neighborhood tables) leads to Networking layer transmission attempts along invalid paths. To prevent the adverse affects that realistic factors (mobility and sleep cycles) have on state-based routing protocols, we introduce a novel location based routing protocol called Implicit Geographic Forwarding (IGF) that is altogether state-free. IGF uses distance and energy to choose a valid receiver at transmission time, eliminating any reliance on knowledge of neighboring nodes. In simulation we compare our work against DSR, LAR, and GF, showing that IGF outperforms these protocols, in some cases delivering close to 100% of the packets transmitted while alternate solutions fail to even find a path between a source and destination. Specifically, we show that our protocol demonstrates a vast improvement over prior work using metrics of delivery ratio, control overhead, and end-to-end delay. In future work we intend to implement the optimizations discussed for extending the forwarding area and invoking backpressure. We also hope to implement and experiment IGF, integrated into a complete WSN architecture, in our MICA test bed.

6 References

[1] L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia, and M. Gerla. "GloMoSim: A Scalable Network Simulation Environment," *UCLA Computer Science Department Technical Report 990027,* May 1999.

[2] F. Bennett, D. Clarke, J. B. Evans, A. Hopper, A. Jones, and D. Leask. "Piconet – Embedded Mobile

Networking," *IEEE Personal Communications,* October 1997.

[3] B. M. Blum, P. Nagaraddi, A. Wood, T. F. Abdelzaher, S. Son, J. A. Stankovic. "An Entity Maintenance and Connection Service for Sensor Networks," *The First International Conference on Mobile Systems, Applications, and Services (MOBISYS '03)*, California, May 2003.

[4] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. "SPAN: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks," *ACM/IEEE International Conference on Mobile Computing and Networks (MOBICOM '01),* Rome, July 2001.

[5] D. Coore, R. Nagpal, and R. Weiss. "Paradigms for Structure in an Amorphous Computer," *MIT Artificial Intelligence Laboratory Memo no. 1614,* 1997.

[6] D. Estrin, R. Govindan, J. S. Heidemann, and S. Kumar. "Next Century Challenges: Scalable Coordination in Sensor Networks," *Proceedings of Mobile Computing and Networking (MOBICOM '99)*, Washington, August 1999.

[7] D. Ganesan, R. Govindan, S. Shenker and D. Estrin. "Highly Resilient, Energy Efficient Multipath Routing in Wireless Sensor Networks," *In Mobile Computing and Communications Review (MC2R '02),* 2002.

[8] T. He, J. A. Stankovic, C. Lu, and T. F. Abdelzaher. "SPEED: A Stateless Protocol for Real-Time Communication in Sensor Networks," *In International Conference on Distributed Computing Systems (ICDCS 2003)*, Rhode Island, May 2003.

[9] C. Intanagonwiwat, R. Govindan and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," *The Sixth Annual International Conference on Mobile Computing and Networks (MobiCOM '00)*, Massachusetts, August 2000.

[10] D. B. Johnson and D. A. Maltz. "Dynamic Source Routing in Ad Hoc Wireless Networks." *In Mobile Computing*, 1996.

[11] P. Karn. "MACA – A New Channel Access Method for Packet Radio," *In ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, September 1990.

[12] B. Karp. "Geographic Routing for Wireless Networks," *Ph.D. Dissertation*, Harvard University, Cambridge, MA, October, 2000.

[13] B. Karp and H. T. Kung. "Greedy Perimeter Stateless Routing for Wireless Networks," *In Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM '00),* Massachusetts, August 2000.

[14] O. Kasten, Energy Consumption – Wireless LAN Module Specs, http://www.inf.ethz.ch/~kasten/research/bathtub/energy_consumption.html, Eldgenossische Technische Hochschule Zurich.

[15] Y. Ko and N. H. Vaidya, "Location-Aided Routing (LAR) in Mobile Ad Hoc Networks," *The Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM '98)*, Texas, October 1998.

[16] C. Lu, B. M. Blum, T. F. Abdelzaher, J. A. Stankovic, and T. He. "RAP: A Real-Time Communication Architecture for Large-Scale Wireless Sensor Networks," *In IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS '02), California,* September 2002.

[17] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, J. Anderson. "Wireless Sensor Networks for Habitat Monitoring," *Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '02),* Georgia, September 2002.

[18] A. B. McDonald and T. Znati. "A Mobility Based Framework for Adaptive Clustering in Wireless Ad-Hoc Networks." *IEEE Journal on Selected Areas in Communication (JSAC '99),* August 1999.

[19] R. Nagpal and D. Coore. "An Algorithm for Group Formation in an Amorphous Computer," *Proceedings of the 10th International Conference on Parallel and Distributed Computing Systems (PDCS '98)*, Nevada, Oct. 1998.

[20] C. E. Perkins and P. Bhagwat. "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *In Symposium on Communications Architectures and Protocols (SIGCOMM '94),* London, September 1994.

[21] C. E. Perkins and E. M. Royer. "Ad-Hoc On Demand Distance Vector Routing," *2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99),* Louisiana, February 1999.

[22] R. C. Shah and J. M. Rabaey. "Energy aware routing for low energy ad hoc sensor networks," *In Proceedings of IEEE Wireless Communications and Networking Conference (WCNC '02)*, Florida, March 2002.

[23] C. Shen, C. Srisathapornphat, and C. Jaikaeo, “Sensor Information Networking Architecture and Applications,” *IEEE Personal Communications*, August 2001.

[24] S. Singh and C. S. Raghavendra. “PAMAS: Power Aware Multi-Access Protocol with Signalling for Ad-Hoc Networks,” *ACM Computer Communications Review,* 1999.

[25] M. Srivastava, R. Muntz, M. Potkonjak. “Smart Kindergarten: Sensor-Based Wireless Networks for Smart Developmental Problem-Solving Environments,” *ACM/IEEE International Conference on Mobile Computing and Networks (MOBICOM ’01),* Rome, July 2001.

[26] F. Stann and J. Heidemann. “RMST: Reliable Data Transport in Sensor Networks.” In Proceedings of the First International Workshop on Sensor Net Protocols and Applications (SNPA ’03), Alaska, Apr. 2003.

[27] M. Stemm and R. H. Katz, “Measuring and Reducing Energy Consumption of Network Interfaces in Hand-Held Devices,” *IEICE Transactions on Communications*, vol. E80-B, no. 8, pp. 1125-1131, Aug. 1997.

[28] L. Subramanian and R. H. Katz. “An Architecture for Building Self-Configurable Systems,” *International Conference on Mobile Computing and Networking (MOBICOM ’00)*, Massachusetts, 2000.

[29] M. T. Sun, L. Huang, A. Arora, and T. H. Lai. “Reliable MAC Layer Multicast,” *In Proceedings of International Conference on Parallel Processing (ICCP ‘02),* Vancouver, 2002.

[30] D. G. Thaler and C. V. Ravishankar. “Distributed Top-Down Hierarchy Construction,” *In Proceedings of the IEEE INFOCOM*, 1998.

[31] C. Y. Wan, A. T. Campbell, and L. Krishnamurthy. “PSFQ: A Reliable Transport Protocol For Wireless Sensor Networks”, *Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA 2002)*, Georgia, Sept. 2002.

[32] A. Woo and D. Culler. “A Transmission Control Scheme for Media Access in Sensor Networks,” *ACM/IEEE International Conference on Mobile Computing and Networks, (MOBICOM ‘01)*, Rome, July 2001

[33] Y. Xu, J. Heidemann and D. Estrin. “Adaptive Energy-Conserving Routing for Multihop Ad Hoc Networks,” *Technical Report 527,* USC/Information Sciences Institute, October 2000.

[34] Y. Xu, J. Heidemann, and D. Estrin, “Geography-informed Energy Conservation for Ad Hoc Routing,” *ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM ‘01)*, Rome, July 2001.

[35] W. Ye, J. Heidemann and D. Estrin. “A Flexible and Reliable Radio Communication Stack on Motes,” *Detailed Documentation of the Radio Communication Stack.* http://www.isi.edu/scadds/projects/commstack/

[36] W. Ye, J. Heidemann and D. Estrin. “An Energy-Efficient MAC Protocol for Wireless Sensor Networks,” *In Proceedings of the 21st International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM ‘02)*, New York, June 2002.

[37] F. Ye, H. Luo, J. Cheng, S. Lu, L. Zhang. “A Two-Tier Data Dissemination Model for Large-Scale Wireless Sensor Networks,” *The 8th Annual International Conference on Mobile Computing and Networking (MOBICOM ’02),* Georgia, September 2002.

[38] F. Ye, G. Zhong, S. Lu, L. Zhang. "A Robust Data Delivery Protocol for Large Scale Sensor Networks," *Information Processing In Sensor Networks (IPSN ’03),* California, April 2003.

[39] F. Ye, G. Zhong, S. Lu, L. Zhang. “PEAS: A Robust Energy Conserving Protocol for Long-lived Sensor Networks," *The 23rd International Conference on Distributed Computing Systems (ICDCS ’03),* Rhode Island, May 2003.

[40] Editors of IEEE 802.11. “Wireless LAN Medium Access Control (MAC and Physical Layer (PHY) Specification,” *Draft standard IEEE 802.11,* 1997.

[41] XBOW MICA Mote Specifications:
http://www.xbow.com/Products/Product_pdf_files/MICA%20data%20sheet.pdf