Highly-Resilient, Energy-Efficient Multipath Routing in Wireless Sensor Networks

Deepak Ganesan[a] *deepak@cs.ucla.edu*
Ramesh Govindan[b] *ramesh@icsi.berkeley.edu*
Scott Shenker[c] *shenker@aciri.org*
Deborah Estrin[a] *destrin@cs.ucla.edu*
[a]Computer Science Department, UCLA
[b]International Computer Science Institute, Berkeley
[c]ACIRI, Berkeley

Previously proposed sensor network data dissemination schemes require periodic low-rate flooding of data in order to allow recovery from failure. We consider constructing two kinds of multipaths to enable energy efficient recovery from failure of the shortest path between source and sink. Disjoint multipath has been studied in the literature. We propose a novel braided multipath scheme, which results in several partially disjoint multipath schemes. We find that braided multipaths are a viable alternative for energy-efficient recovery from isolated and patterned failures.

I. Introduction

Sensor networks [3] and other large-scale networks of small, embedded devices may require novel routing techniques [7] for scalable and robust data dissemination. Directed diffusion [5] is an example of such a technique. Directed diffusion incorporates data-centric routing coupled with application-specific in-network processing. Such techniques can help establish energy-efficient data dissemination paths between *sources* (sensors) and *sinks* (data processing or human interface devices). In addition, directed diffusion allows the design of *localized* algorithms for flexible path construction and recovery, enabling these systems to be robust to dynamics.

Earlier work has explored the design of mechanisms for single-path routing in sensor networks [5]. To route around failed nodes, this work assumed periodic, low-rate, flooding of events that enabled local re-routing around failed nodes. In sensor networks, where energy efficiency is of paramount importance, such flooding can adversely impact the lifetime of network. Accordingly, it is desirable to find alternative techniques to provide greater resilience in the presence of failures.

In this paper, we propose using multipath routing to increase resilience to node failure. Multipath routing techniques have been discussed in the literature for several years now (Section V). However, the application of multipath routing to sensor networks and other systems that permit data-centric routing with localized path setup has not yet been explored. We consider two different approaches to constructing multipaths between two nodes. One is the classical node-disjoint multipath adopted by prior work, where the alternate paths do not intersect the original path (or each other). The disjoint property ensures that, when k alternate paths are constructed, no set of k node failures can eliminate all the paths. The other approach abandons the requirement for disjoint paths and instead builds many *braided* paths. With braided paths, there are typically no completely disjoint paths but rather many partially disjoint alternate paths.

In this paper, we address two issues. First, we define localized algorithms for the construction of alternate paths. While it is straightforward to define idealized notions of disjoint and braided paths, as we do in Section II, these definitions do not lend themselves to scalable implementation. For reasons of robustness and energy-efficiency (Section II), sensor network data dissemination mechanisms use localized decisions for path setup and for recovery from failure. In Section II we propose localized algorithms to compute approximations to the idealized disjoint and braided paths.

Second, we evaluate the relative performance of disjoint and braided multipaths. We use two important metrics in judging the performance of these competing approaches. The *resilience* of a scheme measures the likelihood that, when the shortest path has fails, an alternate path is available between source and sink. The *maintenance overhead* of a scheme is a measure of the energy required to maintain these alternate paths using periodic keep-alives. There is an inherent tradeoff between these two quantities. Becoming more resilient typically consumes more energy. In this paper we investigate the tradeoffs that result from the two proposed routing algoirthms.

In Section III we describe our methodology for evaluating the two mechanisms. We look at two different failure modes: *isolated* node failures, where each individual node has an independent probability of failure; and *patterned* failures, in which all nodes[1] within a certain fixed radius fail simultaneously. In Section IV we use simulation results to compare the resilience and maintenance overhead of the two routing methods. We explore behavior of these approaches across several parameters: density, probability of isolated failure, spatial separation of source and sink, and frequency and radius of patterned failures.

We find that, for comparable resilience to patterned failures, braided multipaths expend only 33% of the energy of disjoint paths for alternate path maintenance in some cases, and have a 50% higher resilience to isolated failures.

II. Disjoint and Braided Multipaths

In this section, we briefly review wireless sensor networks and the principles that govern the design of communication mechanisms for such networks. We then describe the need for more energy-efficient alternatives to achieving robust communication in the face of node failure. This sets up our descriptions of localized mechanisms for disjoint and braided multipaths. We evaluate these mechanisms in a subsequent section.

II.A. Sensor Networks

To motivate the problem discussed in this paper, it helps to briefly review the expected capabilities of sensor nodes and sensor networks. It is not unreasonable to expect the following features in a future sensor node: A matchbox sized form factor, battery power source, an power-conserving processor clocked at several hundred Mhz, program and data memory amounting to several tens of MBytes, a radio modem that employs some form of diversity coding [4], and an energy efficient MAC layer based on, for example, TDMA [13]. As such, this node would be capable of running a possibly stripped-down version of a modern operating system; examples of such operating systems include Windows CE and μCLinux. Such a node could have one or more sensors. Examples of such sensors include seismic geophones, infrared dipoles and electret microphones for acoustic sensing.

Because of their compact form factor and their potential low cost, it might be possible for a densely—within tens of feet of each other—packed cluster of such sensor nodes to be deployed, in a possibly unplanned fashion, *near* the phenomena to be sensed. The advantage of such sensor networks is that, even with relatively cheap sensors, these nodes can obtain high SNR (given that the signal generated by any physical phenomena rapidly attenuates with distance). Furthermore, given the spatial density of these deployments, an individual sensor node may *not* have to frequently perform multi-target resolution (*i.e.*, distinguish between different targets such as individuals and vehicles). Such multi-target resolution can involve complex deconvolution algorithms requiring non-trivial processing capability [12].

Three criteria drive the design of large-scale sensor networks: *scalability* (these networks might involve thousands of nodes), *energy-efficiency* (in particular, wireless communication can incur significantly higher energy cost than computation [12]), and *robustness* (to environmental effects and node and link failures).

Previous work [5] has described one paradigm for sensor network communication that addresses these criteria: *directed diffusion*. Directed diffusion consists of several elements. Data generated by sensors is *named* using attribute-value pairs. A sensing task (or a subtask thereof) is disseminated throughout the sensor network as an *interest* for named data. This dissemination sets up *gradients* within the network designed to "draw" events (*i.e.* data matching the interest). Events start flowing towards the originators of interests along multiple paths. The sensor network *reinforces* one, or a small number of these paths.

Not all the elements of diffusion are germane to the issues addressed in this paper. Of particular interest is the notion of path *reinforcement*; that a node in the network may make a local decision (based possibly on perceived traffic characteristics like the observed delay difference between events received along different paths) to draw data from one or more neighbors in preference to other neighbors. We say that such path setup techniques use *localized* algorithms.

II.B. The Problem

Previous work [5] has also discussed how to use directed diffusion to perform energy-efficient and robust dissemination of surveillance data samples from *sources* to *sinks*. A brief, and necessarily simplified, review of that work highlights some of its shortcom-

---

[1] Although we have not explicitly modeled *link* failure, we believe our conclusions will, at least qualitatively, hold up under the influence of link failure.

ings. This solution, which constructs energy-efficient paths[2] on-demand, works as follows:

- A source of sensory data periodically broadcasts, at a low rate, events describing detections of the external phenomenon that is being sensed.
- Upon receiving multiple copies of these events, the sink sends a *reinforcement* message to one of its neighbors indicating that it prefers to receive notifications of detection events at a higher frequency from this neighbor (Figure 1(a))
- That reinforcement message is propagated to the source, hop-by-hop by nodes. Each node makes an independent, local decision about which of its neighbors it chooses to forward the reinforcement, as shown in Figure 1(b) . As it propagates, the reinforcement message implicitly sets up a data path in the reverse direction; that is, at each node, the reinforcement message sets up state that forwards matching data towards the previous hop.
- When a node on the reinforced path fails (Figure 1(c)), the sink detects an absence of detection events and reinitiates reinforcement. For this to work, the sink must continue to periodically send reinforcement messages.

Two characteristics of this description illustrate the main problem motivating the mechanisms considered in this paper. First, that for energy-efficiency reasons, paths are constructed on-demand and not proactively. Second, for reasons of robustness, a periodic low-rate *flooding* scheme notifies the sink and other nodes of available alternate paths. The periodicity of flooding determines the temporal accuracy of alternate path characteristics.

The major drawback of this scheme, with respect to energy-efficiency, is the periodic flooding of low-rate events. This paper considers mechanisms that allow restoration of paths from source to sink without this periodic flooding. These mechanisms are based on the following observation: While setting up the path between a source and a sink, it might be possible *to set up and maintain alternate paths in advance* (at the expense of some energy), in order to minimize the likelihood of having to invoke data flooding for alternate path discovery.

II.C. Multipath Routing

The term *multipath routing* has been used in the literature (Section V) to describe the class of routing mechanisms that allow the establishment of multiple paths between source and destination. Classical multipath routing has been explored for two reasons. The first is *load-balancing*: traffic between a source-destination pair is split across multiple (partially or completely) disjoint paths. The second use of multipath routing is to increase the likelihood of *reliable data delivery*. In these approaches, multiple copies of data are sent along different paths, allowing for resilience to failure of a certain number of paths.

Both these uses of multipath are applicable to wireless sensor networks. Load balancing can spread energy utilization across nodes in a network, potentially resulting in longer lifetimes. Duplicate data delivery along multipaths can result in more accurate tracking in surveillance applications, at the possible expense of increased energy.

But this is not the focus of our paper. Instead, we use multipath routing to rapidly find alternate paths between source and sink. Our rationale for this use of multipath is as follows. Recall that the goal of localized reinforcement-based mechanisms is to empirically (*i.e.* by actually measuring short-term traffic characteristics) establish the "best" path—for some measure of "best" (low latency, low loss *etc*). In what follows, we use the term *primary*[3] path to denote this best path. Thus, we assume that, from the application's perspective, a desirable goal is to deliver data along this primary path. However, to scalably (*i.e.* without periodic flooding) recover from failure of this primary path, we construct and *maintain* a small number of alternative paths that can be used in case the primary path fails.

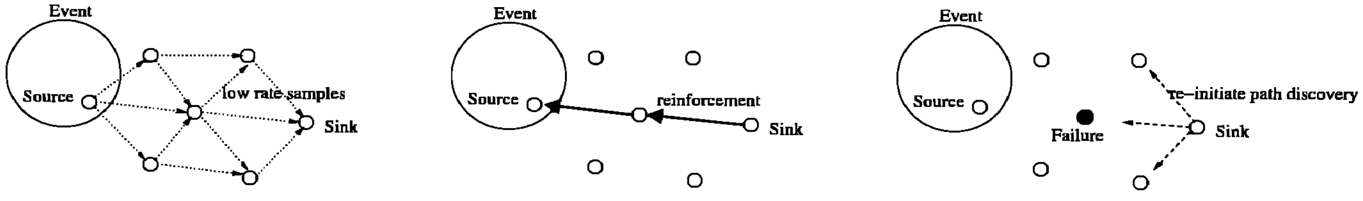
Specifically, when the primary path is set up, the network also sets up multipaths along which data is sent at a low-rate. This low-rate data represents the *energy expended for maintaining multipaths*. We use the term *maintenance overhead* to denote this energy. The low-rate data thus constitutes "keep-alives" on the alternate paths. As soon as a failure is detected on the primary path, nodes can quickly reinforce an alternate path without the need for network-wide flooding to initiate discovery.

However, there may be relatively rare occasions when the primary path, and all alternate paths simul-

---

[2] Diffusion, as described in [5], constructs dissemination paths from multiple sinks to multiple sources. In this paper, we consider multipath dissemination from a single source to a single sink, leaving to future work the extension of our algorithms for multiple source and sinks.

[3] This terminology is appropriate in discussing multipath routing, but is slightly better suited to our purposes than the roughly equivalent term *shortest* path, since we do not restrict ourselves to shortest hop paths.



(a) Low Rate Samples

(b) Path Reinforcement

(c) Recovery from failure along reinforced path

Figure 1: A simplified schematic for Directed Diffusion

taneously fail. In that event, the source or sink must resort to network-wide flooding of data to re-establish the multipath. One measure of the *resilience* of our scheme is how frequently there is a complete failure of the multipath.

Of the many possible designs for multipath routing, we consider two: *disjoint* (Section II.D) and *braided* (Section II.E). Disjoint multipaths have been studied in the literature, but not, to our knowledge, in a sensor network context. Braided multipaths relax the requirement for disjointed-ness. There are many possible ways to relax the disjointed-ness requirements, and we study a particular form of non-disjoint multipaths.

This paper addresses two issues. First, it is not immediately obvious what localized mechanisms might be used to construct disjoint and braided paths. Before we posit the use of multipath routing for energy-efficient recovery in sensor networks, we must clearly demonstrate the existence of such mechanisms. We do this in the following sections. Second, disjoint and braided multipath trade energy for resilience in different ways. We explore this tradeoff via simulation (Section IV).

II.D. Disjoint Multipaths

The first multipath mechanism we consider constructs a small number of alternate paths that are *node-disjoint* with the primary path, and with each other. These alternate paths are thus unaffected by failures on the primary path, but can potentially be less desirable (*e.g.*, have longer latency) than the primary path.

A constructive definition for a k node-disjoint multipath is:

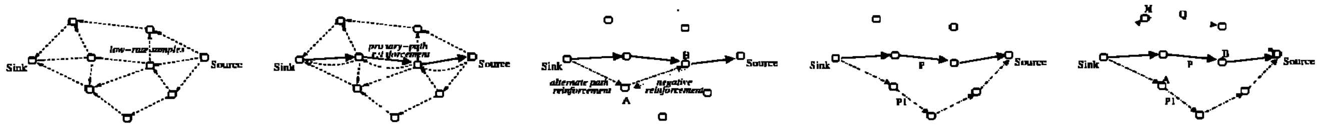
1. Construct the primary path P between source and sink.
2. The first alternate disjoint path P_1 is the best path node-disjoint with P.
3. The second alternate disjoint path P_2 is the best path that is node disjoint with P and P_1, and so on.

This definition assumes global knowledge of topology and network characteristics. For this reason, we call this the *idealized* algorithm for constructing disjoint multipaths, and the resulting multipath the *idealized k-disjoint multipath.*

How do we realize node disjoint multipaths using localized information alone, and not relying on global topology information?

Here's one possible mechanism, which uses two kinds of reinforcements. Assume for the moment that some low-rate samples (Figure 2(a)) have initially been flooded throughout the network (Section II.A). The sink then has some empirical information about which of its neighbors can provide it with the highest quality data (lowest loss or lowest delay). To this most preferred neighbor, it sends out a *primary-path* reinforcement as shown in Figure 2(b). As with the basic directed diffusion scheme, that neighbor then locally determines its most preferred neighbor in the direction of the source, and so on.

After it starts receiving data along the primary path, or perhaps a shortly after sending the primary-path reinforcement, the sink sends an *alternate path* reinforcement to its next most preferred neighbor. This neighbor A propagates the alternate path reinforcement to its most preferred neighbor B in the direction of the source. If B happens to already be on the primary path between the source and the sink (and it can determine this entirely from local state), it sends a *negative reinforcement* to A (Figure 2(c)). A then selects its next best preferred neighbor. Otherwise, B propagates the alternate path reinforcement to its most preferred neighbor and so on (Figure 2(d)). Nodes



(a) Low-Rate samples

(b) Primary-Path P

(c) Alternate-Path - Negative Reinforcement

(d) Alternate-Path P_1

(e) Caveat

Figure 2: Illustrating the construction of localized disjoint paths

other than the sink do not originate alternate path reinforcements.

This mechanism can be extended to construct k disjoint multipaths, by sending out k alternate path reinforcements from the sink, each separated from the next by a small delay. Each node would then be constrained to receive only one reinforcement of either type–primary path, or alternate path. If it receives more than one reinforcement, the node negatively reinforces these, ensuring disjointed-ness.

We call these *localized* disjoint multipaths. They differ from idealized multipaths. In the idealized algorithm, the first alternate path is the primary path which is node-disjoint with the primary path. However, because the localized construction has only local knowledge of alternative paths, it's search procedure may discover longer alternate paths. Figure 2(e) illustrates this difference. In this figure, the sink reinforces A in preference to X, although X leads to a shorter alternate path. This happens because the sink hears events earlier from A, but does not realize that these are forwarded to A by B which is on the *primary-path*. The idealized algorithm would, however, choose Q as the alternate disjoint path. This difference accounts for some performance differences between the two kinds of disjoint multipaths.

II.E. Braided Multipaths

While disjoint paths have some attractive resilience properties, they can be energy inefficient. Alternate node-disjoint paths can be longer, and therefore expend significantly more energy than that expended on the primary path. Since this energy inefficiency can adversely impact the lifetime of a sensor network, we consider a slightly different kind of multipath. Our *braided multipath* relaxes the requirement for node disjointedness. Alternate paths in a braid are partially disjoint from the primary path, not completely node-disjoint.

While there are many possible definitions for non-disjoint multipaths, we pick a simple one for our initial investigation. A constructive definition for our *braided* multipath is (Figure 3): For each node on the primary path, find the best path from source to sink that does *not* contain that node. This alternate best path need not necessarily be completely node-disjoint with the primary path. We call the resulting set of paths (including the primary path) the *idealized* braided multipath. As its name implies, the links constituting a braid either lie on the primary path, or can be expected to be geographically close to the primary path. In this sense, the alternate paths forming a braid would expend energy comparable to the primary path.

One localized technique for constructing braids is described below. Like the idealized algorithm for disjoint multipath, this technique also utilizes two types of reinforcements. However, its local rules are slightly different, resulting in an entirely different multipath structure. As in Section II.D, the sink sends out a primary path reinforcement to its most preferred neighbor A. In addition, the sink sends an alternate path reinforcement to its next preferred neighbor B. Again, as before, A propagates the primary path reinforcement to its most preferred neighbor and so on. In addition, A (and recursively each other node on the primary path) ***originates an alternate path reinforcement*** to its next most preferred neighbor. By doing this, each node thus tries to route around its immediate neighbor on the primary path towards the source. When a node, such as B, not on the primary path receives an alternate path reinforcement, it propagates it towards its most preferred neighbor. When a node already on the primary path receives an alternate path reinforcement, it does not propagate the received alternate path reinforcement any further.

Figure 4 illustrates a localized braid obtained by using the above mechanism. In this figure, n_{k+1} sends an alternate reinforcement to route around n_k that passes through a_k and a_{k-1} before rejoining the primary path at n_{k-2}. In practice, though, our local rules

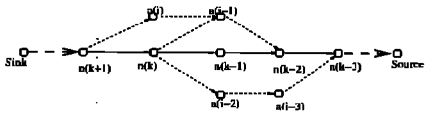


Figure 3: Idealized Braid

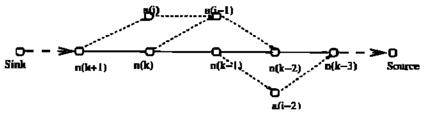


Figure 4: Localized Braid

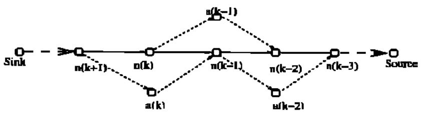


Figure 5: Perfect Braid

cannot always ensure this perfect detour around n_k. An alternate path reinforcement sent out by n_{k+1} can follow any sequence of nodes, possibly completely disjoint from the rest of the primary path, towards the source. Equally, an alternate path reinforcement sent by n_{k+1} can rejoin the primary path at n_k. These effects vary with node density and other factors, and arise due to the incomplete information that the local rules base their decisions upon.

The localized braid is also subtly different from the idealized braid. Consider the alternate path in the idealized braid that does not include n_k. Our idealized construction algorithm does not prevent an alternate path from being chosen which is completely node-disjoint with the primary path. On the contrary, an alternate path on the braid that routes around n_k is constrained to either use the links on the primary path, or other links in the braid, between sink and n_{k+1}. For this reason, the localized braid performs differently from an idealized one, as we shall see in Section IV.

II.F. Qualitative Comparison

Before discussing our simulation results, we try to present some intuition for the energy/resilience tradeoffs of the two multipath schemes we have discussed so far. We use the corresponding idealized mechanisms to guide our intuition, since their behavior easier to reason about than their localized counterparts.

The energy cost of alternate disjoint paths depends on the network density. At low network densities, alternate disjoint paths are significantly longer than, and have higher cost than, the primary path. In addition, for larger k, the overall energy expended in maintaining k-disjoint paths is high. At higher densities, the likelihood of finding node-disjoint alternate paths of shorter length increases, thereby reducing the energy cost of maintaining them.

In the idealized braid an alternate path routes around a single primary path node. The energy cost of an alternate path in the braid is comparable to that of the primary path, more or less independent of density. Thus, at lower densities, the difference in energy expended for multipath maintenance between disjoint multipath and braided multipath is high. This difference decreases with increasing density.

Before we understand the resilience of these multipaths to failure, we need to define our notion of failure. In this paper, we consider two kinds of failures: isolated failures and patterned failures. These failure models are discussed in greater detail in Section III, but intuitively speaking, isolated failures model independent node failure and patterned failures model geographically correlated failure.

Disjoint paths give us independence, *i.e.*, any number of nodes can fail on the primary path without impacting the alternate path. However, the failure of a single node on each alternate path results in the failure of the multipath. By contrast, in braided multipaths, the various alternate paths are not independent, and a combination of failures on the primary path could sever all alternate paths. However, the number of *distinct* alternate paths through a braid is significantly higher than the number of nodes in its primary path. For example, it can be shown that the number of distinct alternate paths in the "perfect" braid of Figure 5 is proportional to the n^{th} Fibonacci number, where n is the number of nodes on the braid's primary path. This contributes to the greater resilience of the braid.

Patterned failures also affect disjoint and braided paths differently. A failure pattern that affects the primary path would be likely to affect alternate paths that are geographically near primary path, and affect less paths that are more distant. Since braiding encourages geographically closer alternate paths, disjoint multipaths are likely to be more resilient to pattern failures than braided multipaths.

This qualitative understanding, however, does not give us any insight into several important questions:

- How much additional energy must one expend in order to increase resilience by a fixed amount?
- How does the energy/resilience tradeoff vary with density or with the extent and frequency of patterned failures?
- How closely do the localized schemes approximate their idealized counterparts?

We explore these questions using simulation.

III. Evaluation Methodology

In this section, we precisely define our two metrics for multipath performance: maintenance overhead and resilience. We also describe the failure models for which we evaluated the resilience of our multipath mechanisms. Finally, we discuss our experimental methodology and list the parameters that affect the multipath schemes.

III.A. Maintenance Overhead

In our multipath schemes, the source periodically floods low-rate data over all alternate paths in the multipath in order to keep alive those paths, thereby permitting fast recovery from failures on the primary path. Clearly, the frequency of these low-rate events determines how quickly our mechanisms recover from failures on the primary path. However, this latency of recovery is not the focus of this paper. Rather, we are interested in knowing the tradeoff between energy expended and the likelihood of total multipath failure. In general, we assume that the latency of recovering from a total multipath failure will be significantly higher than that required for recovering from a failure on the primary path.

For this reason, we *equalize the total volume* of low-rate data sent over alternate paths in each schemes. More precisely, assume that the source disseminates r events in some time interval T over the primary path. Then, we assume that ϵr events are sent on the alternate paths of the disjoint or the braided multipath, with each alternate path receiving equal proportions of this keep-alive traffic. Then, the energy required to maintain the alternate paths is proportional to the average length (in number of hops) of the alternate paths. To meaningfully calibrate the maintenance overhead, we normalize it with respect to the length of the shortest path. Thus, our maintenance overhead metric is:

$$(L_a - L_p)/L_p \qquad (1)$$

where L_a is the average length of an alternate path, and L_p is the length of the primary path.

Because it measures energy dissipation in terms of average path length, our maintenance overhead metric is only a coarse measure of dissipated energy. Our experiments do not attempt to compute per packet energy dissipation with realistic radio simulations. In our case, this is justified since we are only really interested in comparing the performance of our multipath schemes. Furthermore the use of path length in terms of hop counts is a reasonable indicator of energy in radios that have fixed transmission and reception power levels.

III.B. Failures

Lacking realistic failure models or empirical data on node failures in wireless sensors, we study the resilience of our multipath routing schemes to two widely different failure models: independent node failures, and geographically correlated failures. We do not claim that either model is representative of reality. However, these models are different enough that they give us some understanding of the behavior of our multipath schemes across a wide variety of failure types.

Isolated Failures Our first failure model captures independent node failures. More precisely, each node in the multipath has a probability of failure p_i during some small interval T. Then, for each of our multipath schemes, we define *resilience to isolated failure* to mean the probability of at least one alternate path being available within the interval T, given that *at least one node on the primary path has failed.* This latter constraint captures our use of multipath routing for recovery from shortest path failure.

Isolated failures are not completely divorced from reality. They can represent failure due to energy dissipation or localized environmental effects at low deployment densities. For example, if the radius of physical activity is smaller than inter-sensor separation, then localized physical activity may trigger at most one sensor node. In these situations, that sensor may dissipate all its energy tracking local activity, quite independent of neighboring sensors.

Patterned Failures Our second failure model captures geographically correlated failures. Specifically, a patterned failure results in the failure of all nodes a circle of radius R_p. The choice of a circle is somewhat arbitrary, but attempts to model the idealized wave propagation of most physical phenomena. The rough justification for this model is that sustained activity or environmental effects (such as rain fades) within a geographic region can cause such correlated failure, either due to loss of connectivity or due to energy dissipation.

We assume location of the centers of these circles is randomly distributed within the sensor field. Furthermore, lacking any other realistic model, we assume that the number of patterned failures within a given small time interval T is Poisson distributed, with some parameter λ_p.

Then, for each multipath scheme, its *resilience to patterned failure* is defined as the probability that, within a small interval T: at least one alternate path is available between source and sink, given that at least one node on the primary path falls within the circle defining a patterned failure.

III.C. Details of Methodology

In Section IV, we discuss our evaluation of disjoint and braided multipaths via simulation. We implemented the idealized and localized constructions of disjoint and braided multipath in the *ns-2* simulator. For generating reinforcements, our simulations considered the time of arrival of copies of a message from different neighbors. The most-preferred neighbor (Section II.D) was the one from whom a given event was heard first. This heuristic attempts to pick the lowest latency path. This may not always corresponding to the shortest-hop path, because of MAC effects. Message exchange in our localized constructions was simulated over the 802.11-like MAC available in *ns-2*. We're reasonably certain, based on our findings in Section IV, that this choice of MAC does not unduly distort our *comparison* of these mechanisms.

Before describing our experimental methodology, we list our parameter space. All our experiments were conducted by uniformly distributing a number of sensor nodes on a finite plane of dimension 400 meters square. The other parameter that we held fixed was node transmission radius: 40 meters. The parameters we varied, in order to assess their impact on the performance of disjoint and braided multipath were: density (more specifically, we varied the number of nodes within the plane), the spatial separation between source and sink (represented by the length of the shortest-hop path between the two), the failure probability for isolated failures p_i, the arrival rate of patterned failures λ_p, and the radius of patterned failures R_p.

Each run of our experiment corresponded to one choice of number of nodes N and spatial separation between source and sink d. In each run, we randomly selected a large number of source-sink pairs separated by d hops. For each source-sink pair, we computed four multipaths: idealized disjoint, localized disjoint, idealized braided and localized braided. For each multipath, we computed the maintenance overhead as defined in Section III.B.

To compute a multipath's resilience to isolated failures, we repeated the following set of steps a large number of times:

- Fail each node on the multipath with probability p_i.
- If a node on the primary path has failed, then, the assign a value of 1 to this set if at least one alternate path is available, 0 otherwise.

The resilience of the multipath to isolated failures is the average value assigned to sets in which at least one node in the primary path fails. The number of runs of the experiments, and the number of sets in each run were adjusted to obtain acceptable 95% confidence intervals.

To compute a multipath's resilience to patterned failures, we repeated the following set of steps a large number of times:

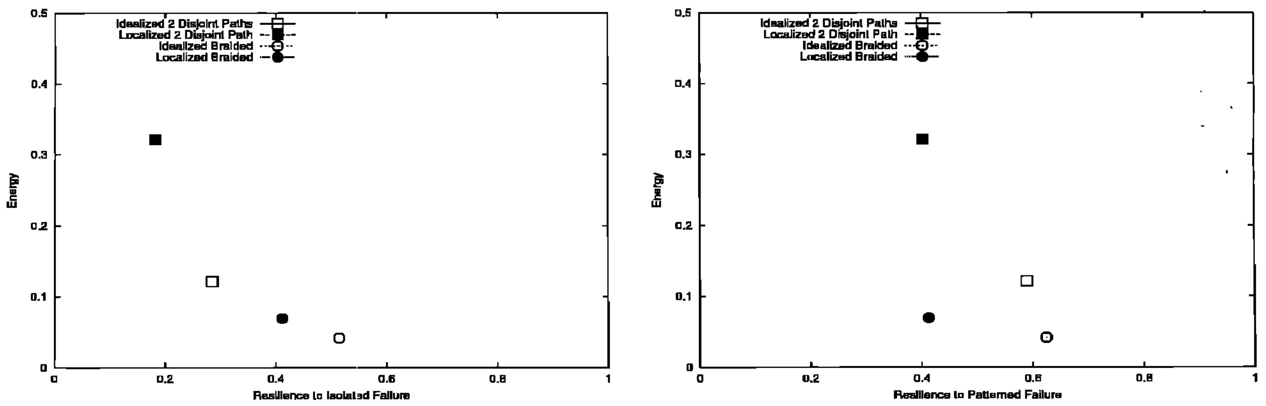
- Pick an integer n from a Poisson distribution with parameter λ_p.
- Randomly place n points on the plane.
- Fail all nodes within a radius R_p of each point in the plane.
- If a node on the primary path has failed, then, the assign a value of 1 to this set if at least one alternate path is available, 0 otherwise.

The resilience of the multipath to patterned failures is the average value assigned to sets in which at least one node in the primary path fails. The number of runs of the experiments, and the number of sets in each run were adjusted to obtain acceptable 95% confidence intervals.

IV. Simulation Results

In performing these simulation experiments, our goal was to understand the energy/resilience tradeoff between our various multipath schemes. One simple instance of this tradeoff is illustrated in Figure 6. We see that for isolated failures, 2-disjoint idealized multipaths are significantly less resilient, and have higher maintenance overhead than idealized braided multipaths. For patterned failures, the idealized schemes have comparable resilience, but 2-disjoint has higher maintenance overhead. Similar distinctions exist for the localized mechanisms.

Clearly, Figure 6 does not represent the whole picture. In the following subsections, we carefully study the impact on each metric of varying different parameters. Our base experiments consider 2-disjoint multipaths. In a later section, we describe the performance of 3-disjoint multipaths.



(a) Isolated: 400 nodes, 6 hop source-sink separation, $p_i = 0.2$

(b) Patterned: 400 nodes, 6 hop source-sink separation, $\lambda_p = 3, R_p = 20$

Figure 6: Illustrating the energy vs resilience tradeoff

IV.A. Maintenance Overhead

Figure 7(a) plots the maintenance overhead as a function of the number of sensor nodes in the plane. The number of nodes is a measure of the deployment density, given that we fix the size of the plane, and the transmission radius. Overall, braided idealized multipaths require lower maintenance overhead than 2-disjoint idealized multipaths. At low densities, 2-disjoint idealized multipaths incur 3 times the maintenance overhead of idealized braided multipaths. At higher densities, the difference between the two decreases; as described in Section II.F, at higher densities, the disjoint alternate paths are comparable in length to the primary path.

The localized braided heuristic, at lower densities, has lower maintenance overhead than its idealized counterpart. In this regime, the localized braided construction often fails to route around a node on the primary path, instead rejoining the primary path immediately. This results in tightly localized braid, but with relatively poor resilience properties, as we shall see later. However, at higher densities, the localized braid closely tracks the idealized braid.

The localized disjoint construction also behaves curiously at low densities. 2-disjoint localized multipath has a lower maintenance overhead than its idealized counterpart. This is because, at low densities, our localized construction sometimes fails to find an alternate path, leading to lower average maintenance overhead. At higher densities, localized 2-disjoint incurs significantly higher energy than its idealized counterpart. Localized 2-disjoint finds significantly longer alternate paths than idealized 2-disjoint, for the reason explained in Section II.F.

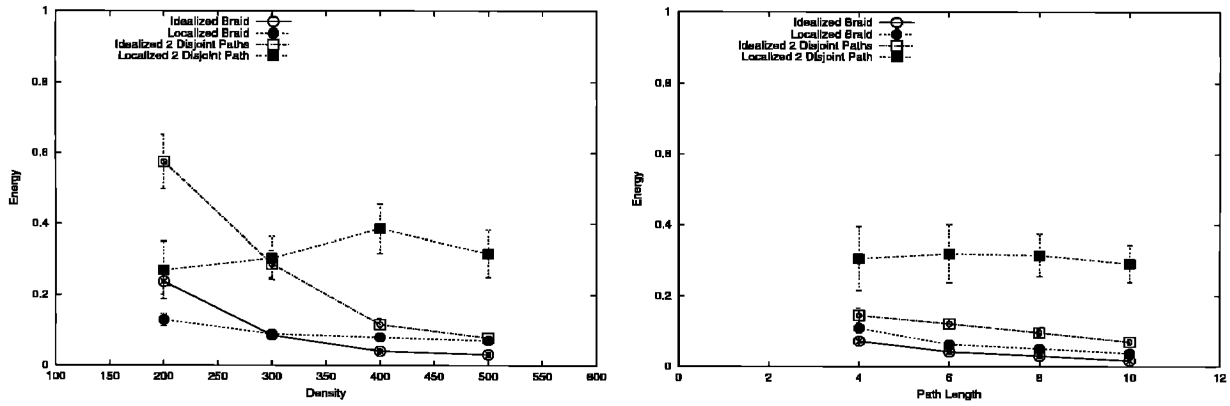
Finally, the maintenance overhead of localized 2-disjoint is nearly an order of magnitude higher than localized braid at high densities. We believe that local algorithms that try to achieve lowest latency disjoint paths face a fundamental problem: they do not have enough information to do so. In other words, we believe these results show that it might be easier to construct low-overhead braids than to construct low-overhead disjoint paths using localized algorithms.

Figure 7(b) shows the impact of source-sink separation on maintenance overhead. Most multipaths have slightly lower overhead at higher source-sink separations. We attribute this to the increased availability of alternate paths at higher separations.

IV.B. Resilience to Isolated Failures

Figure 8 describes the resilience of the different multipaths as a function of p_i, the probability of isolated node failure. In general, the idealized braid is more resilient that the idealized disjoint multipath. At low failure probabilities, it has about 20% higher resilience, and at higher probabilities, it has nearly twice the resilience of the idealized disjoint multipath.

A simple, though not necessarily complete, explanation of this difference is as follows. Consider a 2-disjoint idealized multipath in which the alternate path is of the same length as the primary path. The number of ways in which two nodes can simultaneously fail and sever a 2-disjoint idealized multipath is proportional to n^2, where n is the number of nodes on the primary or alternate path not including the source or sink. However, in the case of a "perfect" braid (one



(a) Density: 6-hop source-sink separation

(b) Source-sink separation: 400 nodes

Figure 7: The impact of density and source-sink separation on maintenance overhead

in which the path around a node has exactly one hop), this number is proportional to n.

Localized algorithms are slightly less resilient than their idealized counterparts. The reasons for these are as described in Section II.F; both the localized braid and the localized disjoint multipath can discover longer paths than their idealized counterparts. Notice that across the range of simulated failure probabilities the localized braid is between 50% and 2 times more resilient than the localized disjoint multipath.

Figure 9(a) shows the impact of source-sink separation on resilience of multipaths to isolated failure. Resilience decreases with increasing separation. This is predicted by our simple explanation above: as separation increases, n increases, as does the number of ways in which either the braid or the disjoint can be severed. Similarly, as density increases (Figure 9(b)), the lengths of the available alternate paths decrease, resulting in fewer ways for severing the multipath and consequently increased resilience.

IV.C. Resilience to Patterned Failures

Figure 10(a) shows the variation of resilience to patterned failure to source-sink separation. Interestingly enough, the resilience of the idealized braid is comparable to that of idealized 2-disjoint. Considering that the disjoint multipath expends significantly more energy, this is surprising. This suggests that 2-disjoint paths do not give adequate geographic spreading of paths to give high resilience. Indeed, using 3-disjoint paths does increase resilience to patterned failure, but at the cost of additional overhead (Section IV.D).

There is some, but small increase in resilience with increasing source-sink separation (Figure 10(a)). This is expected. As the multipath spreads geographically, one might expect that it becomes more resilient to pattern failure. However, the extent of the increase is small, and indicates that even at larger source-sink separations, the alternate paths are not more geographically spread out than at smaller separations.

The locally constructed schemes uniformly have about 25% lower resilience to patterned failures in Figure 10(a). This is a little surprising, considering that in an earlier section, we said that the localized mechanisms tend to find longer (and presumably more geographically spread out) alternate paths than their idealized counterparts. The explanation lies in the fact that localized mechanisms also tend to find longer *primary* paths than their idealized counterparts, which seems to more than negatively compensate for the increased resilience due to spreading.

With increasing density (Figure 10(a)), the resilience of the idealized schemes decreases because the alternate paths are spatially closer to the primary path. The localized schemes *increase* with density for a different reason. At low densities, localized disjoint doesn't find an alternate path, and localized braided sometimes isn't successful in routing around a node. These effects decrease with increasing density, resulting in higher resilience at higher densities.

With increasing frequency of failure, or radius of failure, one would expect the resilience to decrease. Indeed, it does (Figure 11), although the impact of radius is more dramatic.

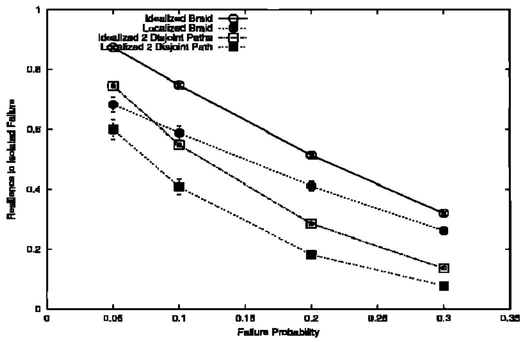
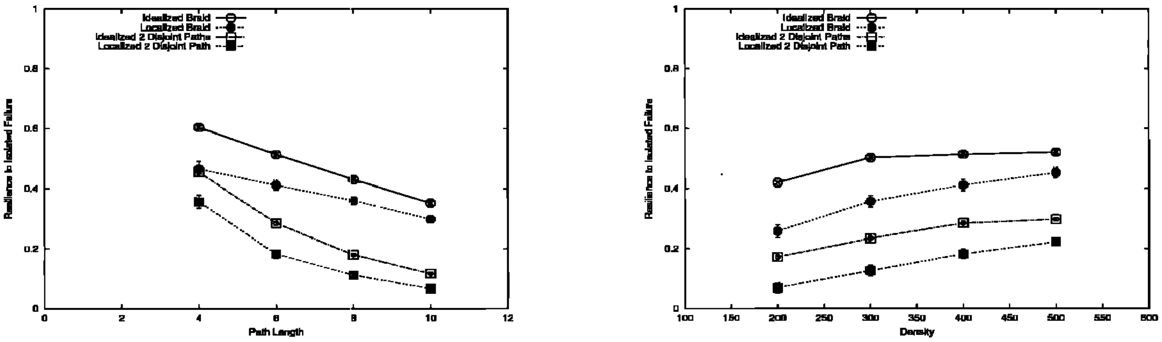


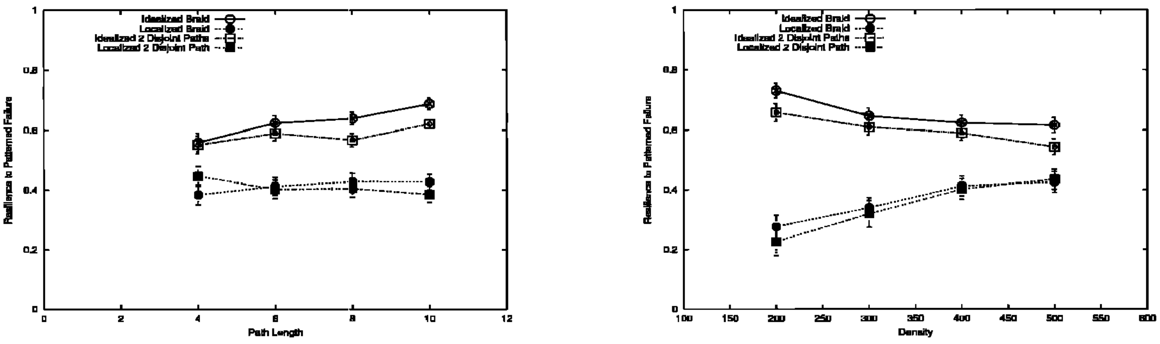
Figure 8: Impact of failure probability on resilience: 400 nodes, 6-hop source-sink separation



(a) Source-sink separation: 400 nodes, $p_i = 0.2$

(b) Density: 6-hop source-sink separation, $p_i = 0.2$

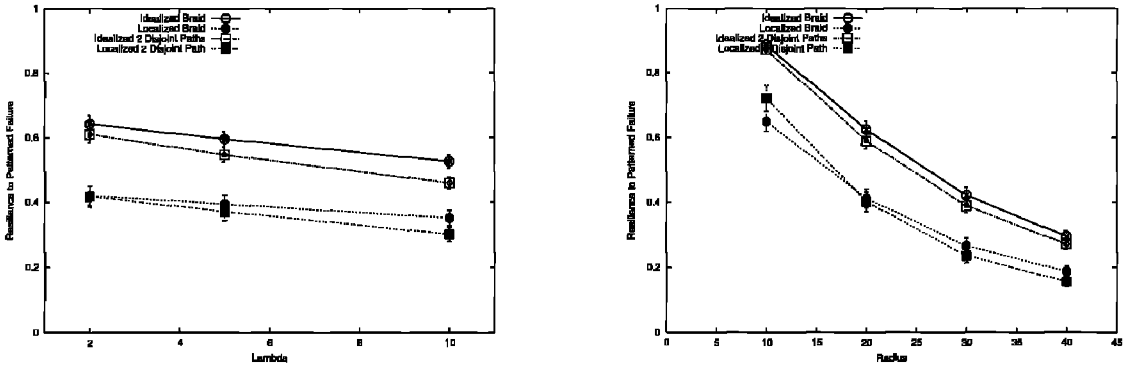
Figure 9: The impact of density and source-sink separation on resilience to isolated failure



(a) Source-sink separation: 400 nodes,$\lambda_p = 3$,$R_p = 20$

(b) Density: 6-hop source-sink separation, $\lambda_p = 3$,$R_p = 20$

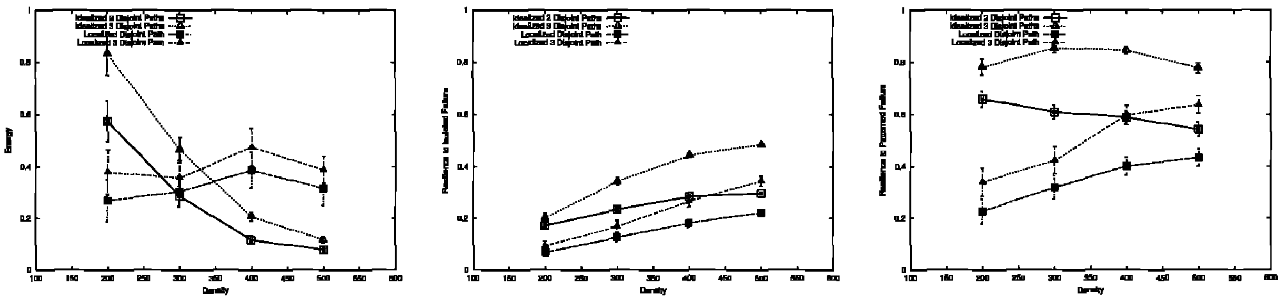
Figure 10: The impact of density and source-sink separation on resilience to patterned failure



(a) Frequency of patterned failure: 400 nodes, 6-hop src-sink separation, $R_p = 20$

(b) Radius of patterned failure: 400 nodes, 6-hop src-sink separation, $\lambda_p = 3$

Figure 11: The impact of λ_p and R_p on resilience to patterned failure



(a) Maintenance Overhead: 6 hop source-sink separation

(b) Resilience to Isolated Failure: 6 hop source-sink separation, $p_i = 0.2$

(c) Resilience to Patterned Failure: 6-hop src-sink separation, $\lambda_p = 3$, $R_p = 20$

Figure 12: Impact of increased disjointedness

IV.D. Sensitivity to Increasing Disjointedness

Finally, we consider the energy/resilience tradeoff of increasing the level of disjointedness from 2 to 3. Figure 12 shows that we get modest improvements (about 25% for patterned and about 40% for isolated failures) in resilience with approximately 30% increase in maintenance overhead. Thus, with the expenditure of more energy one can improve the performance of disjoint paths, but this improvement isn't without its cost. That is, it is not that the resilience of disjoint paths is increased dramatically by only slightly increasing the overhead.

V. Related Work

The literature on multipath routing is vast and we do not attempt to be comprehensive in this summary of related work. To our knowledge, however, ours is the first attempt to evaluate the energy/resilience tradeoff for multipath routing in wireless sensors.

Classical multipath routing has focused on the use of multipath primarily for load balancing and fault tolerance. Proposals for load balancing include circumventing capacity constraints of a single path by aggregating traffic sent on multiple paths [14] and reducing route oscillations and congestion by routing traffic through less congested network areas [15]. In sensor networks, such schemes can distribute energy usage among nodes in the network as means to increase network lifetime. However, this is not the focus of the work presented in this paper.

Multipath routing for fault tolerance has been studied both in the networking literature (ATM [6], OSPF [9]), and in the design of high-speed networks [8], as well as in wireless networks. We describe some related work in these areas in the next few paragraphs.

Resilience to failures in networks has generally dealt with computation (using graph algorithms) and the establishment multiple node-disjoint or edge-disjoint paths [6]. These schemes ensure resilience to at least k-failures by constructing k-disjoint paths between the source and the destination. The key challenge to constructing multiple paths using routing tables, is to provide fast convergence to an efficient set of k-disjoint paths between a set of sources and destinations in the network. Unlike the work presented in this paper, these schemes either assume global information, or require proactive routing information [16] exchange in order to compute the disjoint multipaths.

Application-level dispersity routing [2] proposes reserving multiple paths in the network for fault-tolerance in real-time networks. Nodes send redundant or erasure correcting information through some of the paths and use heirarchical compression techniques for graceful degradation to failure. The tradeoff between the utilization of network capacity and resilience to link failure forms the primary focus. This work is complementary to ours, in that its focus is on redundant data delivery to ensure reliability.

Variants of dispersity routing have proposed relaxing the disjointedness requirement [1]. These variants consider alternate partially disjoint paths where links in the network are restricted to belong to only a subset of these paths. This differs from our braided multipath in two ways. First, knowledge of the underlying topology is assumed, which enables the use of globally optimal mechanisms to construct alternate paths. Unlike braided multipath, this allows the design of non-disjoint multipaths with predictable resilience. Second, there is no cost to maintaining alternate paths, and energy-efficiency or maintenance overhead is not a concern.

Alternate path routing schemes in ad-hoc networks have been investigated, although less extensively. TORA [11] provides multipath by maintaining a destination-oriented DAG for each node in the network, very much like network gradients in directed diffusion. The protocol, however, incurs significant overhead maintaining the DAG in the network. To our knowledge, no work has attempted to evaluate this overhead and its impact on energy-efficiency and resilience. Multipath extensions to DSR [10] support the construction of "good" alternate paths using source routing mechanisms. This work uses disjoint paths from intermediate nodes on the primary path to enhance resilience. The use of source routing, which distinguishes their work from ours, enables the selection of low-latency alternate disjoint paths. While source routing is certainly worth considering for sensor networks, it is unclear how it can be applied to data-centric routing (Section II.A).

To our knowledge, existing literature has not considered patterned failure models either in the context of multipath routing or otherwise. Dispersity routing schemes do consider resilience to isolated failures, expressed in terms of the number of independent faults that can be tolerated by the multipath.

VI. Conclusions

This paper describes the use of multipath routing for energy-efficient recovery from node failures in wireless sensor networks. When a small number of multipaths are kept alive, failures on the primary path can usually be recovered from without invoking network-wide flooding for path discovery. This feature is important in sensor networks since flooding can reduce network lifetimes. We propose and evaluate two kinds of multipath designs: the classical node-disjoint multipath, and a novel braided multipath that consists of partially disjoint alternate paths. We study the energy/resilience tradeoffs of these mechanisms both for independent and geographically-correlated failures. Our exploration of the parameter space gives us a richer understanding of these mechanisms.

Some of the interesting findings of the study are:

- For a disjoint multipath configuration whose patterned failure resilience is comparable to that of braided multipaths, the braided multipaths have about 50% higher resilience to isolated failures and a third of the overhead for alternate path maintenance.
- We believe that it is harder to design localized energy-efficient mechanisms for constructing disjoint alternate paths, because the localized algorithms lack the information to find low latency disjoint paths.
- Finally, increasing the number of disjoint paths does increase the resilience of disjoint multipaths but with a proportionately higher energy cost. It is not the case that a small energy expenditure dramatically improves the resilience of disjoint paths.

For future work, we will consider two new directions. First, we intend to explore other forms of braiding and understand where on the energy/resilience tradeoff spectrum these lie. Second, we will consider extending some of these schemes to multiple source and sinks.

References

[1] Anindo Banerjea. A Taxonomy of Dispersity Routing Schemes for Fault Tolerant Real-Time Channels. In *Proceedings of ECMAST*, volume 26, pages 129–148, Louvain-la-Neuve, Belgium, May 1996.

[2] Anindo Banerjea. Simulation Study of the Capacity Effects of Dispersity Routing for Fault Tolerant Real-Time Channels. In *ACM Computer Communications Review*, volume 26, pages 194–205. ACM Press, October 1996.

[3] Deborah Estrin, Ramesh Govindan, John Heidemann, and Satish Kumar. Scalable coordination in sensor networks. In *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 99)*, pages 263–270, N.Y., aug " 15–20" 1999. ACM Press.

[4] The Bluetooth Special Interest Group. Bluetooth v1.0B Specification. http://www.bluetooth.com, 1999.

[5] Chalermek Intanagonwiwat, Ramesh Govindan, and Deborah Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *Proc. ACM Mobicom*, Boston, MA, 2000.

[6] K. Ishida, Y. Kakuda, and T.Kikuno. A routing protocol for finding two node-disjoint paths in computer networks. In *International Conference on Network Protocols*, pages 340 – 347, Nov 1992.

[7] Esler M., Hightower J., Anderson T., and Borriello G. Next century challenges: Data-centric networking for invisible computing: The portolano project at the university of washington. In *Proc. ACM Mobicom*, 2000.

[8] N.F. Maxemchuk. Dispersity routing in high-speed networks. In *Computer Networks and ISDN Systems*, volume 25, pages 645–661, Jan 1993.

[9] J. Moy. The ospf specification. In *RFC 1131*, Oct 1989.

[10] Asis Nasipuri and Samir R. Das. On-Demand Multipath Routing for Mobile Ad Hoc Networks. In *Proceedings of the 8th Int. Conf. on Computer Communications and Networks (IC3N)*, Boston, MA, 1999.

[11] V. D. Park and M. S. Corson. A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks. In *Proceedings of INFOCOM 97*, pages 1405–1413, April 1997.

[12] G. Pottie and W. Kaiser. Wireless Sensor Networks. *Communications of the ACM*, 43(5):51–58, May 2000.

[13] G. Pottie, W. Kaiser, L. Clare, and H. Marcy. Wireless Integrated Network Sensors. submitted for publication, 1998.

[14] H. Suzuki and F.A. Tobagi. Fast bandwidth reservation scheme with multi-link and multi-path routing in atm networks. In *Proceedings of the IEEE Infocom*, 1992.

[15] D. Zappala. Alternate path routing for multicast. In *Proceedings of the IEEE Infocom*, March 2000.

[16] W.T. Zaumen and J.J. Garcia-Luna-Aceves. Shortest multipath routing using generalized diffusing computations. In *Proceedings of the IEEE Infocom*, March 1998.