SICS Technical Report T2012:08
ISSN: 1100-3154

Title: Electronically-switched Directional Antennas for Low-power Wireless Networks: A Prototype-driven Evaluation

Authors: Luca Mottola, Thiemo Voigt, Gian Pietro Picco, Antonio Quartulli, and Fredrik Österlind

Date: September 17, 2012

Electronically-switched Directional Antennas for Low-power Wireless Networks: A Prototype-driven Evaluation

Luca Mottola[†‡], Thiemo Voigt[‡], Gian Pietro Picco[*],
Antonio Quartulli[*], and Fredrik Österlind[‡]
†Politecnico di Milano (Italy)
‡Swedish Institute of Computer Science
*University of Trento (Italy)

Abstract

We study the benefits of electronically-switched directional antennas in low-power wireless networks. This antenna technology may improve energy efficiency by increasing the communication range and by alleviating contention in directions other than the destination, but in principle requires a dedicated network stack. Unlike most existing works, we start by characterizing a *real-world* antenna prototype, and apply this to an *existing* low-power wireless stack, which we adapt with minimal changes. Our results show that: *i)* the combination of a low-cost directional antenna and a conventional network stack already brings significant performance improvements, *e.g.*, nearly halving the radio-on time per delivered packet; *ii)* the margin of improvement available to alternative clean-slate protocol designs is similarly large and concentrated in the control rather than the data plane; *iii)* by artificially modifying our antenna's link-layer model, we can point at further potential benefits opened by different antenna designs.

1 Introduction

Electronically-switched directional (ESD) antennas are capable of *dynamically* conveying the radiated power along given directions, as opposed to the isotropic propagation of omni-directional ones. Therefore, they may alleviate wireless contention by not involving nodes other than the destination, and increase communication range essentially at no additional energy cost. This technology is appealing for low-power wireless, where energy consumption is key, but in principle requires a dedicated protocol stack to leverage its dynamic abilities. As we discuss next, only a few existing works tackle this problem. Thus, the question whether one can harvest these benefits as of today—*i.e.*, in a *complete* system immediately usable for *real* applications—is still open.

State of the art. Dai et al. [8] survey the application of directional antennas to wireless networks in general. Particularly, the literature investigating the use of ESD antennas in low-power wireless can be roughly divided in two categories, focusing either: *i)* on the design of antenna prototypes; or *ii)* on clean-slate design of network protocols.

The works in the first category were instrumental in concretely demonstrating the viability of ESD antennas in low-power wireless, showing that current technology can meet the requirements on form factor and energy consumption. However, these works rarely assess the impact of the prototypes on the network stack. Specifically, Giorgetti et al. [13] assess the improvements in link performance with a prototype designed by combining four patch antennas. Viani et al. [26] present a design based on parasitic elements, and analyze its ability to reduce interference and to support localization. Parasitic elements are a common choice to reduce cost and size [5] used also by Nilsson [19] in the SPIDA prototype, later assessed based on common metrics for low-power wireless [20].

In the second category, protocols are often designed based on idealized antenna models defined purely by geometrical properties. For example, Felemban et al. [10] propose a clean-slate protocol stack that relies solely on directional communications, emphasizing neighbor discovery based on their previous work [11] and a dedicated MAC protocol. Although their stack may, in principle, be implemented atop ESD antennas, their simulation results do not translate immediately to a real antenna. Other works focus on specific network services considered in isolation, *e.g.*, neighbor discovery [25] and MAC [10], only seldom validating the findings on real prototypes. Similar validations are more common in works for mobile ad hoc networks, *e.g.*, by Choudhury et al. [5, 6].

Contribution and roadmap. In contrast with the state of the art, our goal is to assess the *readily available* benefits of using a real-world ESD antenna prototype with *existing* protocol stacks for low-power wireless.

The prototype we use is the SPIDA antenna mentioned earlier, whose features we summarize in Section 2. In the same section, we derive an empirical link-layer model for SPIDA, allowing us to analyze through simulation large-scale settings that would be otherwise difficult to reproduce in a real deployment, due to limited availability of antenna prototypes.

As for the protocol stack, we examine the staple functionality of multi-hop data collection in sensor networks as a representative application of low-power wireless. Sensor networks are indeed an area where the potential benefits of ESD antennas are likely to be most significant. We specifically consider Collect [15], the reference tree-based collection protocol in the Contiki operating system.

When used *as is*, namely with the antennas in omni-directional mode, the stack serves as a yardstick to measure the improvements brought by ESD antennas. To evaluate the latter, we adapt Collect with minimal modifications to support a form of *directional packet forwarding* (DPF) from a node to its parent in the tree. The rest of the routing protocol (*e.g.*, tree maintenance) and of the stack (*e.g.*, MAC) is unaltered, and relies on omni-directional communication.

We illustrate the details of the protocols in Section 3.

The simple modifications above are only one of the possibilities, and are biased by the fact that the routing tree is still built through omni-directional communication. As in this case the antenna range is shorter than in directional mode, the resulting tree topology is less efficient than what could achieved in principle. An increase in communication range may indeed correspond in fewer hops to traverse to reach the destination, reducing the number of transmissions.

To understand the impact of a dedicated strategy to build the routing tree with directional transmissions, we also compare our modified stack against a DPF schema operating on an optimal topology, computed offline by taking into account the increased communication range of SPIDA. Section 4 describes the analytical formulation of the corresponding routing problem and its optimal solution.

The methodology we use in our evaluation is described in Section 5, whereas we present experimental results in Section 6. The main findings, constituting the core contribution of the paper, are that: *i)* the combination of a low-cost directional antenna and a conventional network stack brings significant performance improvements, *e.g.*, nearly halving the radio-on time per delivered packet; *ii)* the margin of improvement available to alternative clean-slate protocol designs is similarly large and concentrated in the control rather than the data plane; *iii)* by artificially our antenna's link-layer model, we can point at further potential benefits opened by different antenna designs.

Section 7 ends the paper with brief concluding remarks.

2 Antenna Prototype and Link-layer Model

Our stepping stone is a real-world ESD antenna prototype we use to derive an empirical link-layer model. In Section 6, we use this model to assess the impact of using this antenna technology in low-power wireless.

2.1 Antenna Prototype

Most radio chips for low-power wireless networking operate in the 2.4 GHz ISM band. This, together with the requirement of small form factor—*e.g.*, to address the requirements of sensor network applications—limits the choice of antenna technology in this context [19].

We use a *switched parasitic element* antenna [24] called SPIDA, designed by Nilsson [19]. It consists of a central active element surrounded by "parasitic" elements, as shown in Figure 1. The central element is a quarter-wavelength whip antenna, *i.e.*, a traditional omni-directional antenna. The parasitic elements can be switched between ground and isolation: when grounded, they work as reflectors of radiated power; when isolated, they act as directors of radiated power. SPIDA has six parasitic elements, yielding six possible "switches" to control the shape and direction of the antenna main lobe. The parasitic elements are individually controllable: when all isolated, SPIDA behaves as an omni-directional

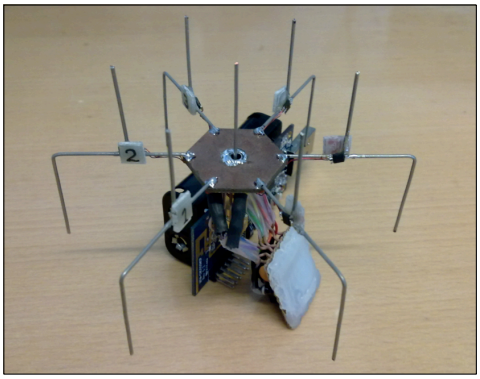


Figure 1: SPIDA prototype, connected to a TMote Sky [21] node.

antenna, which simplifies broadcasting and neighbor discovery.

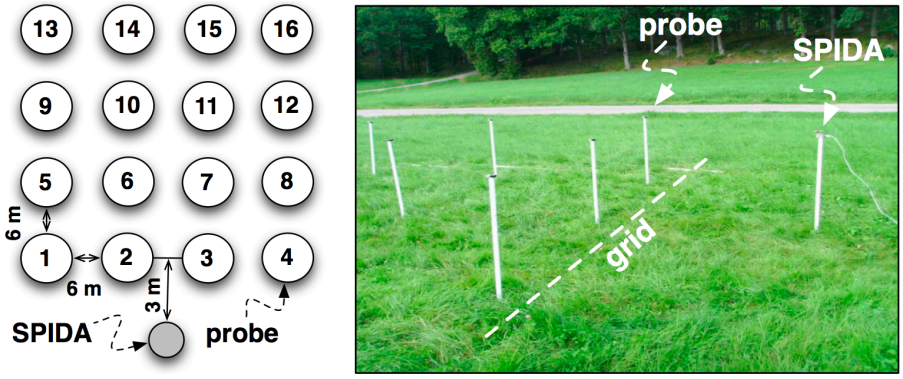
The antenna gain is designed to smoothly vary as an offset circle from approximately 7 dB to -4 dB in the horizontal plane, with the highest gain in the direction of the isolated element(s). The antenna is also straightforward to manufacture, and its most expensive part is the SMA connector (about $6 in single quantities). The cost, size, and radiation characteristics of SPIDA are therefore comparable with the state of the art in directional antennas for low-power wireless [5, 13, 26], rendering our results of general applicability.

2.2 Link-layer Model

We present the derivation of our empirical link-layer model and the corresponding experimental validation.

Modeling. We setup a test network with a SPIDA-equipped TMote Sky in front of a 4x4 grid of standard TMote Sky [21], as shown in Figure 2(a). SPIDA has only one parasitic element isolated, directed towards the center of the grid. The standard TMote Sky act as probes, uniformly sampling the environment where SPIDA radiates the highest power. We already verified that the packet delivery rate (PDR) is null in other directions [20]. We deploy the nodes in an open grass field, atop 1 m tall cardboard pillars to avoid signal reflections from the ground, as shown in Figure 2(b). Distances and transmission power are set to find a compromise between logistic issues and spatial accuracy. We also verify that the location has no interference from other networks in the ISM band. Before the experiments, we check that all probes do not exhibit significant drifts in the $RSSI$ readings among each other when in comparable conditions.

In every experiment, the SPIDA node starts by broadcasting a `start` packet at the highest transmission power to synchronize the probes and inform them on the expected duration of the run. Then, it switches to a lower transmission power and transmits 1000 test packets with an inter-packet interval (IPI) of 500 ms. This makes each packet independent of each other [22], avoiding the bias due to bursts of packet losses. The probes log the received packets, along



(a) Network layout. (b) Experiment environment.

Figure 2: Network layout and environment for experiments.

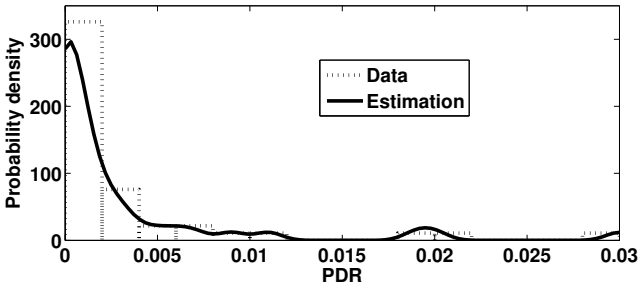
with their *RSSI* as indicated by the radio chip. At the end of the run, they report back to the SPIDA node the average *PDR* and *RSSI*. We repeat such experiment about 50 times in highly varying environmental conditions.

Based on the data above, we derive a spatio-temporal model of *Packet Delivery Rate* (*PDR*) and *Received Signal Strength Indicator* (*RSSI*) for SPIDA, borrowing from Cerpa et al. [4]. In contrast to their work, however, our model also considers the directionality of the antenna. This requires two independent variables describing coordinates in a plane, rather than only distance from the source. We model the average *PDR* and *RSSI* measured by a probe at the end of a batch as instances of a random variable. We then apply kernel density estimation to identify the corresponding probability density function (PDF). This method is particularly accurate in absence of information on the underlying probability distribution. The data we obtain grants a 95% confidence interval.

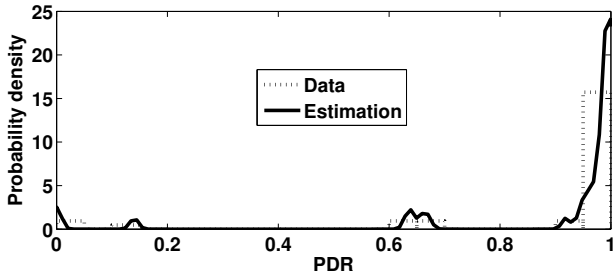
To exemplify the results of our modelling, Figure 3 depicts two example PDFs for *PDR*, corresponding to different probes. Figure 3(a) describes a probe outside the main antenna lobe. The PDF shows a single maximum for low values of *PDR*, although some packets are still occasionally received. We show in Figure 3(b) the PDF for *PDR* at a probe in the middle of the main antenna lobe: the situation is opposite, as the PDF shows a maximum for high values of *PDR*. We also obtain PDFs for nodes in a gray area at the boundary of the main antenna lobe, where the PDF exhibits multiple peaks. We obtain similar results for *RSSI*.

By discretizing the PDF curves, we generate probability tables with arbitrary granularity that associate given probability densities with specific values of *PDR* and *RSSI*. We use these tables to obtain the corresponding empirical cumulative distribution functions (ECDFs). Based on these, we apply inverse transform sampling to generate new random values with the same statistical trends as the original data.

Figure 4 shows the region of space with average $PDR > 10\%$ obtained from the empirical model, with SPIDA isolating only the element aligned with the



(a) Outside SPIDA main lobe with low PDR.



(b) Inside SPIDA main lobe with high PDR.

Figure 3: PDF at different probe nodes.

vertical axis and pointing upwards. Compared to the omni-directional setting, the area with $PDR > 10\%$ becomes slightly narrower and is offset in the direction of the isolated element, verifying the original design [19]. The figure also provides a basis to point at different antenna designs, as we discuss in Section 6.3.

Validation. We validate our model to assess: *i)* the extent the model adheres to the real-world behavior beyond the experimental data it is derived from; and *ii)*

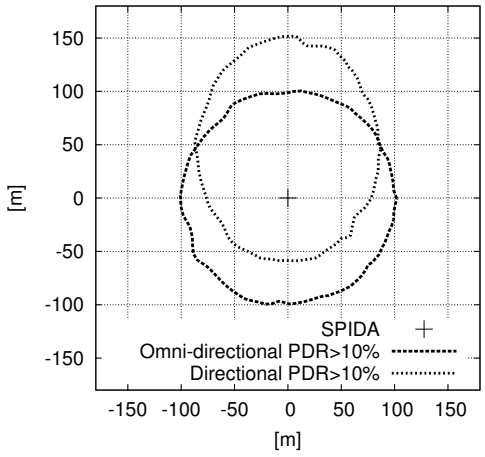
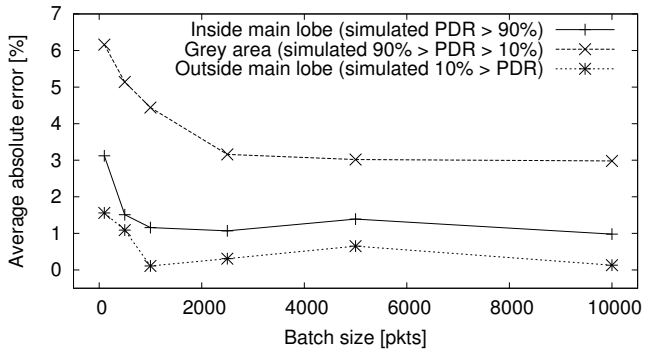
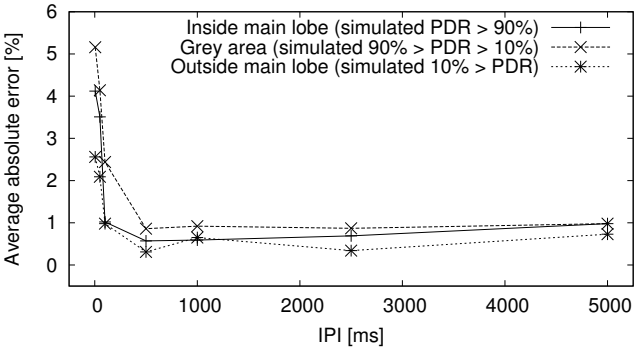


Figure 4: Region of space with $PDR > 10\%$. from the empirical model.



(a) Simulation vs. real experiments against different packet batches.



(b) Simulation vs. real experiments against different IPIs.

Figure 5: Average absolute error simulation vs. real experiments.

how general is our choice of batch size and IPI during the original experiments.

We gather further real-world data we compare with data generated by the implementation of our model in the Cooja [9] simulator. The validation data is gathered at the same place as the original data, but with a random placement of the probes and under very different temperature and humidity conditions. We let the SPIDA node send another one million packets at different IPIs. We replicate the same setup and positioning in Cooja and simulate the experiment. We repeat this about 50 times by varying positions of the probes. We then split the data in batches of different sizes.

Figure 5 compares the *PDR* obtained in simulation against the real experiment. The results show that our model is accurate also against the validation data. The worst case error is slightly above 6% and holds for nodes with neither very strong or very weak links, as the statistical variability is higher for this kind of links [23]. Moreover, the error is generally higher for smaller packet batches and smaller IPIs. This is expected because: *i)* when the IPI is sufficiently small, packet losses become dependent [23]; and *ii)* with too few packets, the result are no longer statistically significant.

3 Protocols

We describe the protocols we consider to understand the immediate benefits of ESD antennas, along with the changes we apply to enable directional packet forwarding (DPF).

Base protocol stack. We define an OMNI protocol configuration that exclusively uses omni-directional transmissions. To this end, we use Contiki's default tree-based data collection protocol, Collect [15]. The protocol is very similar to the Collection Tree Protocol (CTP) [14] in the TinyOS distribution. Several real-world deployments are based on these protocols or variations thereof [12].

Like most data collection protocols, Collect includes two core functionality: *i)* building and maintaining multi-hop routes in a tree-shaped topology; and *ii)* forwarding application data up the routing tree. For the former, Collect relies on ETX [7] as a routing metric. ETX measures the number of 1-hop transmissions to deliver a packet until receiving an explicit acknowledgment. Collect builds routes to minimize the total ETX along the path from sources to the data sink. Although in newer Contiki versions the ETX is estimated using explicit unicast messages [15], in our version nodes broadcast these advertisements as in CTP.

We use Contiki's default low-power MAC protocol, ContikiMAC. Similar to X-MAC [1], senders transmit small probe packets containing the receiver's identifier until the latter wakes up and acknowledges the strobe packet, whereupon the sender transmits the data packet.

Directional packet forwarding (DPF). Our goal is to explore the advantages we can readily harvest with *minimal* changes to the OMNI stack. Therefore, we modify only the forwarding of data to the tree parent to leverage directional transmissions. Exploiting the latter also for tree building and maintenance is considerably more complex, as we further discuss in Section 5. We devise three simple schemes:

- BLIND forwarding entails quickly transmitting every packet in all six SPIDA directions, in sequence. As the packet remains in the radio buffer while switching direction, subsequent transmissions happen very fast: no change to the duty cycle of ContikiMAC is required to accommodate these sequential transmissions. Implementing BLIND requires only 4 lines of C code in Collect.
- NARROW forwarding randomly selects only one direction. Should the receiver send an acknowledgement back, the sender continues sending using the same direction, until either an acknowledgment is lost or the parent changes. When so, it re-starts from another random direction. NARROW requires changes to 12 lines of code.
- SMART forwarding searches for a working configuration by quickly sweeping all directions, as in BLIND. However, to detect which direction succeeded, it appends the identifier of the active one to the packet, encoded with a corresponding number of padding bytes. Packets can therefore be transmitted without re-loading the radio buffer, since only the total packet

size is changed, but not the content. SMART requires changes to 16 lines of code.

In all these schemes, we change only the *data* forwarding mechanism, therefore *i)* the MAC protocol is unaltered, including the strobing occurring in omni-directional mode; and *ii)* the changes are restricted to the sender side: packet receptions always occur with SPIDA in omni-directional mode.

4 Routing Problem and Optimal Solution

We formulate the routing problem as an integer linear program (ILP), which we use later to compute the optimal configuration of the routing tree by exploiting directional transmissions. Our approach builds upon the multi-commodity problem [27], a formulation already applied in wireless networks [16], including low-power ones [18].

We consider a directed graph, *e.g.*, representing a transportation network, with node set $\mathcal{N}$ and arc set $\mathcal{A}$, and a set of commodities $\mathcal{C}$, *e.g.*, goods. The goal is to route each commodity $k \in \mathcal{C}$ from a set of origins $O(k) \subseteq \mathcal{N}$ to a set of destinations $D(k) \subseteq \mathcal{N}$ by minimizing a given metric.

System model. We model a wireless network as a directed graph where $\mathcal{N}$ is composed of the wireless nodes, commodities in $\mathcal{C}$ are packets flowing from sources to destinations, and $\mathcal{A}$ contains an arc (i, j) if a packet sent by node i has non-zero probability of being received at node j. Unlike existing works [16, 18], we do not model this notion of communication with a boolean value indicating whether communication is possible. Instead, each link has an associated weight $w_{i,j}$ representing the quality of the link based on its *PDR*.

We empirically derive *PDR* information between any two nodes with our empirical model. A simple discovery protocol instructs each node i to send 10,000 broadcast messages *for each direction* the SPIDA allows by isolating a single parasitic element (*i.e.*, 60,000 messages in total). All other nodes j log the messages received. If communication between two nodes is possible through different directions, the highest *PDR* is assigned as a weight to the link and the corresponding antenna configuration recorded with the link (i, j). Thus, we do not explicitly model the different SPIDA directions, rather we associate them to links, simplifying the modeling.

In our context, the destinations $D(k)$ always consist of a single node, the tree root d. Without loss of generality, we assume a commodity to flow from a single origin to destination [27]. Since commodities flowing from the same origin to the destination follow the same route, we can state a one-to-one mapping between the route connecting any pair $\langle o(k), d\rangle$, and any commodity k.

We model the fact that a link from i to j is used for routing from a given source k to the tree root d by using boolean variables attached to each link:

$$r_{i,j}^{k} = \begin{cases} 1 & \text{if route from source } k \text{ contains link } (i,j) \\ 0 & \text{otherwise} \end{cases}$$

Next, we define the objective function and the constraints to satisfy for any admissible solution.

Objective function. Collection protocols for low-power wireless often rely on ETX to build end-to-end estimates of a routing path's quality. ETX measures the number of 1-hop transmissions until a packet acknowledgment arrives from the receiver. To embed ETX information within our modeling, given the *PDR* information in the weights of links (i, j) and (j, i), we first compute the probability that a packet is sent n times on (i, j) before an acknowledgment is received on (j, i):

$$P(RTX = n)_{i,j} = (1 - PDR_{i,j}PDR_{j,i})^{(n-1)} PDR_{i,j}PDR_{j,i}$$

Let us define the probability of a successful bidirectional transmission $P_{TX} = PDR_{i,j}PDR_{j,i}$. The expected number of transmissions for a packet from i to j, *i.e.*, what ETX tries to measure, is then:

$$E(RTX)_{i,j} = \sum_{n=1}^{\infty} nP(RTX = n)_{i,j} = \frac{1}{P_{TX}}$$

The cost of a link from i to j in a route from source k is:

$$u_{i,j}^k = \begin{cases} E(RTX)_{i,j} = \frac{1}{P_{TX}} & \text{if } r_{i,j}^k = 1 \\ 0 & \text{otherwise} \end{cases}$$

The goal is then to minimize the overall expected number of transmissions for all links used by all routes, that is:

$$\mathit{TreeCost}(\mathcal{C}, \mathcal{A}) = \sum_{k \in \mathcal{C}, (i,j) \in \mathcal{A}} u_{i,j}^k$$

Optimal solution. We are to find the value assignment of $r_{i,j}^k$, $\forall k \in \mathcal{C}$, $\forall (i, j) \in \mathcal{A}$ such that $\mathit{TreeCost}(\mathcal{C}, \mathcal{A})$ is minimum. We derive the optimal solution to the problem using mathematical programming, which entails specifying the necessary constraints on the decision variables.

First, we require $r_{i,j}^k$ to be an integer binary variable. Second, we state that the assignment to $r_{i,j}^k$ must contain a connected end-to-end path for each route k. This constraint can be expressed by requiring every node different from the source $o(k)$ and the destination d to "preserve" packets. In other words, if a route "enters" node i, it must also exit from it, unless it is a source or the destination:

$$\forall i \in \mathcal{N}, \forall k \in \mathcal{C},$$
$$\textstyle\sum_{m:(i,m) \in \mathcal{A}} r_{i,m}^k - \sum_{n:(n,i) \in \mathcal{A}} r_{n,i}^k = \begin{cases} 1 & \text{if } i = o(k) \\ -1 & \text{if } i = d(k) \\ 0 & \text{otherwise} \end{cases}$$

Next, we briefly illustrate the methodology we apply, before discussing our experimental results.

5 Methodology

We describe the metrics, settings, and baselines for comparison. Throughout the study, we use the Cooja/MSPSim simulator [9]. MSPSim emulates the MSP430 MCU at instruction level, providing time-accurate executions.

Comparison baselines. Our primary goal is to compare the DPF protocols against the standard Omni stack, both described in Section 3. However, our DPF variants all rely on a routing tree built atop omni-directional communication, and disregard the ability to build and maintain the tree using directional communication. This could exploit the increased range of ESD antennas, reducing the route stretch and ultimately improving reliability and lifetime. The problem is open [8], and it is difficult to establish a term of comparison. Therefore, we opt for evaluating the trade-offs along two components: packet forwarding and control traffic.

As for the former, we compare the protocols against a DPF variant called DirTree, operating on the optimal routing topology given by the solution to the ILP problem in Section 4. Simulations with DirTree execute with this routing information hard-wired in the nodes. To make the executions of DirTree comparable with the others—which also account for the control traffic—we record the control traffic generated with Omni, and re-play it during DirTree simulations.

On the other hand, a real protocol exploiting directional communications would hopefully improve over the Omni baseline. For this reason, we also compare against the traffic generated solely by data forwarding. This assesses the efficiency of the routing topology alone, no matter how it is obtained. In a sense, these two extremes—an optimal directional tree with the same traffic as the standard omni-directional stack, and one built "for free"—define the margin of improvement available to clean-slate protocols based on directional communication.

Metrics. We consider two key performance metrics commonly used in studying low-power wireless protocols [14]: *i) packet delivery* at the sink, defined as the fraction of application packets successfully received at the sink over those sent; and *ii) radio duty cycle*, computed as the fraction of time a node keeps the radio on over the experiment duration. The former is an indication of the level of service provided to users in delivering application data, whereas the latter provides a measure of a protocol's energy efficiency.

To determine packet delivery at the sink, we embed sequence numbers within data packets. We measure the radio duty cycle in software, using Contiki's power profiler. For each setting and protocol analyzed, we compute these metrics based on at least 50 statistically independent runs and report network-wide averages, along with their standard deviation. Simulations lasts until the system reaches stable conditions, a situation we verify by imposing that all per-node standard deviations for all metrics are below 5% of the average value.

Settings. We explore different system dimensions, illustrated in Table 1 with their default values. The packet generation rates reflect the settings of real-world

Dimension	Values	Unit metric
Packet generation rate	1...**3**...10	packets/min
Number of nodes	100...**300**...500	nodes
Network density	4...**12**...20 (2.88)...**(8.64)**...(14.4)	nodes/unit square (neighbors in omni)

Table 1: System dimensions. Default values are in boldface.

sensor network deployments [2, 3, 12]. Instead, the number of nodes involved purposely exceeds the size of real installations, yielding challenging conditions for the protocols. As for network density, Table 1 reports the number of nodes per unit square, 100 m in our experiments. The actual network density, however, is a function of the antenna employed. To relate our values with the existing literature, the table reports also the average number of neighbors when using SPIDA in omni-directional mode, which is in line with existing deployments [12].

We use 80-byte packets for all experiments, again inspired by real deployments [12]. We start measuring after 10 minutes of simulated time to let the protocols converge to an initial routing topology. In ContikiMAC, we set the periodic wake-up interval to 8 Hz. We use the same default settings for all protocols' parameters.

We employ our empirical link-layer model, described in Section 2. For consistency of results, we simulate omni-directional transmissions still with our model as opposed to Cooja's built-in models, by isolating all parasitic elements of SPIDA. For each simulation run, we generate the network topologies randomly using Cooja's default mechanisms. The antenna orientation in the two-dimensional plane and the placement of the data sink are also randomly chosen and different for every run.

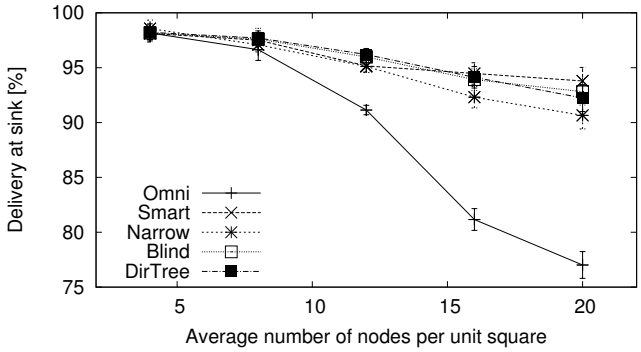
6 Evaluation

We assess the benefits of DPF and discuss where further improvements are available based on deeper modifications to the protocols or to the antenna design.

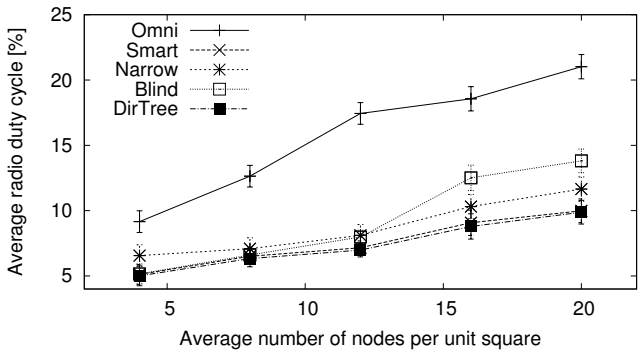
6.1 Assessing Directional Packet Forwarding

We evaluate our changes to Collect to enable DPF. Our results reveal that: *i)* DPF provides significant gains in packet delivery and radio duty-cycle compared to Omni; *ii)* these gains apply to a variety of settings, and are mainly due to the ability of reaching the parent with higher signal strength and, for Smart and Narrow, to better spatial utilization of the channel; and *iii)* DirTree, which additionally uses optimal routing topologies based on directional transmission, does not provide significant additional gains, except when the size of the network grows.

Effect of network density. Figure 6 depicts the performance of the protocols we test against the number of nodes per unit square. The DPF solutions show



(a) Average packet delivery: *All* DPF *solutions improve packet delivery for higher node densities.*



(b) Average radio duty-cyle. DPF *also reduces the radio duty cycle, saving energy.*

Figure 6: Performance against varying network density.

improved delivery at the sink, as shown in Figure 6(a). By steering the radiated power towards the parent, both SMART and NARROW generate less contention on the channel, yielding fewer packets lost in collisions. The benefits become more evident as more nodes are potentially in reach. NARROW performs slightly worse than other directional solutions as it takes more time to find another working direction upon changes in the link quality.

On the other hand, the good performance of BLIND in Figure 6(a) is somewhat unexpected, as it sweeps all antenna directions for *every* packet, supposedly causing more collisions—and therefore packet losses—than SMART and NARROW. Looking at the simulation logs, we verify that BLIND causes about 27% more collisions on average than OMNI. However, most of these collisions happen when using directions other than the one best for a parent, and are thus immaterial. Nevertheless, BLIND still ensures that the best direction to send data is eventually used. As a result, BLIND improves performance by about the same amount than SMART and NARROW, compared to OMNI.

In Figure 6(a) we also note that DIRTREE performs comparably to other directional solutions. In these scenarios and with the protocol stack we con-

sider, leveraging routes found with directional transmissions does not provide additional gains. We verified in the simulation logs that with increasing network density it becomes equally likely for parents found beyond the omni-directional range to be affected by collisions. This neutralizes the benefits of the increased communication range when finding parents, and even causes DirTree to perform slightly worse than other directional solutions at times.

Figure 6(b) shows the radio duty cycle in the same settings. The gains in packet delivery using DPF yield savings in the energy spent for delivering packets. Indeed, improved reliability at packet level entails fewer re-transmissions. Among the protocols in Section 3, Blind exhibits the worst performance. Blind always sweeps all directions for every packet, and thus it necessarily keeps the radio on slightly longer.

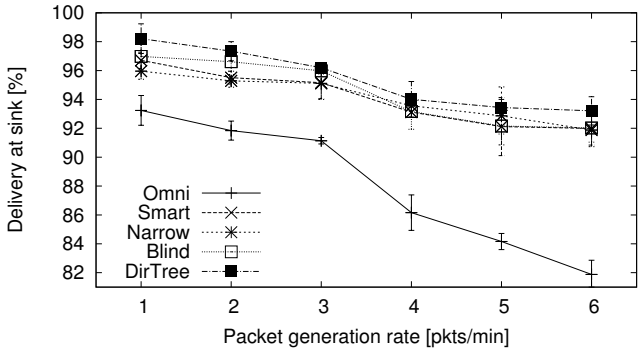
Effect of network traffic. The protocols' performance against increasing packet generation rates, reported in Figure 7, confirms the discussion above. Indeed, injecting more packets into the network ultimately causes channel contention similarly to increasing network density. Packet delivery improves with directional transmissions compared to Omni as the packet generation rate increases, as shown in Figure 7(a). DirTree, on the other hand, provides again only limited additional gains, due to collisions occurring in the region where omni-directional transmissions do not reach.

DPF also delivers more packets with smaller radio duty cycles, as reported in Figure 7(b), with Blind performing the worst for the reasons explained previously. It attains performance similar to or better than other directional solutions only at low traffic or in sparse networks, where not much contention exists. Here, Blind leverages the fact that the best direction for a parent is always eventually used. In the same chart, DirTree is indeed the lower bound, but again it improves over Smart only by 1.7% on average.

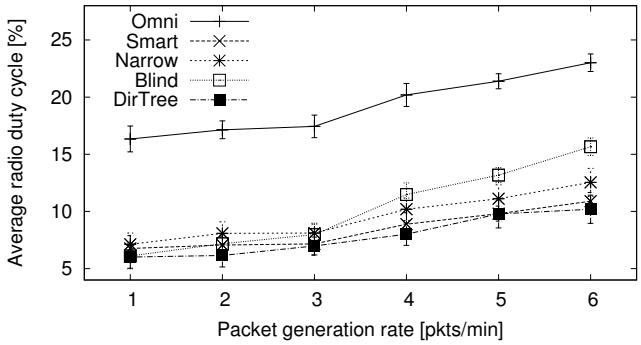
Figure 7(a) and 7(b) stop at 6 packets/min because we find all protocols unable to sustain higher traffic loads. Indeed, ContikiMAC reduces the bandwidth because of radio duty cycling: at 7 packets/min, all protocols deliver less than 40% of packets to the sink, rendering the results meaningless. To study higher traffic loads, we temporarily disable radio duty cycling in ContikiMAC, which leads to the performance in Figure 7(c). The trends in the chart mirror Figure 7(a), but on a different X- and Y-scale, demonstrating that the MAC protocol was the limiting factor. The conclusions on the influence of DPF therefore still apply, including the limited gains of DirTree that additionally leverages optimal routing topologies.

Effect of network size. Figure 8 illustrates the performance with increasing scale. As with the previous results, all other dimensions—network density and packet generation rate in this case—remain unaltered. Unlike with these two dimensions, however, increasing scale shows some advantages for DirTree besides the gains of DPF.

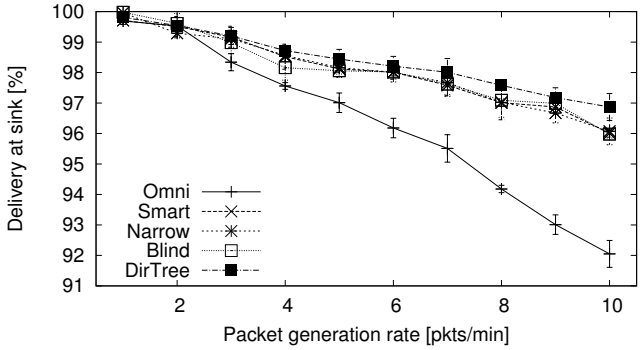
Figure 8(a) shows DirTree increasingly delivering more packets than all other solutions as the network size grows. It does so with progressively better performance in radio duty-cycling, as illustrated in Figure 8(b). This is because



(a) Average packet delivery. DPF *sustains high packet delivery for high traffic loads.*



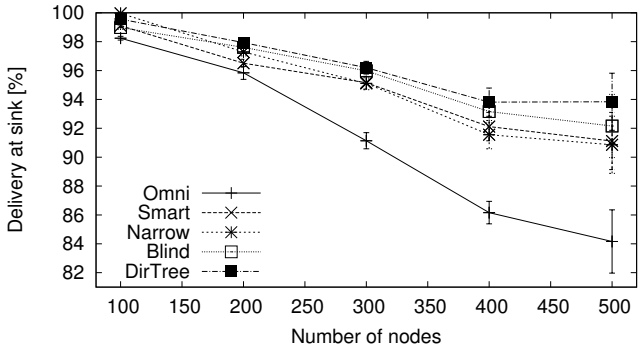
(b) Average radio duty-cycle. DPF *improves the radio duty-cycle at different traffic loads.*



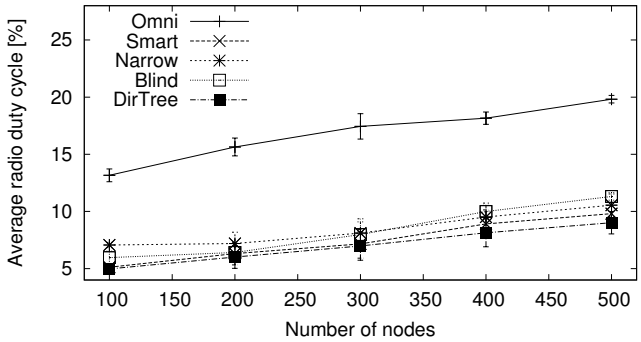
(c) Average packet delivery - no radio duty cycle. DPF *sustains high packet delivery even at very high traffic loads.*

Figure 7: Performance against varying packet rate. Note the different scales in the packet delivery charts.

the gains of DirTree are essentially "localized" in the single neighborhoods, and increasing the network size replicates such gains across more neighborhoods.



(a) Average packet delivery. DPF *sustains high packet delivery for large networks.* DirTree *shows additional improvements.*



(b) Average radio duty-cycle. DPF *keeps the radio duty-cycle low for large networks.*

Figure 8: Performance against varying number of nodes.

Differently, network density and packet generation rate mainly affect channel contention, a situation where building routes based on directional transmissions does not help more than using DPF alone.

Summary. These results exemplify the *readily available* benefits of employing ESD antennas in low-power wireless. The question is then whether further gains are available with deeper modifications to the stack or different antenna designs. We investigate the problem in the next two sections.

6.2 The Cost of Control Traffic

As we mention in Section 5, DirTree runs by re-playing the control traffic of Omni. This traffic is intimately tied to the specific stack employed, which uses ETX as link quality metric and a CSMA policy to access the radio medium. In the following, we factor out the control traffic of the routing protocol to understand the potential improvements with different routing strategies.

We re-run the simulation scenarios of Section 6.1 with two additional baselines: *i)* DirTree-NoControl works the same as DirTree *without* re-playing

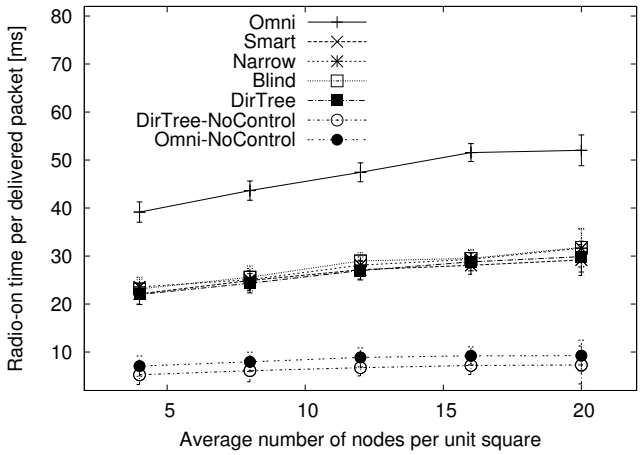


Figure 9: Radio-on time per delivered packet against network density. DPF *halves radio-on time. Without control traffic, radio-on time is further reduced.*

any control traffic; and *ii)* Omni-NoControl works the same as Omni but also spares the control traffic by relying on an optimal routing topology computed similarly to DirTree, based solely on omni-directional transmissions. These schemes provide a means to measure the effectiveness of the packet forwarding functionality alone, without any control overhead. Therefore, they represent a theoretical upper bound for any solution using directional and omni-directional transmissions, respectively.

We measure the average radio-on time spent per delivered packet—representing the unit cost of delivering a packet to the sink—to provide a concise indication on the effectiveness of a given schema. In most cases, both DirTree-NoControl and Omni-NoControl deliver close to 100% of packets at the sink: without control traffic, the few remaining collisions rarely cause packet losses.

Results. Figure 9 shows the trends w.r.t. varying network density, although the following observations also apply w.r.t. varying packet generation rates and number of nodes.

The chart shows that DPF saves about half of the control overhead in Omni: the curves for DPF solutions lie halfway between Omni and Omni-NoControl. This complements the results of Section 6.1, further evidencing that simple modifications to existing protocols coupled with a practical antenna design already enable significant improvements.

We also observe a significant gap between DPF solutions and DirTree-NoControl, roughly as large as the one between the former and Omni. The overhead here is entirely due to the way routes are built and maintained—absent in DirTree-NoControl—based on ETX and broadcast beaconing in our case. As we discussed in Section 5, this is the mainstream technique with omni-directional transmissions, but the extensive use of broadcast poorly matches the operation of directional antennas, for which more efficient protocols can be designed. The aforementioned overhead gap quantifies the margin

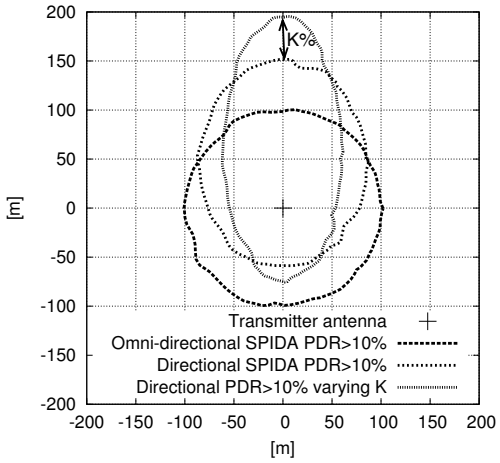


Figure 10: Region of space with $PDR > 10\%$, depending on K.

of improvement available to these protocols.

Finally, we note that DirTree-NoControl performs about 30% better than Omni-NoControl, which corresponds to the reduction in overall route stretch enabled by the larger communication range of SPIDA when in directional mode (Figure 4). Indeed, in our simulations the average tree depth for DirTree-NoControl is about one third smaller than Omni-NoControl. We further discuss this aspect next, as this is where the operation of network protocols meets the antenna characteristics, and one may find opportunities for different antenna designs.

6.3 Opportunities for Antenna Designers

Intuitively, the effect of reducing the route stretch—key to many of the improvements discussed thus far—should be amplified by an antenna design that further increases the communication range in the direction of maximum gain. The open question, however, is to what extent this impacts the overall performance.

To answer this question, we artificially adapt our empirical link-layer model, as illustrated in Figure 10. We change the shape of the main lobe to increase the communication range at the expense of spatial coverage perpendicular to the direction of maximum gain. In particular, we maintain constant the area covered by the main lobe—in a sense proportional to the energy invested in transmission—but redistribute the PDR information to increase the maximum communication range by $K\%$.

Formally, we compute first:

$$H_{K=0\%} = \int \int_{x \in A} PDR(x)$$

where x is a point in the two-dimensional area of space where our empirical link-

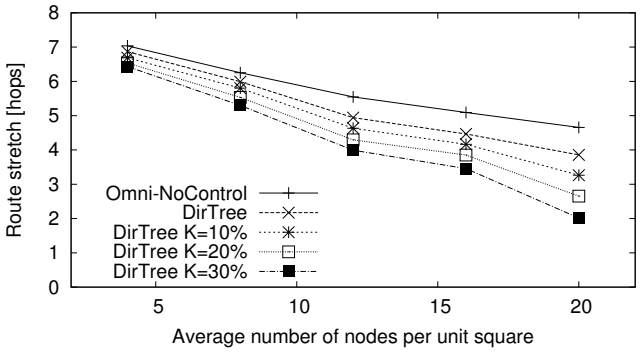


Figure 11: The route stretch decreases when K and network density increase.

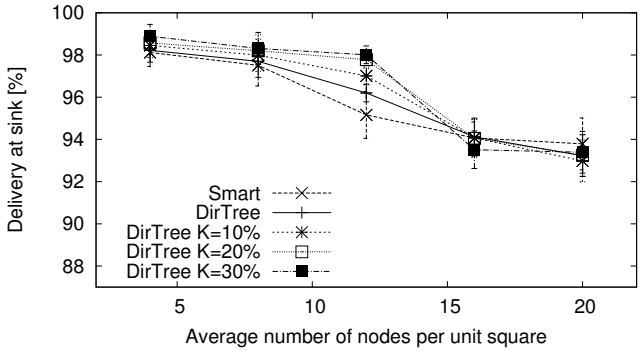
layer model determines that $PDR > 0\%$. Next, we consider the value of $PDR(x)$ as the pixel value of a bi-dimensional image. Using Matlab's image processing toolbox, we process such image by applying a dilation operator subject to the constraint $\forall K, H_K = H_{K=0\%}$. We iteratively dilate the image until the value of K in Figure 10 is either 10%, 20%, or 30%. Such values are in line with the technological limits of practical directional antenna designs in low-power wireless [24]. The pixel values of the processed image give us back the PDR value of the adapted link-layer model.

Figure 11 shows how the route stretch varies w.r.t. network density, for the optimal off-line topology used in DirTree and DirTree-NoControl. For comparison, the omni-directional case shown is also based on the optimal routing topology used in Omni-NoControl. The chart shows that, as expected, increased network density alone causes a decrease in route stretch, as seen in all curves. As more nodes are in reach, shorter routes become possible. However, directional antennas amplify this effect. At the maximum density we consider, DirTree with the original SPIDA yields routes 30% shorter; when $K = 30\%$, routes are 60% shorter.

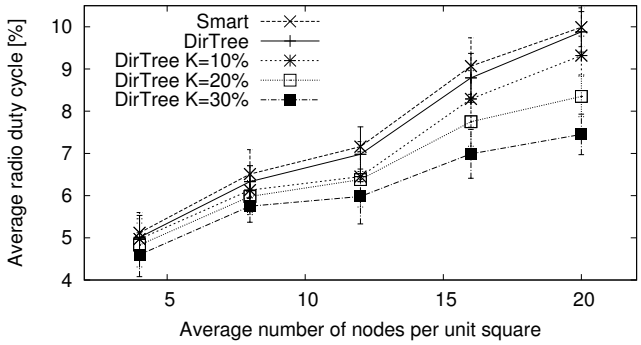
To evaluate the impact of reduced route stretch on the overall performance, we use our modified link-layer model to re-run the simulations in Section 6.1. We consider only Smart among the DPF solutions of Section 3, as it represents the best trade-off between packet delivery and radio duty-cycle.

Results. Figure 12(a) shows that antennas with larger communication ranges improve packet delivery only at a specific network density. The latter is the "sweet spot" where the benefits to route stretch of directional transmissions are not countered by an excessive network density. When the network is too sparse, the increased communication range does not make a difference; there are few nodes to choose from, and routes are almost identical regardless of K. When the network is too dense, collisions dominate performance, annihilating the benefits of a reduced route stretch.

On the other hand, the radio duty-cycle improves in all settings, as shown in Figure 12(b). In sparse networks, the savings are enabled by the reduced need for retransmissions, given that the route topology is almost identical in all



(a) Average packet delivery. *Packet delivery improves only at a specific network density with increasing K.*



(b) Average radio duty-cycle. *The radio duty-cycle decreases for increasing K in all settings, saving energy.*

Figure 12: Performance of antenna designs against varying network density.

cases. Indeed, increasing K yields a stronger signal and therefore a better *PDR* at the parent. We expect this behavior to manifest more prominently in real networks, due to the capture effect [17]. In denser networks, routes are shorter and therefore fewer overall transmissions are necessary.

Similar trends and considerations hold for the results obtained by varying packet generation rates, as in Section 6.1. When varying the number of nodes, instead, the gains "multiply" across more nodes, as larger communication ranges allow to cover the network in fewer hops: DirTree with $K = 30\%$ almost halves the radio duty-cycle compared to Smart.

These quantitative results offer antenna designers a clear indication of the margin of improvement available, motivating real-world designs similar to the one we artificially create.

7 Conclusions

We explored the benefits of ESD antennas in low-power wireless. We showed that minimal changes to an existing protocol enable significant improvements with a low-cost ESD antenna, *e.g.*, almost halving the radio-on time per delivered packet. Furthermore, our results suggest that clean-state protocol designs that focus on the control rather than the data plane may bring similar additional gains, and indicate further room for improvements with different antenna designs.

References

[1] M. Buettner, G. V. Yee, E. Anderson, and R. Han. X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks. In *SENSYS*, 2006.

[2] M. Ceriotti et al. Monitoring heritage buildings with wireless sensor networks: The Torre Aquila deployment. In *IPSN*, 2009.

[3] M. Ceriotti et al. Is there light at the ends of the tunnel? Wireless sensor networks for adaptive lighting in road tunnels. In *IPSN*, 2011.

[4] A. Cerpa, J. L. Wong, L. Kuang, M. Potkonjak, and D. Estrin. Statistical model of lossy links in wireless sensor networks. In *IPSN*, 2005.

[5] R. Choudhury, T. Ueda, J. Bordim, and N. Vaidya. Beamnet: An ad hoc network testbed using beamforming antennas. In *Vehicular Technology Conference*, 2005.

[6] R. Choudhury, X. Yang, R. Ramanathan, and N. Vaidya. On designing MAC protocols for wireless networks using directional antennas. *IEEE Transactions on Mobile Computing*, 5(5), 2006.

[7] D. D. Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. *Wirel. Netw.*, 11(4), 2005.

[8] H. Dai, K. Ng, M. Li, and M. Wu. An overview of using directional antennas in wireless networks. *Communication Systems*, Nov. 2011.

[9] J. Eriksson et al. COOJA/MSPSim: Interoperability testing for wireless sensor networks. In *SIMUTools*, 2009.

[10] E. Felemban et al. Samac: A cross-layer communication protocol for sensor networks with sectored antennas. *IEEE Transactions on Mobile Computing*, 9(8), 2010.

[11] E. Felemban et al. Sand: Sectored-antenna neighbor discovery protocol for wireless networks. In *SECON*, 2010.

[12] E. Gaura, L. Girod, J. Brusey, M. Allen, and G. Challen. *Wireless Sensor Networks: Deployments and Design Frameworks*. Springer, 2010.

[13] G. Giorgetti, A. Cidronali, S. Gupta, and G. Manes. Exploiting low-cost directional antennas in 2.4 GHz IEEE 802.15.4 wireless sensor networks. In *European Conf. on Wireless Technologies*, 2007.

[14] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis. Collection tree protocol. In *SENSYS*, 2009.

[15] J. Ko, J. Eriksson, N. Tsiftes, S. Dawson-Haggerty, M. Durvy, J. Vasseur, A. Terzis, A. Dunkels, and D. Culler. Beyond interoperability: Pushing the performance of sensornet IP stacks. In *SENSYS*, 2011.

[16] M. Kodialam and T. Nandagopal. Characterizing achievable rates in multi-hop wireless networks: The joint routing and scheduling problem. In *MOBICOM*, 2003.

[17] J. Lu and K. Whitehouse. Flash flooding: Exploiting the capture effect for rapid flooding in wireless sensor networks. In *INFOCOM*, 2009.

[18] L. Mottola and G. P. Picco. Muster: Adaptive energy-aware multisink routing in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 10(12), 2011.

[19] M. Nilsson. Directional antennas for wireless sensor networks. In *Scandinavian Workshop on Wireless Adhoc Networks*, 2009.

[20] E. Öström, L. Mottola, and T. Voigt. Evaluation of an electronically switched directional antenna for real-world low-power wireless networks. In *Real-World Wireless Sensor Networks (REALWSN)*, 2010.

[21] J. Polastre, R. Szewczyk, and D. Culler. Telos: Enabling ultra-low power wireless research. In *IPSN*, 2005.

[22] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis. An empirical study of low power wireless. *ACM Transactions on Sensor Networks*, 2010. To appear.

[23] K. Srinivasan, M. Kazandjieva, S. Agarwal, and P. Levis. The β-factor: measuring wireless link burstiness. In *SENSYS*, 2008.

[24] D. Thiel and S. Smith. *Switched parasitic antennas for cellular communications*. Artec House, London, 2002.
[25] S. Vasudevan, J. Kurose, and D. Towsley. On neighbor discovery in wireless networks with directional antennas. In *INFOCOM*, 2005.
[26] F. Viani, L. Lizzi, M. Donelli, D. Pregnolato, G. Oliveri, and A. Massa. Exploitation of parasitic smart antennas in wireless sensor networks. *Electromagnetic Waves and Applications*, 24(7), 2010.
[27] B. Y. Wu and K.-M. Chao. *Spanning Trees and Optimization Problems*. Chapman & Hall, 2004.