

A Metric for Opportunistic Routing in Duty Cycled Wireless Sensor Networks

Euhanna Ghadimi ^{*}, Olaf Landsiedel [†], Pablo Soldati [‡] and Mikael Johansson ^{*}
euhanna@kth.se, olaf@chalmers.se, pablo.soldati@huawei.com, mikaelj@kth.se

^{*}School of Electrical Engineering, KTH - Royal Institute of Technology, Sweden

[†] Chalmers University of Technology, Sweden

[‡] Huawei Technologies Sweden AB

Abstract—Opportunistic routing is widely known to have substantially better performance than traditional unicast routing in wireless networks with lossy links. However, wireless sensor networks are heavily duty-cycled, i.e. they frequently enter deep sleep states to ensure long network life-time. This renders existing opportunistic routing schemes impractical, as they assume that nodes are always awake and can overhear other transmissions.

In this paper, we introduce a novel opportunistic routing metric that takes duty cycling into account. By analytical performance modeling and simulations, we show that our routing scheme results in significantly reduced delay and improved energy efficiency compared to traditional unicast routing. The method is based on a new metric, EDC, that reflects the expected number of duty cycled wakeups that are required to successfully deliver a packet from source to destination. We devise distributed algorithms that find the EDC-optimal forwarding, i.e. the optimal subset of neighbors that each node should permit to forward its packets. We compare the performance of the new routing with ETX-optimal single path routing in both simulations and testbed-based experiments.

I. INTRODUCTION

Opportunistic routing has shown the potential to significantly increase throughput in multi-hop wireless mesh-networks compared to the traditional unicast routing [1]–[4]. It exploits the fact that in wireless mesh-networks, radios are always-on and hence can overhear messages at practically no additional cost. Hence, a single transmission is often received by multiple receivers, each providing a specific routing progress to the destination. The receiver that provides the maximum routing progress will then forward the packet [1]. In summation, opportunistic routing leverages spatial reuse to realize two key benefits: it provides high routing progress and limits the impact of link dynamics. This leads to a substantial throughput improvement relative to traditional routing schemes [1], [2].

While routing protocols for wireless sensor networks (WSNs) also aim for high routing progress and resilience to link dynamics, existing proposals for opportunistic routing cannot be directly applied. In order to guarantee long network lifetime, WSNs are commonly duty-cycled. Nodes in deep sleep states have their radios turned off to save energy and will not be able to overhear transmissions from other nodes. The main distinction of this work from existing opportunistic routing schemes is that it considers the practical issues of

duty cycling and presents an opportunistic routing protocol tailored to WSNs. A packet is forwarded by the first awoken neighbor that successfully receives it and offers routing progress towards the destination. As a result, by minimizing the average number of duty-cycles required for end-to-end packet delivery, we significantly improve energy efficiency and reduce delay compared to unicast routing in WSNs. We achieve this by introducing a new anycast routing metric, EDC (Estimated Duty Cycled wake-ups), that focuses on energy efficiency and delay instead of high throughput. EDC estimates the on-time, i.e., energy consumption, of the radios of the nodes that a packet traverses from sink to destination. Similar to routing with the widespread metric ETX [5], that minimizes the expected number of transmissions required to forward a packet from sink to source, routing with EDC minimizes the energy consumption of packet forwarding in an opportunistic WSN.

This paper builds upon our previous work [6] which introduces ORW, a practical opportunistic routing scheme for wireless sensor networks. While [6] presents a practical realization of opportunistic routing and evaluates its benefits in both simulation and TinyOS based testbed experiments, current paper focuses on theoretical aspects of opportunistic routing in WSNs and makes four core contributions:

- 1) We introduce a detailed analytical model to measure the expected number of duty cycled wake-ups as an indicator of the average end-to-end delay in WSNs.
- 2) We present the EDC metric as a heuristic to estimate this value. Compared to the detailed analytical expressions, EDC is computationally simple while maintaining key properties of loop freeness and local computability.
- 3) In our evaluation we show that the analytical model and the heuristic lead to the nearly the same routing topologies.
- 4) In simulations and deployments we compare our anycast EDC metric to the widespread ETX metric for unicast routing, and show that EDC outperforms ETX significantly in terms of radio duty cycles and delay.

The remainder of this paper is structured as follows: After reviewing the opportunistic routing in wireless networks in Section II, we introduce the basic concepts of our anycast routing metric in Section III. We propose a theoretical framework to calculate the expected number of wake-ups in Section IV.

This work was partially funded by the Swedish Research Council (VR) and the Swedish Foundation for Strategic Research (SSF).

Section V presents EDC metric. Forwarder set construction and algorithmic properties of EDC metric is discussed in Section VI. Section VII evaluates and compares our metric against ETX. We discuss related work in Section VIII. Section IX concludes.

II. BACKGROUND

In this section, we provide the required background on opportunistic routing in mesh networks and discuss why it cannot be directly utilized in wireless sensor networks.

A. Opportunistic Routing

Opportunistic routing [1]–[4] improves network throughput in the context of multi-hop, mesh networks. In contrast to traditional unicast routing, opportunistic routing delays the forwarding decision until after the transmission. In ExOR [1] each packet is sent to a set of potential forwarding nodes, prioritized by routing progress. Based on its priority, each node in the forwarder set is assigned a time slot for forwarding, which it only utilizes if it did not overhear the packet being forwarded in a previous time slot. Relying on such a consensus protocol, opportunistic routing avoids duplicate forwarding. To sum up, opportunistic routing leverages spatial reuse to ensure high routing progress and to limit the impact of link dynamics. This leads to a significant throughput improvement when compared to traditional routing schemes [1], [2].

Routing protocols in WSNs follow similar goals: high routing progress and resilience to link dynamics. However, opportunistic routing cannot be directly applied: Wireless sensor networks and their applications pose special requirements, such as low-power networking and resource constraints, that distinct them from traditional multi-hop mesh networks. These limit the direct applicability of opportunistic routing in following aspects:

- 1) **Reliability and Energy Efficiency vs. Throughput:** Opportunistic routing is designed to improve network throughput. However, WSN applications commonly demand reliable forwarding at high energy-efficiency and not high throughput. In this paper, we show how opportunistic routing can be adapted to improve the energy efficiency when compared to traditional WSN routing.
- 2) **Duty Cycling in Sensor Networks:** Commonly, sensor networks are duty-cycled to ensure long node and network lifetime. Hence, nodes are in deep sleep states most of the time, with their radios turned off. Duty-cycling limits the number of nodes that concurrently overhear a packet (assuming no prior synchronization). As a result, it prevents the spatial reuse in the forwarding process, one of the key benefits of opportunistic routing.

B. Adapting Opportunistic Routing to WSNs

After introducing the concept of opportunistic routing and discussing its limitations in the context of WSNs, we discuss its adaptation to the special requirements of WSNs. Our work targets duty-cycled protocol stacks. For simplicity we here illustrate the basic concept of our opportunistic routing scheme for WSNs utilizing an asynchronous low-power-listening MAC, such as in X-MAC [7]. In low-power-listening a sender transmits a stream of packets until the intended receiver wakes up

and acknowledges it (see Fig. 1a and Fig. 1b). To integrate opportunistic routing into duty cycled environments, we depart from this traditional unicast forwarding scheme in one key aspect: The first node that (a) wakes up, (b) receives the packet, and (c) provides routing progress, acknowledges and forwards the packet, see Fig. 1c.

Our design enables an efficient adaptation of opportunistic routing to the specific demands of wireless sensor networks: (1) In contrast to opportunistic routing in mesh networks, forwarder selection in our scheme focuses on energy efficiency and delay instead of network throughput: It minimizes the radio-on time until a packet is received by a potential forwarder. (2) It integrates well into duty-cycled environments and ensures that many potential forwarders can overhear a packet in a single wake-up period. Thereby, we exploit spatial and temporal link diversity to improve resilience to wireless link dynamics.

In our routing scheme, a packet is forwarded by the first awoken neighbor that provides routing progress. As a result, a node has multiple parents and the routing topology towards a destination is not a tree anymore as in traditional unicast-based routing protocols. It assembles a Directed Acyclic Graph (DAG) with a single destination (Destination Oriented DAG, DODAG). In this DODAG each packet potentially traverses on a different route to the destination (anycast).

III. BASIC CONCEPT: ANYCAST ROUTING METRIC EDC

To enable energy-efficient opportunistic routing in duty-cycled WSNs, we introduce EDC (Estimated Duty Cycled wake-ups) as routing metric. EDC is an adaptation of ETX [5] to energy-efficient, anycast routing. ETX estimates the number of transmissions required to deliver a packet from source to sink. In contrast, our metric EDC estimates the radio on-time, i.e., the radio duty cycle.

We rely radio duty-cycles as key metric as it – in contrast to transmission counts – reflects the energy consumption of the communication in duty cycled WSNs. Hence, minimizing the EDC of the node directly reduces its energy consumption and increases node and network lifetime. Furthermore, we show in our evaluation that EDC also reduces delay when compared to ETX while achieving similar hop counts.

A. Trade-Offs in Parent Selection

Using EDC as routing metric, each node i maintains a set of admissible neighbors called *forwarder set* in which every member, upon receiving a packet from i , is eligible to relay the packets. The optimal forwarder set F_i of a node i is the subset of its neighbors that leads to the minimum EDC of i . Two factors impact the choice of the forwarder set F_i : (1) adding more neighboring nodes to the forwarder set reduces the time until one of the potential forwarders wakes up to receive. Hence, it decreases the single-hop EDC of the forwarding node and improves spatial diversity (see Fig. 1c). However, (2) adding too many neighboring nodes to the forwarder set may decrease its average routing progress, as commonly not all neighbors provide good progress.

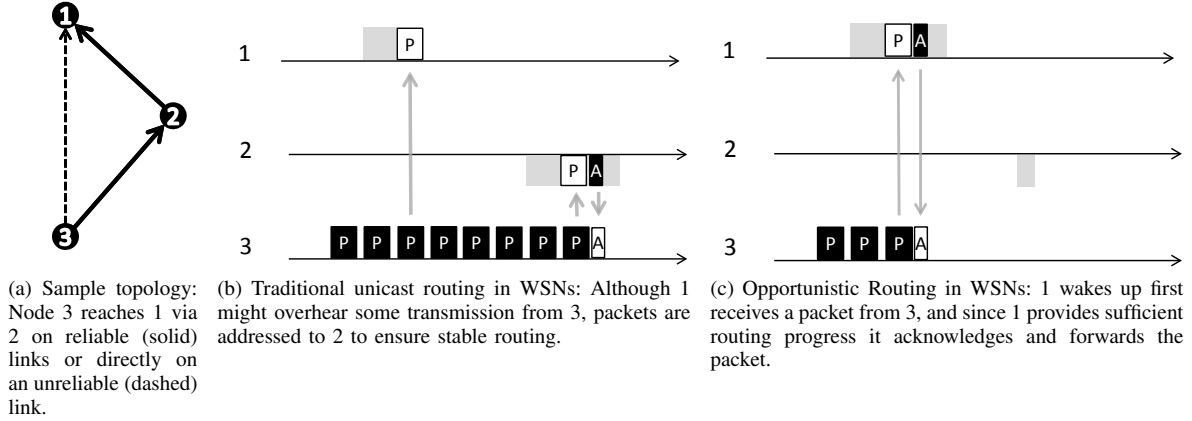


Fig. 1: Main idea of opportunistic routing scheme for WSNs is to exploit spatial and temporal link diversity to reduce energy consumption and delay.

B. Requirements for EDC as Routing Metric

As routing metric, we require EDC the following set of properties:

- **Loop Free Topology:** To ensure stable routing, the resulting topology is required be loop-free.
- **Distributed Computability:** Computation of EDC of a node shall only require information from neighboring nodes.
- **Low Algorithmic Complexity:** To match the scarce resources of a sensor node in terms of computing capabilities, memory, and energy resources, the our algorithms must be lightweight.

Next we provide an analytical model to calculate the average number of wake-ups for end-to-end packet delivery. Later, we will show that EDC, as an approximation for the theoretical model shows a strongly reduced computational complexity when compared the analytical model.

IV. EXPECTED NUMBER OF DUTY CYCLED WAKE-UPS

We represent the topology of the network as a directed graph $\mathcal{G} = \{\mathcal{N}, \mathcal{L}, \mathcal{P}\}$, with a set of nodes $\mathcal{N} = \{1, \dots, N\}$, a set of links \mathcal{L} , where a link is represented by an ordered pair (i, j) of distinct nodes, and a set of probabilities \mathcal{P} . The packet loss process on each link $(i, j) \in \mathcal{L}$ follows a Bernoulli process with success probability p_{ij} , and we assume that packet loss processes on links are independent of each other. Let F_i be the set of forwarders of node i . For the moment, we assume that the forwarder set of each node is fixed. Later, in Section VI, we address how individual nodes can perform this forwarder selection. Further, we will assume that nodes use the same duty cycle length T . At duty cycle k and before going to dormant state, node i draws the next wake-up time t_k uniformly in the interval $[0, T]$; *i.e.* node i wakes up at $t_0, T + t_1, 2T + t_2$, etc. In this setting, we are interested in analyzing the expected number of duty cycled wake-ups for a transmission from source to destination. To this end, let DC be the random number of duty cycled wake-ups required to complete the end-to-end transmission, and note that we can

divide it into two independent components. First, the number of wake-ups required for a single-hop transmission from the source node to one of its forwarders, and second, the number of wake-ups the packet takes to reach from the forwarder node to the sink. Since link losses are assumed to be independent, we can write

$$\mathbf{E}\{DC(i)\} = \mathbf{E}\{DC_s(i)\} + \mathbf{E}\{DC_m(i)\}, \quad (1)$$

where $\mathbf{E}\{DC_s(i)\}$ is the expected number of duty cycled wake-ups until the packet has been received by one of the forwarders, and $\mathbf{E}\{DC_m(i)\}$ is the expected number of remaining duty cycled wake-ups it takes to complete the multi-hop transmission. The number of wake-ups required for a single-hop transmission can be seen as the sum of two independent random variables, $DC_s(i) = X_s(i) + Y_s(i)$ where $X_s(i)$ is the number of failed intervals (in which all forwarders wake up once and fail to receive the transmission from node i) and $Y_s(i)$ is the waiting time within a duty cycle with successful transmission. A failed interval requires that all transmissions between i and $j \in F_i$ fail, hence $X_s(i)$ follows a geometric distribution with pdf

$$\Pr\{X(i) = k\} = \prod_{j \in F_i} (1 - p_{ij})^k \left(1 - \prod_{j \in F_i} (1 - p_{ij}) \right),$$

and mean

$$\mathbf{E}\{X_s(i)\} = \sum_{k=0}^{\infty} k \Pr\{X(i) = k\} = \frac{\prod_{j \in F_i} (1 - p_{ij})}{1 - \prod_{j \in F_i} (1 - p_{ij})}. \quad (2)$$

To characterize $Y_s(i)$, note that for node $j \in F_i$ to be the forwarder, it must experience a successful transmission and transmissions to all other nodes that wake up before j must fail. Note that the underlying event is conditioned on that at least one node will successfully receive in the current duty cycle. Hence, the probability of reception for $j \in F_i$ in an arbitrary duty cycle conditioned on at least one reception is given by

$$p_s(j) = \frac{p_{ij}}{1 - \prod_{k \in F_i} (1 - p_{ik})}. \quad (3)$$

In general, this event can happen for $2^{|F_i|-1} - 1$ cases which is based on exploring the probabilities of having all nodes included in the possible subsets of forwarders $\{\forall f \subset F_i \setminus j\}$ have failed before node j . Let $\mathcal{F}_{k \setminus j}$ denote the set of all the subsets of forwarders of node i with cardinality equal k not containing node j . The probability of having exactly k failed transmission excluding node j is given by

$$p_f(k \setminus j) = \sum_{l \in \mathcal{F}_{k \setminus j}} \prod_{m \in l} (1 - p_{im}). \quad (4)$$

Due to continuity of the random variable, the probability of having two nodes with the same activation time is zero. Thus, the mean waiting time is achieved by iterating among all the nodes $j \in F_i$ with i.i.d uniform wake-ups and different link qualities. Namely,

$$\mathbf{E}\{Y_s(i)\} = \frac{1}{T} \int_0^\infty x \sum_{j \in F_i} \sum_{k=0}^{|F_i|-1} \frac{p_s(j)}{T} p_f(k \setminus j) \left(\frac{x}{T}\right)^k \left(\frac{T-x}{T}\right)^{|F_i|-k-1} dx. \quad (5)$$

Note that since $\mathbf{E}\{Y_s(i)\}$ does not have dimension -it is a portion of duty cycle- we normalize the above equation by dividing it by T . As an example consider node i with k forwarders with the same success probabilities $p_{ij} = 1$. It turns out that in this case, $\mathbf{E}\{DC_s(i)\}$ has a closed form. One can see that $\mathbf{E}\{X_s(i)\} = 0$ and $\mathbf{E}\{DC_s(i)\} = \mathbf{E}\{Y_s(i)\} = \frac{1}{T} \int_0^T k \frac{x}{T} \left(\frac{T-x}{T}\right)^{k-1} dx = \frac{1}{k+1}$. We observe that the mean single hop waiting time decreases hyperbolically with increasing number of neighbors.

$\mathbf{E}\{DC_m(i)\}$ is the expected number of duty cycled wake-ups which it takes to send the packet from the forwarder set F_i to the sink given that a successful transmission has already took place between i and F_i . $\mathbf{E}\{DC_m(i)\}$ is given by

$$\mathbf{E}\{DC_m(i)\} = \sum_{j \in F_i} \Pr\{j \text{ is the forwarder}\} \mathbf{E}\{DC(j)\}, \quad (6)$$

where

$$\Pr\{j \text{ is the forwarder}\} = \sum_{k=0}^{|F_i|-1} \frac{1}{\binom{|F_i|}{k}} \sum_{l \in \mathcal{F}_{k \setminus j}} \prod_{m \in l} (1 - p_{im}) \frac{p_s(j)}{|F_i|}.$$

In words, this corresponds to the probability of node j being the first successful receiver given that at least one forwarder receives in current duty cycle. Up to now, we have developed an analytical framework to measure the cost of packet delivery for each sender in terms of the average number of duty cycles. In contrast of a few rudimentary cases -like when all link reliabilities are equal 1- the analysis and even simulations tend to be intractable with respect to increased network density. Thus, the the expected number of duty cycled wake-ups $\mathbf{E}\{DC(i)\}$ cannot be used for selecting the optimal forwarder set in practical opportunistic routing protocols.

We next introduce a new metric, referred to as EDC, that properly mimics the behaviour the exact $\mathbf{E}\{DC(i)\}$. This metric shall be the basis for the design of a new low-complexity and distributed algorithm for opportunistic routing, where the selection and the ordering of the forwarders set of each node is in accordance with the exact $\mathbf{E}\{DC(i)\}$ metric.

V. THE EDC METRIC

We have seen that an accurate evaluation of the expected number of duty cycled wake-ups under opportunistic forwarding is quite complex, even when the forwarder sets are fixed. While the protocol does not need an analytical formula for the expected number of wake-ups, it does need a procedure for selecting the optimal forwarder sets. To this end, we need to define a lightweight metric that captures the essential features of opportunistic forwarding, yet allows us to develop provably correct algorithms for distributed forwarder set selection.

We have found that a metric which we call EDC strikes an appealing balance between effectiveness in forwarder set selection and simplicity of protocol analysis. The EDC of a node i can be computed recursively via

$$\text{EDC}(i) = \frac{1}{\sum_{j \in F_i} p_{ij}} + \frac{\sum_{j \in F_i} p_{ij} \cdot \text{EDC}(j)}{\sum_{j \in F_i} p_{ij}} \quad (7)$$

The first term approximates the expected one-hop forwarding delay $\mathbf{E}\{DC_s(i)\}$ while the second term attempts to capture the essential features of the subsequent delay from forwarders to the sink. The single-hop cost shares many important features of the analytical model: it is a hyperbolic function of the link reliabilities of the forwarders, and adding a forwarder or increasing the link reliabilities decrease the single-hop cost. Figure 2 illustrates a situation with homogeneous links ($p_{ij} = p$ for all j) and compares $\mathbf{E}\{DC_s(i)\}$ with the proposed hyperbolic approximation in EDC metric. We see that the analytical model and approximation agree with each other when of the number of forwarders increases.

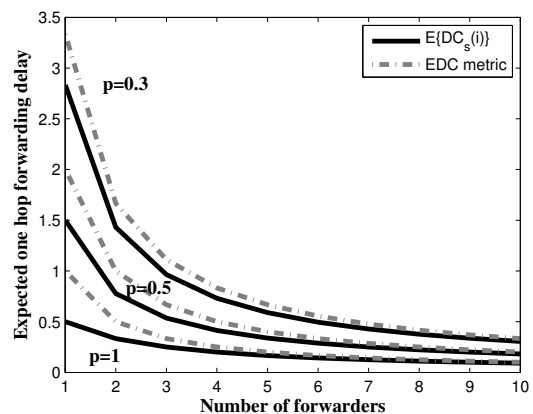


Fig. 2: Comparison of $\mathbf{E}\{DC_s(i)\}$ versus corresponding term of EDC metric for $F_i = [1, 10]$ forwarders. The success probabilities p are equal for each forwarder.

In the analytical model, the complexity of the multi-hop cost comes from the fact that the probability of j being a forwarder depends on the link reliabilities of the other nodes (the probability of a node j being a forwarder not only depends on that it could successfully decode the transmission when it is awake, but it also depends on the probability that nodes that woke up earlier all failed to decode the packet from i). The EDC metric simply assumes that the probability

that j is a forwarder is directly proportional to p_{ij} . With this assumption, the probability of being forwarder for each node j is independent of others. However, since probability of forwarders must add up to one, we normalize each individual forwarding probability. Hence, in EDC metric, the probability that node j with reliability p_{ij} being forwarder is $\frac{p_{ij}}{\sum_{j \in F_i} p_{ij}}$. Evaluation results in section VII confirm the accuracy of EDC metric compared with the analytical scheme under real network scenarios.

VI. FORWARDER SET CONSTRUCTION

In this section, we describe how to maintain the EDC metric in the network. Our key contribution is a distributed algorithm for forwarder selection that minimizes EDC. In particular, we show that after the appropriate ordering of potential forwarders, nodes can use a greedy algorithm to find the optimal forwarder set, and that this algorithm is loop free.

In the forwarder selection problem, each node i is given a set of potential forwarders N_i , their EDC metrics $EDC(j)$ for $j \in N_i$, and the probability of p_{ij} successful transmission from i to $j \in N_i$, and should determine the subset of forwarders $F_i^* \subseteq N_i$ that minimizes $EDC(i)$. To develop our algorithm, we first review some rudimentary properties of the EDC metric.

Lemma 6.1: Let $p_{ij} \geq 0$ and $c_j > 0$ for every $j \in N_i$. Define the set function $f_i : 2^{N_i} \rightarrow \mathbb{R}$ with $f_i(\emptyset) = \infty$ and

$$f_i(A) = f_i^{(1)}(A) + f_i^{(2)}(A) = \frac{1}{\sum_{j \in A} p_{ij}} + \frac{\sum_{j \in A} c_j \cdot p_{ij}}{\sum_{j \in A} p_{ij}},$$

for $A \subseteq N_i$. Then $f_i^{(1)}(A)$ is strictly decreasing in p_{ij} and $f_i^{(2)}(A)$ is strictly increasing in c_j .

Proof: The first result can be verified by inspecting $f_i^{(1)}(A \cup \{x\}) - f_i^{(1)}(A)$. For second result one can check that $f_i^{(2)}(A \cup \{k\}) - f_i^{(2)}(A) > 0$ given that $c_k > c_j$ for all $j \in A$. ■ The following lemma presents a sufficient condition for when the insertion of a new member $k \in N_i \setminus A$ in the forwarder set decreases $EDC(i)$,

Lemma 6.2: Let $p_{ik} > 0$. If $c_k < f_i(A)$ then $f_i(A \cup \{k\}) < f_i(A)$ and vice versa.

Proof: The sign of $c_k - f_i(A)$ is the same as the sign of $f_i(A \cup \{k\}) - f_i(A)$, since

$$f_i(A \cup \{k\}) - f_i(A) = \frac{p_{ik} \cdot (c_k - f_i(A))}{\sum_{j \in A} p_{ij} + p_{ik}}$$

and $p_{ik} > 0$. Our result follows. ■

The next result observes the behavior of $f(A)$ after adding a new member into the forwarding set F_i .

Lemma 6.3: If $c_k < f_i(A)$ and $p_{ik} > 0$ then $f(A \cup \{k\}) > c_k$.

Proof: Consider

$$f_i(A \cup \{k\}) - c_k = \frac{f_i(A) - c_k}{1 + \frac{p_{ik}}{\sum_{j \in A} p_{ij}}}$$

Our assumptions imply that the right-hand side is positive, hence $f_i(A \cup \{k\}) > c_k$. ■

Up to now, we have concluded that it is beneficial for node i to add a new neighbor k to the forwarder set if $EDC(k)$ is

less than the EDC of node i when k is not in the forwarder set. Moreover, after adding node k , the updated EDC of node i is greater than $EDC(k)$. The next Theorem characterizes the optimum forwarder set of node i .

Theorem 6.4: Let π be an ordering of the nodes in N_i such that $\{c_{\pi(1)} \leq c_{\pi(2)} \leq \dots \leq c_{\pi(|N_i|)}\}$. Then, the optimal forwarder set is $F_i^* = \{\pi(1), \dots, \pi(k)\}$ where k satisfies $c_k < f_i(F_i^*)$ and $c_{k+1} > f_i(F_i^*)$.

Proof: Assume that $F_i^* = \{\pi(1), \dots, \pi(k)\} \setminus \pi(m)$ for some $m < k$, i.e. a node m with $c_m \leq c_k$ has been excluded from the forwarder set. According to Lemma 6.3 we have $c_j < f_i(F_i^*)$, $\forall j \in F_i^*$. Lemma 6.2 ensures that adding m with $c_m \leq c_k$ will decrease $f_i(F_i^*)$, i.e., $f_i(F_i^* \cup \{c_m\}) < f_i(F_i^*)$ so F_i^* is not optimal and a contradiction achieved. ■

One question here is what happens if two or more neighbors have equal EDC's but different link reliabilities. The next lemma shows that even though the node with higher reliability results in a larger decrease of the EDC, the optimal forwarder set either includes both nodes or none.

Lemma 6.5: if $c_k = c_{k'} < f_i(A)$ and $p_{ik} > p_{ik'} > 0$ then $f_i(A \cup \{k\}) < f_i(A \cup \{k'\}) < f_i(A)$. Moreover, if $c_k \in F_i^*$, then $c_{k'} \in F_i^*$ and vice versa.

Proof: The first part of the proof follows the Lemma 6.2 and inspecting the inequality

$$f_i(A \cup \{k'\}) - f_i(A \cup \{k\}) = \frac{(p_{ik} - p_{ik'})(f_i(A) - c_k)}{(p_{ik} + \sum_{j \in A} p_{ij})(\frac{p_{ik'}}{\sum_{j \in A} p_{ij}} + 1)} > 0.$$

For the second part, assume $c_k \in F_i^*$ and $c_{k'} \notin F_i^*$. Since $c_k \in F_i^*$ then by Lemma 6.3, $f_i(F_i^*) > f_i(c_k)$. Now, due to $c_{k'} < f_i(F_i^*)$ and $c_{k'} \notin F_i^*$ we conclude that $f_i(F_i^* \cup \{k'\}) < f_i(F_i^*)$ and consequently contradiction is achieved. So $c_{k'} \in F_i^*$. A similar argument applies to the inverse case, i.e., if $c_{k'} \in F_i^*$, then $c_k \in F_i^*$. ■

Given a network topology $\mathcal{G} = \{\mathcal{N}, \mathcal{L}, \mathcal{P}\}$, Lemmas 6.2-6.3 and Theorem 6.4 suggest a greedy algorithm to compute the EDC metric in the network and to locally construct the set F_i^* at each node. Starting from the sink with $EDC = 0$, each node i in the network sorts its neighbors in the set N_i in increasing order of their EDCs. A potential forwarder $j \in N_i$ is added to the set of forwarders F_i^* of node i if $EDC(j) < EDC(i)$, upon which $EDC(i)$ is updated on the basis of the new set $F_i^* \cup \{j\}$. The procedure repeats until the forwarding list and the EDC values of all the nodes remain unchanged.

Each iteration k of Algorithm 1 produces a new routing topology $\mathcal{R}^{(k)} = \{\mathcal{N}, \mathcal{L}^{(k)}, \mathcal{E}^{(k)}\}$ where $\mathcal{L}^{(k)}$ consists of links $l = (i, j)$ from a node i to all its forwarders $j \in F_i^*$, and $\mathcal{E}^{(k)}$ network-wide set of updated EDC. We next prove that each routing topology $\mathcal{R}^{(k)}$ is loop-free.

Lemma 6.6: Any routing topology $\mathcal{R}^{(k)} = \{\mathcal{N}, \mathcal{L}^{(k)}, \mathcal{E}^{(k)}\}$ produced by Algorithm 1 is a directed acyclic graph (DAG).

Proof: According to Lemma 6.2, at each iteration k Algorithm 1 a new member $j \in N_i$ is added to the forwarder set F_i^* for node $i \in \mathcal{N}$ if $EDC(j) < EDC(i)$. By Lemma 6.3, this event can only happen since at iteration $k-1$ node $i \notin F_j^*$, i.e. otherwise $EDC(j) > EDC(i)$. Furthermore, upon adding j to F_i^* we have $EDC(i) = f(F_i^* \cup \{j\}) > EDC(j)$ by Lemma 6.3, which

Algorithm 1 EDC-based opportunistic routing.

Input: $\mathcal{G} = \{\mathcal{N}, \mathcal{L}, \mathcal{P}\}$, neighbor set $N_i \forall i \in \mathcal{N}$.
Initialize: $EDC(\text{Sink}) \leftarrow 0$, $EDC(i) \leftarrow \infty$.
repeat
 Set $F_i^* = \emptyset \forall i \in \mathcal{N}$
 for all $i \in \mathcal{N}$ **do**
 Sort $N_i = \{n_1, \dots, n_{|N_i|}\}$ s.t. $EDC(n_j) < EDC(n_{j+1})$.
 for $j = 1 \rightarrow |N_i|$ **do**
 if $EDC(j) < EDC(i)$ **then**
 a. Update: $F_i^* = F_i^* \cup \{n_j\}$.
 b. Update: $EDC(i)$ based on eq. (1).
 end if
 end for
 end for
until EDC of all nodes are unchanged.

Testbed	Size nodes	Sink id	Tx Power dBm	Diameter hops
Motelab	123	1	0	5.2
Twist	86	229	0	3.4

TABLE I: Testbed Overview: In our evaluation we use PRR traces from two testbeds.

ensures that node i cannot be added to the optimal forwarder set F_j^* of j in future iterations of the Algorithm. Thus any two nodes $i, j \in \mathcal{N}$ can only be connected by either an arc (i, j) or (j, i) in each rooting topology $\mathcal{G}^{(k)}$.

Let now consider a path of arbitrary length $P_i = \{i, j_1, j_2, \dots, j_n\}$ from node i to the sink in $\mathcal{G}^{(k)}$, with $j_1 \in F_i^*$, $j_2 \in F_{j_1}^*$, and so on. Let assume that P_i has a loop involving node i , i.e. there exist a $j_k \in P_i$ such that $i \in F_{j_k}^*$, hence by Lemma 6.2 $EDC(i) < EDC(j_k)$. However, by applying Lemma 6.2 to each hop of the path P_i we have that $EDC(i) > EDC(j_n)$, $\forall j_n \in P_i$ and a contradiction is achieved. ■

VII. EVALUATIONS

We conduct simulations and real hardware implementation to evaluate how EDC metric performs in terms of average delay, average hop count and average forwarder count. Our evaluation consist of four parts: (1) We validate the analytical model in extensive simulations. (2) We show that our approximation reaches a high level of accuracy. (3) We compare our anycast metric EDC to the widespread routing metric ETX. (4) We provide deployment results and compare an EDC based opportunistic routing protocol to the widespread Collection Tree Protocol (CTP) [8], which uses ETX as routing metric.

In the first three parts we solely focus on the EDC metric and not protocol implementations using EDC. Hence, we deliberately exclude artifacts from protocol implementations such the slow spreading of route updates or packet collisions. To ensure a fair and realistic evaluation our network profile is based on PRR traces from two testbeds: Twist [9] and Motelab [10] with 86 and 123 nodes, respectively (see Table I).

A. Validating the analytical model

We first compare the analytical $\mathbf{E}\{DC\}$ versus simulation and EDC metric in a small example. We consider a node with 1 to 10 forwarders with fixed average wake-ups and link reliabilities. Figure 3 shows the average wake-ups versus forwarder set. Each data point in the curves corresponds to the updated y-axis values after adding a new forwarder. The first observation is the accurate match between theoretical model and the simulation. Leveraging on this match, we will continue our evaluations through analytical model. Second, with growing number of forwarders, EDC values eventually tend to the theoretical values. We intentionally have sorted forwarders in increasing order of average wake-ups. The plot represents the behavior of forwarder selection algorithm. Starting from the forwarder with minimum EDC, sender inserts a new member into forwarding set, if the EDC of the new forwarder is less than current value of y-axis, in which in this example happens until the third forwarder.

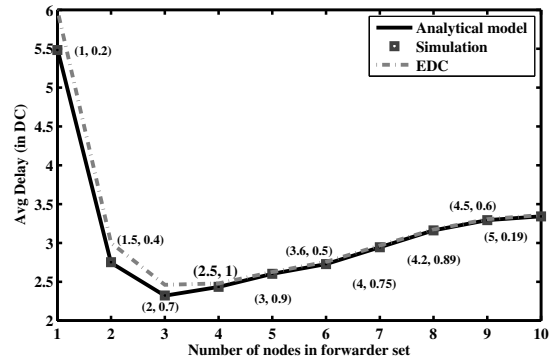


Fig. 3: Comparison of analytical model versus simulation and EDC metric for a node with $[1, 10]$ forwarders. Forwarder set grows incrementally with new members which are shown by tuple (EDC, reliability).

B. EDC versus analytic model

To evaluate EDC metric, we run forwarder set construction mechanism (Algorithm 1) over Twist and Motelab data traces and compare it versus exhaustive search based on analytic model (1). The number of forwarders for each node is limited up to 10 nodes. This restriction is made due to the exponential complexity of the analytical model as well as the exhaustive search over the forwarder candidates: with a state of the art PC, current setup requires 2 and 4 hours of simulations for Twist and Motelab profiles, respectively. Figures 4a and 4b show optimal duty cycles ($\mathbf{E}\{DC\}$) versus EDC metric for each node of the network. We note that EDC values are very close to the analytical values. Figures 4c and 4d illustrate the number of forwarders versus node index. For each node, the number of optimal forwarders (given by exhaustive search) and the numbers from the EDC metric are depicted. Moreover, the number of common forwarders in these two schemes as well as the number of truly ordered common forwarders are plotted. We observe that the last three curves coincide to each other.

In other words, EDC metric for each node, picks a subset of forwarders with the same order as the optimal set. Roughly speaking, for each node, the best forwarder in optimal set (the one with lowest $E\{DC\}$) is also the first forwarder that EDC metric picks and so on.

The EDC values of nodes based on different restrictions on the number of forwarders is illustrated in Figure 5. For some nodes (the ones closer to the sink) the EDC value does not change. The reason is that due to good paths towards the sink, they do not add more forwarders. In contrast, the nodes farther to the sink (the ones with higher EDC values) will benefit with having more forwarders. We observe that the values for these profiles does not change significantly after 20 forwarders.

C. Comparison with ETX

After evaluating the accuracy of EDC and the impact of routing table size, we next compare EDC to the widespread routing metric Expected Transmission Count (ETX) [5]. ETX is a unicast routing metric that estimates the number of (re)transmission a packet is expected to require to reach its intended destination. Routing protocols using ETX aim to minimize the transmission count. In contrast to EDC, ETX does not take duty cycling into account. Hence, reducing transmission counts in ETX does not necessarily lead to low delays nor does it reduce radio on-time.

Figures 6a and 6d show that EDC significantly reduces delivery delay when compared to ETX in both testbeds. In our deployment results, we also show that this directly reduces radio duty cycles and hence radio energy consumption. Furthermore, we show that depending on the neighbor table size, EDC achieves hop counts similar to ETX (see Figures 6b and 6e). In some situations it even outperforms ETX.

Due to its opportunistic nature, EDC based routing explicitly utilizes all neighbors: Instead of waiting for one specific neighbor to wake up as ETX based routing, EDC utilizes the first neighbor that wakes up and provides routing progress. As a result, EDC outperforms ETX in terms of delay, while leading to more hops. This result shows that relying merely on hop count as an indicator of delay and energy consumption is not efficient since a sender may waste both time and energy by waiting for a best neighbor to wake up. However, limiting the number of parents in the routing table, we also limit this aggressive forwarding and force EDC based routing to select a small number of good parents (see Figures 6c and 6f). In this way we trade delay for hop counts and maintenance time of the forwarder set: By letting more candidates in the forwarder set we experience less delay meanwhile routing algorithm requires more steps to stabilize. Note that even though EDC requires larger forwarder set than ETX, the overhead of routing table management remains unchanged. In practice, ETX also needs to keep the same number of neighbors as EDC to keep track the best routing option.

D. Initial Deployment Results

We conclude our evaluation, by presenting initial deployment results for our ongoing implementation of EDC based oppor-

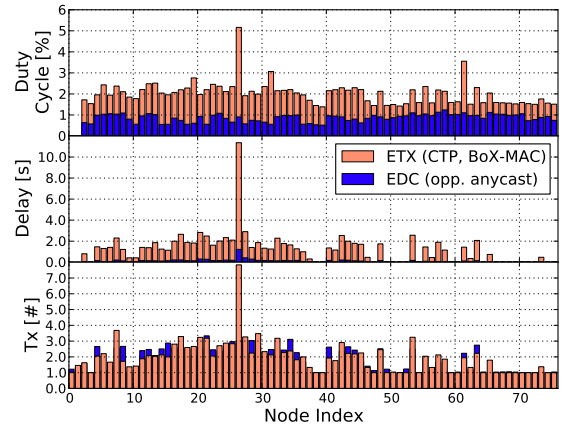


Fig. 7: Initial Deployment Results on the Twist Testbed: EDC based routing outperforms ETX based CTP in terms of radio-duty cycles by a factor of about 2 and delay by a factor of 10 while achieving similar but slightly higher hop counts.

tunistic anycast routing in TinyOS. We compare our approach to the Collection Tree Protocol (CTP) [8], the de-facto standard collection protocol in TinyOS. We use the default TinyOS BoX-MAC [11]. Based on X-MAC, it is an asynchronous duty cycled MAC layer and has shown good results for CTP. Figure 7 shows that EDC based opportunistic routing significantly improves duty cycles and delay on the Twist testbed. On average, it doubles the energy efficiency, individual nodes show improvements up to a factor of five. Additionally, it improves delay by an average factor of 10 and achieves transmission counts that are similar, but slightly higher when compared to CTP.

VIII. RELATED WORK

Similar to our work, [12]–[16] consider anycast routing in WSNs. Their results show that opportunistic routing can improve energy efficiency and delay compared to traditional unicast routing. These results strongly motivated our work.

LCAR [12] assigns a relay candidate-set to each node in order to minimize the expected cost of forwarding a packet to the destination. The expected cost is recursively constructed by assuming that the relay nodes already know their own forwarding cost to the destination. Mao *et al.* [13] study the selection and prioritizing the forwarding list to minimize the overall expected energy consumption of WSNs. In contrast to our work, they do not consider the duty cycling in the theoretical model.

Joint study of anycast forwarding and duty cycling has been done in some recent works. Ashref *et al.* [14] argues that forwarding to an earliest awoken neighbor can decrease per hop delay. Basu [15] estimates the end-to-end latency of a duty cycled wireless network under random walk routing strategy. Kim *et al.* [16] investigates the optimal anycast forwarding policy for a poisson wake-up model to minimize the expected end-to-end delay in the event-driven WSNs. However, in all of them forwarder selection merely depends on the wake-up process and the probability of link failure is not considered:

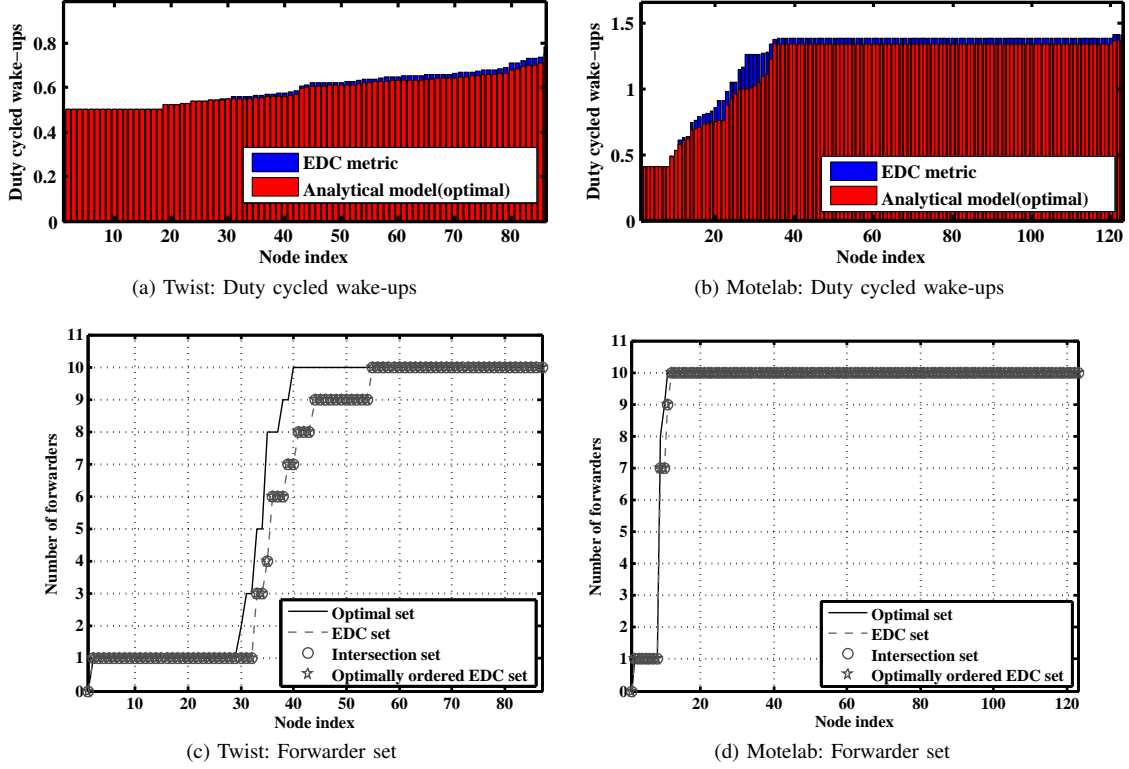


Fig. 4: Per node comparison of EDC values and forwarder set in motelab and twist profile. The maximum number of forwarders is restricted to 10. The plots show high accuracy of EDC metric in both duty cycles and forwarder selection compared with optimal solution derived by exhaustive search.

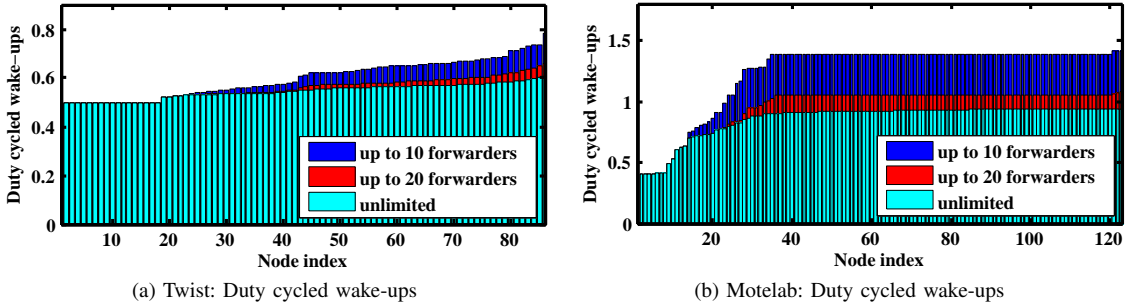


Fig. 5: Per node comparison of EDC values in motelab and twist profile with different number of forwarders.

We consider this a key requirement, as due to the low-power nature of the WSNs, their links are highly dynamic [17].

Opportunistic routing in WSNs also has received great attention from a more practical perspective. GeRaF [18] and CMAC [19] utilize geographic routing for opportunistic forwarding. Relying solely on geographic routing, they do not address the key challenges for opportunistic routing in duty-cycled WSNs such as wireless link dynamics, anycast routing metrics, and energy efficiency. DSF [20] selects the next hop of a packet based on the sleep schedule of neighboring nodes and different metrics such delay, reliability, and energy consumption. Similar to our work, DSF shows strong improvements over unicast

routing in these metrics. However, it focuses on synchronized networks.

IX. CONCLUSIONS

This paper introduced a novel routing metric, Estimated Duty Cycled wake-ups (EDC), for opportunistic routing in duty-cycled WSNs. Reducing radio duty-cycles directly impacts the key resource in battery-powered sensor networks: the severely limited energy supplies. We established key properties of EDC as routing metric and showed that it can be computed distributively and leads to a loop free topology. Comparisons with a detailed analytical model established that EDC is an accurate

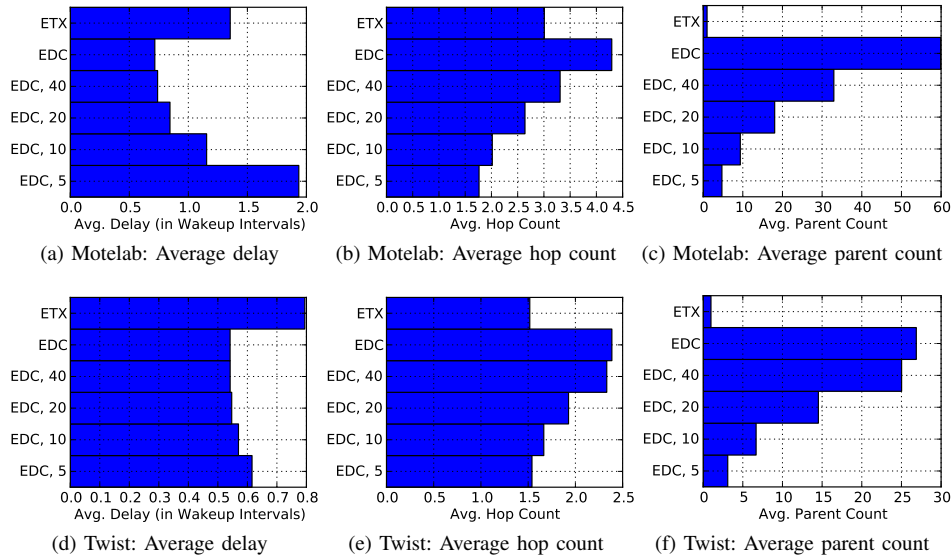


Fig. 6: Comparing EDC and ETX on Motelab and Twist traces: Independent of routing table size, EDC outperforms ETX in terms of delay. Additionally, EDC outperforms ETX in terms of hops for small routing tables.

approximation of the true number of duty-cycled wakeups required to forward the packet. Finally, we showed in both simulations and initial deployments that EDC yields significantly improved radio-duty cycle counts and delays compared to ETX.

REFERENCES

- [1] S. Biswas and R. Morris. ExOR: Opportunistic Multi-Hop Routing for Wireless Networks. In *SigComm: Proc. of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, 2005.
- [2] S. Chachulski, M. Jennings, S. Katti, and D. Katabi. Trading Structure for Randomness in Wireless Opportunistic Routing. In *SigComm: Proc. of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, 2007.
- [3] P. Larsson. Selection Diversity Forwarding in a Multihop Packet Radio Network with Fading Channel and Capture. *SIGMOBILE Mob. Comput. Commun. Rev.*, 5, 2001.
- [4] R. Choudhury and N.H. Vaidya. MAC-Layer Anycasting in Ad Hoc Networks. *SIGCOMM Comput. Commun. Rev.*, 34(1), 2004.
- [5] S. J. Douglas, D. Aguayo, J. Bicket, and R. Morris. A High-Throughput Path Metric for Multi-Hop Wireless Routing. In *MobiCom: Proc. of the ACM Int. Conference on Mobile Computing and Networking*, 2003.
- [6] O. Landsiedel, E. Ghadimi, S. Duquennoy, and M. Johansson. Low power, low delay: Opportunistic routing meets duty cycling. In *IPSN: Proc. of ACM/IEEE Int. Conference on Information Processing in Sensor Networks*, 2012.
- [7] Michael Buettner, Gary V. Yee, Eric Anderson, and Richard Han. X-MAC: a Short Preamble MAC Protocol for Duty-Cycled Wireless Sensor Networks. In *SenSys: Proc. of the ACM Int. Conference on Embedded Networked Sensor Systems*, 2006.
- [8] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis. Collection Tree Protocol. In *SenSys: Proc. of the ACM Int. Conference on Embedded Networked Sensor Systems*, 2009.
- [9] V. Handziski, A. Köpke, A. Willig, and A. Wolisz. TWIST: a Scalable and Reconfigurable Testbed for Wireless Indoor Experiments with Sensor Networks. In *Proc. of the Int. Workshop on Multi-hop Ad Hoc Networks: from Theory to Reality*, 2006.
- [10] G.W. Allen, P. Swieskowski, and M. Welsh. MoteLab: a wireless sensor network testbed. In *IPSN: Proc. of ACM/IEEE Int. Conference on Information Processing in Sensor Networks*, 2005.
- [11] D. Moss and P. Levis. BoX-MACs: Exploiting Physical and Link Layer Boundaries in Low-Power Networking. Technical Report SING-08-00, Stanford, 2008.
- [12] A. Dubois-Ferrière, M. Grossglauser, and M. Vetterli. Valuable Detours: Least-Cost Anypath Routing. *IEEE/ACM Trans. Netw.*, 19(2), 2011.
- [13] X. Mao, S. Tang, X. Xu, X. Li, and H. Ma. Energy-efficient opportunistic routing in wireless sensor networks. *IEEE Trans. Parallel and Distributed Systems*, 22(11), 2011.
- [14] F. Ashref, R.H. Kravets, and N.H. Vaidya. Exploiting Routing Redundancy using MAC Layer Anycast to Improve Delay in WSN. *SIGMOBILE Mob. Comput. Commun. Rev.*, 14, 2010.
- [15] P. Basu and C. Chau. Opportunistic Forwarding in Wireless Networks with Duty Cycling. In *CHANTS: Proc. of the ACM Workshop on Challenged Networks*, 2008.
- [16] J. Kim, X. Lin, N. Shroff, and P. Sinha. Minimizing Delay and Maximizing Lifetime for Wireless Sensor Networks with Anycast. *IEEE/ACM Trans. Netw.*, 18, 2010.
- [17] K. Srinivasan, M.A. Kazandjieva, S. Agarwal, and P. Levis. The β Factor: Measuring Wireless Link Burstiness. In *SenSys: Proc. of the ACM Int. Conference on Embedded Networked Sensor Systems*, 2008.
- [18] M. Zorzi and R. Rao. Geographic Random Forwarding (GeRaF) for Ad Hoc and Sensor Networks: Multihop Performance. *IEEE Trans. on Mobile Computing*, 2, 2003.
- [19] S. Liu, K. Fan, and P. Sinha. CMAC: An Energy-Efficient MAC Layer Protocol using Convergent Packet forwarding for Wireless Sensor Networks. *ACM Trans. Sen. Netw.*, 5, 2009.
- [20] Y. Gu and T. He. Data Forwarding in Extremely Low Duty-Cycle Sensor Networks with Unreliable Communication Links. In *SenSys: Proc. of the ACM Int. Conference on Embedded Networked Sensor Systems*, 2007.