Toward Stable Network Performance in Wireless Sensor Networks: A Multilevel Perspective

SHAN LIN, Stony Brook University
GANG ZHOU, College of William and Mary
MO'TAZ AL-HAMI, Temple University
KAMIN WHITEHOUSE, YAFENG WU, and JOHN A. STANKOVIC, University of Virginia
TIAN HE, University of Minnesota
XIAOBING WU, Nanjing University
HENGCHANG LIU, University of Science and Technology of China

Many applications in wireless sensor networks require communication performance that is both consistent and of high quality. Unfortunately, performance of current network protocols can vary significantly because of various interferences and environmental changes. Current protocols estimate link quality based on the reception of probe packets over a short time period. This method is neither efficient nor accurate enough to capture the dramatic variations of link quality. Therefore, we propose a link metric called *competence* that characterizes links over a longer period of time. We combine competence with current short-term estimations in routing algorithm designs. To further improve network performance, we have designed a distributed route maintenance framework based on feedback control solutions. This framework allows every link along an end-to-end (E2E) path to adjust its link protocol parameters, such as transmission power and number of retransmissions, to ensure specified E2E reliability and latency under dynamic link qualities. Our solutions are evaluated in both extensive simulations and real system experiments. In real system evaluations with 48 T-Motes, our overall solution improves E2E packet delivery ratio over existing solutions by up to 40% while reducing transmission energy consumption by up to 22%. Importantly, our solution also achieves more stable and better transient performance than current approaches.

Categories and Subject Descriptors: C.2.2 [**Computer-Communication Networks**]: Network Protocols

General Terms: Design, Algorithms, Performance

Additional Key Words and Phrases: Wireless sensor networks, performance stability, transmission power control, stable routing, wireless interference, real-time routing

This work is supported in part by the National Science Foundation (NSF) and SAIC under NSF grants CNS-1239108, CNS-1218718, IIS-1231680, CNS-1444021, CNS-0614870, and CNS-0626632, and NSF CAREER grant CNS-1253506. Dr. Wu is supported by NSFC grants 61373130 and 61321491, NSF of Jiangsu Province grant (BK20141319, and the EU FP7 CROWN project under grant PIRSES-GA-2013-610524. Dr. Liu is supported by NSFC grant 61472384. We thank the anonymous reviewers for their insightful comments.
Authors' addresses: S. Lin, Department of Electrical and Computer Engineering, Stony Brook University; email: shan.x.lin@stonybrook.edu; G. Zhou, Computer Science Department, College of William and Mary; email: gzhou@cs.wm.edu; M. Al-Hami, Computer and Information Sciences Department, Temple University; email: motaz@temple.edu; K. Whitehouse, Y. Wu, and J. A. Stankovic, Computer Science Department, University of Virginia; email: {whitehouse, yw5s, stankovic}@cs.virginia.edu; T. He, Computer Science Department, University of Minnesota; email: tianhe@cs.umn.edu; X. Wu, Department of Computer Science and Technology, State Key Lab for Novel Software Technolgoy, Nanjing University; email: wuxb@nju.edu.cn; H. Liu, School of Computer Science and Technology, University of Science and Technology of China; email: hcliu@ustc.edu.cn.

DOI: http://dx.doi.org/10.1145/2700272

ACM Reference Format:
Shan Lin, Gang Zhou, Mo'Taz Al-Hami, Kamin Whitehouse, Yafeng Wu, John A. Stankovic, Tian He, Xiaobing Wu, and Hengchang Liu. 2015. Toward stable network performance in wireless sensor networks: A multilevel perspective. ACM Trans. Sensor Netw. 11, 3, Article 42 (February 2015), 26 pages.
DOI: http://dx.doi.org/10.1145/2700272

1. INTRODUCTION

Extensive studies [Abdelzaher et al. 2008; He et al. 2006; Lu et al. 2002; Buttazzo 2006] suggest that predictable end-to-end (E2E) reliability and delay are critical for many wireless sensor network applications, such as surveillance and emergency response, to meet performance guarantees. However, these wireless sensor networks are exposed to various interferences from their environments, which causes the network performance to vary dramatically and unpredictably. Therefore, it is both important and challenging to provide good network performance consistently.

Low-power wireless link qualities in sensor networks can vary at a wide range of timescales. For example, when packet loss occurs because of Wi-Fi interference, link quality can degrade in a matter of seconds, whereas environmental changes can affect link quality at much longer timescales. In wireless sensor networks, many network protocols estimate link quality based on recent probe packets over a short time period, such as the widely used expected number of transmission (ETX) metric [Bicket et al. 2005; Fonseca et al. 2007; Woo et al. 2003]. These solutions can achieve high reliability as long as the estimations accurately reflect the link quality when a packet is actually transmitted. Unfortunately, short-term estimations may not accurately reflect the performance in indoor environments [Hackmann et al. 2008; Srinivasan et al. 2008]. In our experiments on an indoor testbed, we have observed two types of links, which we refer to as stable and unstable links. Although the link quality of a stable link remains around a certain level, the quality of an unstable link often changes dramatically within a few seconds or minutes. Current short-term link estimations are not effective in differentiating between these two types of links, as both may maintain good qualities over a short time period. Moreover, current short-term estimations are not efficient for unstable links, as the high frequency of link measurement that they require leads to increased energy consumption and interference. Further, such links may not estimate their qualities accurately, and when selected for routing, they may not be discarded the moment their qualities drop dramatically. As a result, E2E communication quality drops and energy consumption due to retransmission increases. In addition, the network may experience cascading route changes: newly selected routes introduce interference to other nearby routes, triggering even more packet loss, energy consumption, and route changes. The cascading route changes can result in significant E2E quality variations and energy consumption.

To address this problem, it is essential to differentiate between stable and unstable links, and give preference to stable links. We notice that these two types of links have different qualities over long-term periods, in the tens of minutes. Therefore, we propose a new link metric, *competence*, to characterize the long-term link quality. The competence metric can help choose those good and stable links for routing and drop those currently good but unstable ones. However, a system using only long-term estimations would react too slowly to link quality changes. To react quickly and provide stable performance, we combine competence with current short-term estimations in novel routing algorithm designs, selecting links that are good in both the short and the long term.

To assist in achieving stable network performance, especially stable reliability and latency, we also design a route maintenance framework based on competence. Our framework integrates feedback control solutions at both the link and network layers.

In the link layer maintenance, nodes use per-link transmission power control and retransmission control. Under certain dynamics, they help unstable links achieve stable performances at a specified level and help stable links become more robust. The per-link performance level requirement is injected by the network layer maintenance. The network layer maintenance uses a feedback loop along an active path to translate a given E2E performance specification into per-link requirements to minimize total transmission energy consumption along the path. This loop also distributes these requirements to link layer control modules at each node.

We evaluated the designs on an indoor testbed with 48 T-Motes, which showed that our solution improves packet delivery ratios (PDRs) over existing solutions by up to 40% and reduces power consumption by up to 22%. In addition, real system experiments demonstrated more stable performance with less variance and better transient performance than existing solutions. For the evaluation of E2E latency, we employed extensive trace-driven analysis. Analysis results show that our solution meets specified E2E latency requirements by regulating transmission power levels and the number of retransmissions of each sensor node along routing paths.

The contributions of our work are as follows:

—We have established that long-term link estimation is important to achieve stable and efficient networking in the presence of interference and environmental changes. We propose the competence metric to characterize the long-term quality of links for wireless sensor networks.
—We demonstrate that with a control-based design, reliability of existing solutions can be further improved, and more stable and better transient performances can be achieved.
—We evaluate our solutions in a 48-node wireless sensor network testbed under real scenarios to demonstrate that stable network performances can be achieved under various interference and environmental changes.

We present related work in Section 2. We introduce the competence metric design with an empirical study in Section 3. In Section 4, we present competence-enhanced routing. In Section 5, we describe our feedback control framework design. In Section 6, we evaluate this system on an indoor wireless testbed and in simulations. We draw the conclusions in Section 7.

2. RELATED WORK

There are a number of wireless networking protocols that use various techniques to deal with the dynamics of wireless communication quality. At the MAC layer, short-term link quality estimation [Kim and Noble 2001; Woo et al. 2003; Fonseca et al. 2007; Zhao and Govindan 2003; Miluzzo et al. 2008; Wachs et al. 2007] is critical, but we established that the long-term link quality estimation is also important, as link qualities can vary dramatically. We propose a metric called *competence* to quantify long-term link quality and help design routing algorithms and a framework to achieve stable E2E performance. Kim and Noble [2001] use statistical results to choose the short-term best link estimation filter dynamically for mobile systems. In our work, the competence metric focuses on the long-term link estimation, which is used alongside the short-term link estimation. We also use it in control designs to achieve stable network performance. Srinivasan et al. [2008] propose a β factor to quantify the short-term correlations among successes and failures of transmissions. Competence is different, because it emphasizes the long-term communication quality.

At the network layer, the existing routing protocols [Gu and He 2007; Woo et al. 2003; Intanagonwiwat et al. 2003; Polastre et al. 2005; Dunkels et al. 2007; Cerpa

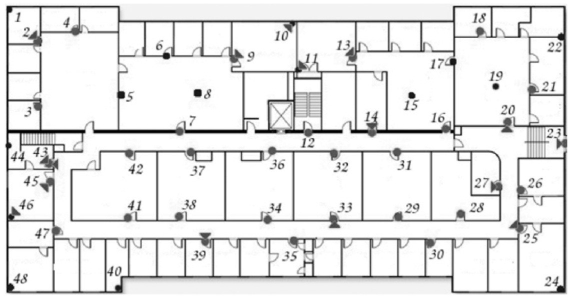


Fig. 1. The testbed network layout.

et al. 2005; Kim et al. 2003] have developed mechanisms to select good links when link quality changes. In this work, we have demonstrated that selection of long-term good links is also critical for high reliability and reducing cascading route changes. In our routing algorithm design, we use the competence metric in addition to previous link metrics to achieve better performance.

Many control-based designs have been proposed for E2E quality of service (QoS) in computing systems. These studies present elegant designs in different systems with their specific constraints, such as data servers [Abdelzaher et al. 2008], distributed real-time embedded systems [Lu et al. 2005; Shankaran et al. 2008; Lu et al. 2002], wireless sensor networks [He et al. 2006; Chipara et al. 2006], topology control [Santi 2005; Puccinelli et al. 2011], and Internet protocol design [Schulzrinne and Casner 1993; Comer 2000]. Our control design is unique in its coordination of pairwise control at the link layer and E2E control at the network layer, based on reliability and energy constraints in wireless sensor networks.

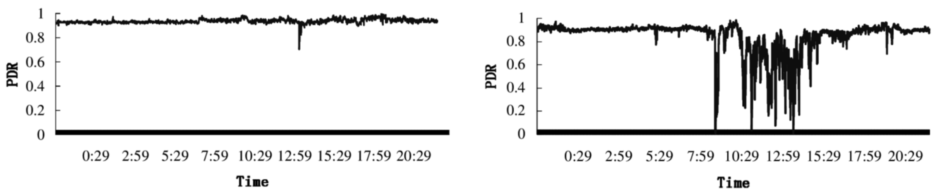
For QoS in wireless sensor networks, many studies [Chakrabarti and Mishra 2001; Chen and Heinzelman 2005; Zhou et al. 2011; Asare et al. 2012] discuss challenging issues, where there is accurate knowledge about the network state. Other studies focus on selecting the path for real-time routing. Liu et al. [2012] proposes a framework for quantifying the probabilistic path delay in real time for distributed resource-constrained devices. The work uses multitimescale estimation to capture the highly varying nature of link delay. Some other works focus on average path delay [Xue et al. 2011; Felemban et al. 2005]. Xue et al. [2011] propose the SDRCS communication scheme for real-time traffic sensing in wireless sensor networks, using geographical forwarding without considering the link probabilistic nature delay. Unlike these works, this article focuses on providing stable E2E delay that meets the real-time requirements of wireless sensor network applications. Our solution employs a feedback control–based design to maintain delay stability over dynamic link qualities.

3. THE COMPETENCE METRIC FOR QUANTIFYING LONG-TERM QUALITY

3.1. Empirical Study

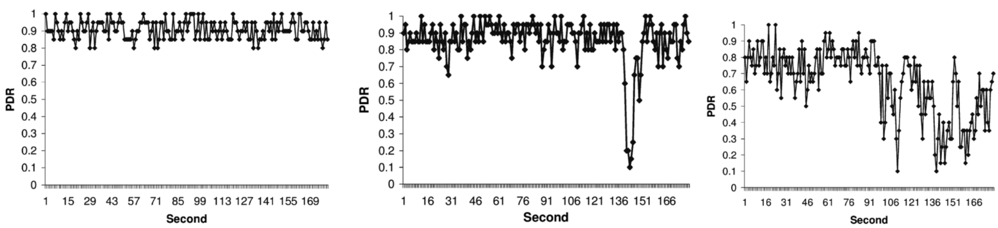
We built a wireless testbed in our computer science building, as shown in Figure 1. It consists of 48 T-Motes with Chipcon CC2420 low-power radios. For sensing purposes, we placed these nodes at various heights along the wall. Some of them are close to the doors, and some of them are on the top of the office cubicles.

In the first experiment, we programmed three source nodes to broadcast at a rate of 20 packets per second, whereas all of the other nodes just listened and recorded the packets that they received. We scheduled transmissions to avoid collisions. The link-level retransmission and acknowledgement functionalities were disabled.



(a) A Stable Link (b) An Unstable Link

Fig. 2. Stable link versus unstable link.



(a) Night: Radio Signal Fading (b) Day: People Walking Around (c) Day: People Using Wi-Fi

Fig. 3. Link quality variation under different interference.

We made three observations based on these experiments:

—There are two kinds of links in the deployed system—stable links and unstable links— given the specific transmission rate. An example of such links is shown in Figure 2.
—There are three main causes for the link quality variations of unstable links. We distinguished these causes in the testbed through successfully reproducing patterns of PDRs in different scenarios. These patterns are shown as small fluctuation (Figure 3(a)), large disturbance (Figure 3(b)), and continuous large fluctuation (Figure 3(c)). The small fluctuations in Figures 3(a) are mainly caused by multipath fading of wireless signals. The large disturbances in Figure 3(b) are caused by shadowing effects of humans, doors, and other objects. The continuous large fluctuations in Figure 3(c) are caused by Wi-Fi interference. The duration of these disturbances can vary from fractions of a second to tens of minutes. The variance in link qualities is largely due to a combination of instances of these three patterns.
—We also identify that there are temporal and spatial impacts of human-related activities on link quality. The quality of links in an office decreased in the morning when people walked in and started using Wi-Fi. The quality of links in the lounge demonstrated a noticeable variation at noon when people had lunch. Moreover, similar trends were observed from links situated near each other, because human-related activities have an impact on these links at the same time. However, the degree of impact is different, depending on many factors such as the distance to the interference source. Similar results were observed in indoor [Hackmann et al. 2008; Srinivasan et al. 2008] and outdoor environments [He et al. 2006; Selavo et al. 2007].

3.2. Competence Metric Design

These experiments motivated us to study the stability and transient performance for wireless sensor networks in the presence of significant and rapid (sometimes within a matter of seconds) changes in communication quality. Stability and transient performance are two of the main foci of control theory [Abdelzaher et al. 2008; Hellerstein et al. 2004], so we review the metrics with control theory before presenting our design.

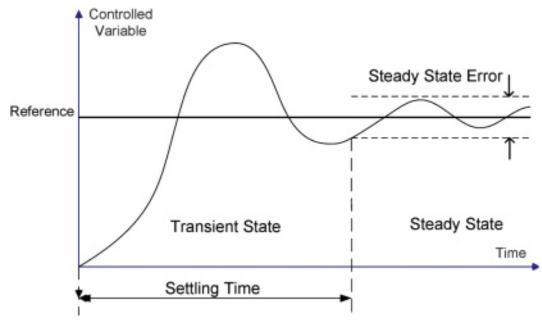


Fig. 4. Transient performance metric in control theory.

Figure 4 presents the basic metrics for studying transient performance in control theory. When a certain change occurs in the system, the controlled variable deviates from the reference value. The reference value defines the level at which the controlled variable is expected to stabilize. The system is in a *steady state* when the controlled variable lies within the range between reference ± steady state error. Otherwise, the system is in a *transient state*. Another important metric is *settling time*, which defines the amount of time the system takes to stabilize to a steady state when disturbances occur. The values of reference, steady state error, and settling time are specified as the control goals of the system.

These concepts and metrics are foundations of stability and transient performance analysis. However, directly applying these metrics is not reasonable, as the distributed wireless network system is open and involves many uncertainties. For example, classic control systems adjust the control variable to converge to a single reference value, within the bounds defined by a very small steady state error (2% of reference value is a reasonable design for a closed, well-modeled control system [Hellerstein et al. 2004]). For wireless communication quality, however, a single reference value with a very small steady range is not feasible, as fading of wireless signals can cause the PDR to vary more than 20% (Figure 3(a)). Therefore, we need a different way to quantify stability in wireless sensor networks.

We formally define a performance metric: *competence*. Competence is a long-term performance metric that is based on a short-term performance measure s. s is a binary function indicating whether the current signal is within a desired range. The metric $c(t)$ for competence value at time t is defined in Equation (1):

$$c(t) = \alpha \cdot c(t-1) + (1-\alpha) \cdot s, 1 > \alpha > 0 \tag{1}$$

$$s = \begin{cases} 1 & y(t) \in [T_{lower}, T_{upper}] \\ 0 & otherwise. \end{cases}$$

We define T_{upper} and T_{lower} as the upper and lower bounds that specify a desired range for a network performance measure, such as communication quality. An exponential weighted moving average (EWMA) filter is used on the binary function s that indicates whether the current communication quality is within the specified range or not. α is a smoothing factor indicating the weight of history when calculating the current value. $y(t)$ is the currently observed communication quality, like PDR. The value of $c(t)$ is between 0 and 1. If the communication quality always falls into a specified range, the value of $c(t)$ is always 1.

There are several research works on link quality estimations, using filter design [Kim and Noble 2001] and other indications [Fonseca et al. 2007]. These works provide

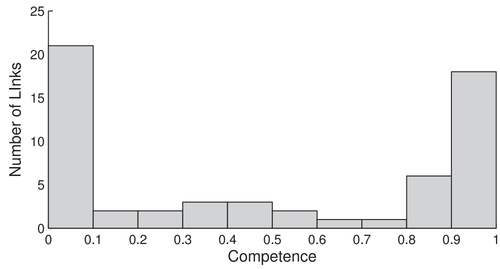


Fig. 5. Link distribution on link competence.

valuable results for network protocol designs. However, considering that the communication quality may vary significantly within seconds, it is not effective or energy efficient to use more probes for a more accurate estimation. The competence metric is new in focusing on characterizing long-term stability of the communication quality at a desired level. It is a complementary technique to previous link estimations. Given the different spatial and temporal patterns of different links, it is beneficial to use the long-term characterization of communication qualities. We note that long term and short term mentioned here are relative to the sampling period. This work is motivated by the empirical observations obtained from real system experiments, as shown in previous sections.

The long-term characterization is represented by a large smoothing factor, such as $\alpha = 0.9$, in the EWMA filter. We note that EWMA is just one of various mathematical techniques [Montgomery 2005] to emphasize long-term quality. On the other hand, because wireless communication quality can be highly variable, competence uses two bounds $[T_{lower}, T_{upper}]$ to specify a desired performance level, allowing small variation of the signal between specified bounds. In other words, this range eliminates any insignificant changes of quality.

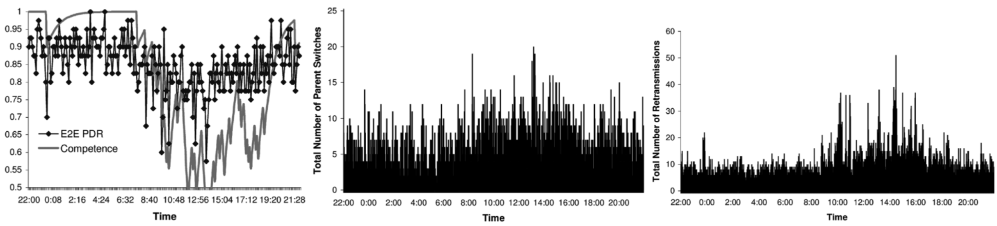
The distribution of links from our first experiment on the competence metric is plotted in Figure 5, with a specified link quality range [80%, 100%] and $\alpha = 0.9$. Here, 30.5% of the links are competent ($competence \geq 0.8$), and 69.5% of the links are not. These competent links have stable qualities within the specified bounds.

We also employ settling time, which is another important metric adopted from the classical control theory [Hellerstein et al. 2004], to quantify transient performance for wireless sensor networks. Settling time represents the amount of time a performance measure takes to deviate from and then return to a desired performance level. It quantifies a system's capacity to react to changes and return to normal performance level in the time dimension, especially when feedback control designs are applied. We use reliability as an example to demonstrate how settling time st is calculated in Equation (2):

$$
\begin{aligned}
& st = t_2 - t_1, t_2 > t_1 \\
& y(t_1), y(t_2) \in [T_{lower}, T_{upper}] \\
& \wedge \forall t \in (t_1, t_2), y(t) \notin (T_{lower}, T_{upper}).
\end{aligned} \tag{2}
$$

The β factor [Srinivasan et al. 2008] is a recent metric to quantify the correlations among successes and failures of transmissions at packet level. Different from the β factor, settling time focuses on performance resilience at a desired level with unexpected disturbances.

To explain the importance of stability and transient performance, we consider the example of VigilNet [He et al. 2004], which is a military surveillance system deployed on battlefields. In this application, data packets are required to be delivered to a



(a) E2E PDR over 24 Hours (b) Total Number of Parent Switches (c) Total Number of Retransmissions

Fig. 6. Performance evaluation of MultiHopLQI over 24 hours.

base station with a bounded reliability, say [80%, 100%]. This range is much bigger than the range defined by steady state error in classic control theory, and this range is required by the application. The lower PDR bound is chosen for guaranteeing a specified surveillance quality. For tracking mobile targets, data packets must be delivered above a certain rate. If the PDR is less than a lower bound, say 80%, important traces of the target may be missing. Moreover, missing important traces may lead to the inability to distinguish two targets moving closely together. Although the upper PDR bound can be set as 100%, most of multihop communication paths in this application are set lower for the sake of energy efficiency. Achieving perfect quality consumes significantly more transmission energy than a reasonable communication quality (95%) due to significantly increased control overhead. In this scenario, a reasonably good communication quality meets the application goals. Similar ideas apply for a number of environmental data collection applications [Girod et al. 2006; Selavo et al. 2007]. Generally, perfect communication quality is unnecessary for these applications as long as constant good performance quality is achieved. An acceptable settling time is also required for VigilNet to successfully capture the traces of a target in case the system performance is compromised or disturbed. If the settling time is too long, a high speed target on the edge of the surveillance area may pass across without being detected. In other applications, the settling time is also an important measure of how consistently the system can perform under significant changes.

We conducted another set of experiments to study the performance of existing protocol MultiHopLQI [2006] under the presence of unstable links. We ran the default configuration of MultiHopLQI on our indoor testbed for 24 hours. There were eight source nodes, each generating one packet per 10 seconds. These packets were sent to a base station via multihop paths. As suggested by previous studies [Paek and Govindan 2007; Kim et al. 2007], this traffic load should not cause packet loss due to queue overflow.

We made three observations from this experiment:

—The E2E PDR varies significantly, especially during the daytime, as shown in Figure 6(a). The plotted data represents the E2E PDR from a source to the sink. In this example, we observed that E2E PDR is around 90% in the evening hours. However, between the hours of 8 AM and 8 PM, the PDR drops to around 75%, with dips as low as 55% to 60%. Given the desired communication range [80%, 100%], the E2E competence measure is poor during the day. This result clearly shows that current link estimation is not effective when link qualities change dramatically. When these links are selected for routing, they may not be discarded the moment their qualities drop dramatically. As a result, E2E communication quality drops.

—The total number of parent switches increases significantly during the day, as shown in Figure 6(b). This result implies that nodes do not stick to the good and stable links but often choose unstable links. When the qualities of unstable links drop, the number of parent switches increases. Furthermore, cascading route changes occur: new routes can cause interference with other nearby routes, triggering even more packet losses, energy consumption, and route changes. Such route changes do not improve but degrade the E2E PDR. These cascading route changes should be avoided in network protocol design for highly dynamic networks. Figure 6(c) presents the total number of retransmissions. As a result of selecting unstable links and cascading route changes, the number of retransmissions required during the daytime increases substantially, which consumes more energy.

—This E2E PDR deviated from the desired range 27 times. For example, a drop of E2E PDR occurs between 1 PM and 2 PM, which lasts for about 60 minutes. Overall, the stability and transient performance of current protocol is not satisfactory.

4. COMPETENCE-ENHANCED ROUTING

In this section, we explain how we adapted the distance vector (DV) algorithm to exploit the competence metric for improved route performance in highly dynamic environments. First, we review the DV algorithm, which is based on a cost function describing the resource needed to perform an operation, such as energy. Let us denote the cost of a link from node i to node j as A_{ij} and the cost of the minimum cost route from node i to the destination as B_i. The DV algorithm can efficiently calculate B_i for all nodes by first having all nodes i calculate A_{ij} for all neighbors j. Then, each node chooses its *parent* node to be the neighbor k that minimizes that value $B_i = A_{ik} + B_k$. The DV algorithm can be performed in a distributed fashion by having each node i broadcast its own estimate of B_i every time it changes. The algorithm starts when the sink node broadcasts the value $B_{sink} = 0$. All neighbors of the sink estimate their own values of B_i, and the process repeats until the values at all nodes converge. If we define A_{ij} to be the ETX on the link from i to j, then B_i is the cost of the route from node i to the base station with the smallest number of expected transmissions. Here, ETX represents the widely used "expected number of transmission" metric [Woo et al. 2003] in wireless sensor networks, which typically employs an EWMA filter to emphasize on the short-term link quality.

The DV algorithm can easily be adapted to exploit the competence metric. We propose two solutions: (1) a node can choose its parent k as the node with the most competent link from among all nodes j with low values B_j, or (2) a node can choose its parent k as the node with lowest value of B_j from among all nodes j to which it has a competent link. The choice between these schemes depends on application requirements, as well as the quality of links and the dynamics of the environment.

In scheme 1, a node periodically selects a neighbor k to be its parent node. This neighbor is selected via two steps. First, the node selects the lowest value $\hat{B}$ from among the values of all neighboring nodes. It then selects the set of all neighbors j with values close to the lowest value: $j : B_j \leq R \cdot \hat{B}$, where R is a specified range parameter, such as 120%. Second, the node selects the neighbor with the highest competence value among this set of neighbors as the forwarding node.

In scheme 2, a node first periodically selects the highest competence value of all of its neighbor nodes. It then selects the set of all neighbors j with competence values close to the highest one: $j : Competence_j \geq T \cdot Competence_{lowest}$, where T is a specified range parameter, such as 80%. Subsequently, the node selects its parent to be the neighbor with the lowest value B_j from among all nodes j in this set.

Routing scheme 1 uses the competence metric to break ties between routes that are otherwise equivalent in terms of cost and performance. The definition of a tie is defined by the parameter R. This algorithm allows the routing scheme to tolerate small variations as specified by the competence bounds but reacts to big variations via parent switches. This scheme should be used when performance is more important than robustness to network dynamics. This would likely be true in networks where network dynamics affect the routing algorithm but do not overwhelm it.

Routing scheme 2 chooses routes that are the least susceptible to network dynamics and breaks ties using the cost metric ETX. The definition of a tie is defined by the parameter T and can be used to find cheap routes as long as they have competence levels similar to the best route. This scheme should be used when competence is more important than routing cost. Such would be the case in highly dynamic networks, or when guaranteeing performance at all times is more important than maximizing performance.

There are other factors that may influence the performance gain of our routing design, such as the network density. In a sparse network, nodes may not have many competent links available to choose from among the low-cost links, but even in the worst case, the performance will be similar to the original algorithm without competence consideration. We also note that stable routing may potentially put traffic loads on competent routes, causing these nodes' batteries to deplete sooner.

5. ROUTE MAINTENANCE FRAMEWORK

In a sparse highly dynamic network, the number of competent links can be limited. The routing structure needs to use some other links. Efficiently making use of these links is the key to reduce unnecessary route changes and improve reliability. With link quality improvement techniques [Lin et al. 2006; Polastre et al. 2004; Cao et al. 2007; Katti et al. 2006], some links can become competent links or more competent than before. However, those techniques have their cost in terms of energy consumption and overhead. We found that the cost is associated with long-term quality of links at a specified level. Due to the high overhead to deal with variations, maintaining a stable link within certain bounds costs less than maintaining an unstable link within the same bounds. Actually, maintaining a stable link within high bounds can cost less than maintaining an unstable link within low bounds. Therefore, we propose a route maintenance framework based on competence to maintain routes and optimize maintenance cost. Given a selected path, this framework globally assigns different performance levels to links along an active path and locally maintains assigned performance levels. This two-level maintenance design is both necessary and efficient, as a single link layer solution would lead to (1) local nonoptimal decisions, (2) unbalanced cost at different links, and (3) fluctuating E2E performance due to uncoordinated control along a path.

In competence-enhanced routing, we use values of the competence metric as a routing metric, whereas in the route maintenance framework, we use the bounds of the competence metric as parameters for E2E performance control. The architecture of this maintenance framework is shown in Figure 7. The control modules are located at two layers: the network layer and the link layer. At the network layer, there are a performance monitor with specified requirements, a competence controller, and a route monitor. At the link layer, there are a link monitor and controllers. We focus on reliability as the performance requirement in this work. The performance requirement consists of specified competence bounds on E2E PDR. Given the specified bounds, the performance monitor calculates competence based on observed E2E PDR. When competence drops below a certain threshold, E2E PDR error is passed to the route controller. With an E2E feedback loop along this path, the route controller collects costs from each link and allocates the stable link performance requirements to optimize transmission energy consumption. Then the link performance requirements are injected to link control modules along this path. At the link layer, both the transmission power control and the retransmission control are used to enforce the link performance requirements, which are adaptive and low-cost solutions to control single-link reliability.

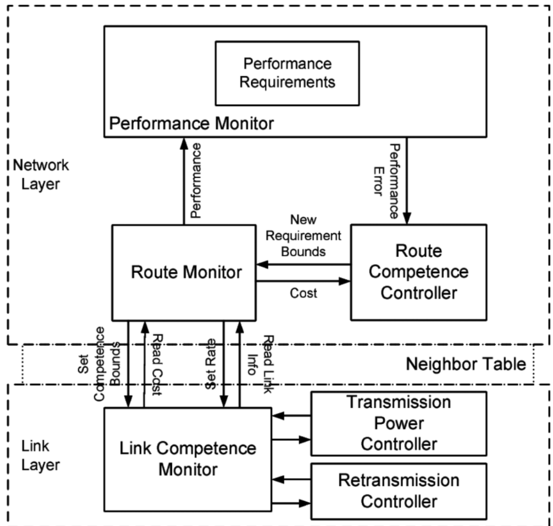


Fig. 7. Control architecture.

Table I. Main Terms

Term	Definition
A_{Time}	Time required for one attempted transmission
tp_i	Transmission power level on link i
p_i	Quality of link i
I_i	Interference on link i
T	E2E delay deadline
f	Frame size
l	Preamble size
$NumD_{ij}$	Number of packets transmitted from node i using the power level j
TE_j	Transmission energy consumed per bit
LD	Length of data packet

The main terms that are used in this work are described in Table I.

5.1. Link Layer Competence Maintenance

We use power control and retransmission control as two general techniques for the link layer maintenance design. The link layer control design is shown in Figure 8. The controlled variables are the transmission power level and the number of link-level retransmissions. These two controllers work independently. The link competence monitor measures PDR competence. If PDR competence drops below a certain threshold, control actions are triggered. The set points and bounds for PDR are specified by network-level maintenance.

$$PDR(x) = 1 - [1 - p]^x \tag{3}$$

$$u_{rt}(t) = u_{rt}(t-1) + \frac{\log e_{PDR}(t-1)}{\log(1 - p(t-1))} \tag{4}$$

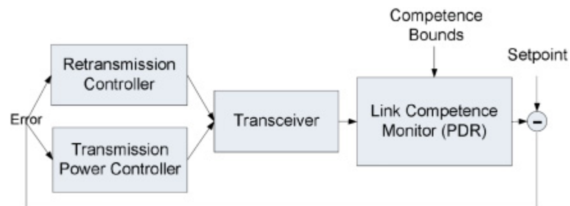


Fig. 8. Link layer maintenance.

The retransmission control model is based on Equation (3). The p stands for the probability of successful transmission for a single attempt. We assume that the success probabilities of transmissions are independent of each other. The input x is the maximum number of retransmissions. The controller form is derived from Equation (3) and shown in Equation (4). $u_{rt}(t)$ is the representation of the maximum number of retransmissions x. This controller takes PDR error $e_{PDR}(t)$ as input and adjusts the number of retransmissions $u_{rt}(t)$ as output.

$$RSS = \beta \cdot tp + \gamma \tag{5}$$

The goal of transmission power control is to achieve high $p(t)$ while saving transmission energy. A control model designed in Lin et al. [2006] is shown in Equation (5). The RSS refers to the signal strength of the link, tp represents the transmission power level applied at the transmitter of the link, and β and γ are link-specific time-varying parameters that depend on the environment.

$$\begin{aligned} u_{tp}(t) = u_{tp}(t-1) + K_p[e_s(t) - e_s(t-1)] \\ + K_i e_s(t) + K_d[e_s(t) - 2e_s(t-1) + e_s(t-2)] \end{aligned} \tag{6}$$

Based on this adaptive control model, we extend it using a proportional-integral-derivative (PID) control [Hellerstein et al. 2004] shown in Equation (6). This controller takes signal strength error $e_s(t)$ as input and adjusts transmission power level u_{tp}. K_p, K_i, and K_d are proportional, integral, and derivative gains of the controller. To obtain the lowest settling time and highest reliability, we tuned this PID controller on different unstable links during unstable periods in the daytime and stable periods at nighttime. We obtained two different sets of gain values in these two periods. The integral gain tuned for the unstable periods is noticeably larger than that of the stable periods, which compensates for the quality fluctuations and optimizes settling time. The transmission power controller uses a gain scheduling approach. The link competence monitor triggers the switches of gain values. When competence measure becomes lower than the setpoint, the controller starts using the gains for the unstable period. When competence measure becomes higher than the setpoint, the controller employs the gains for the stable period. We also use a conservative threshold for robust link quality estimation and nonlinearity of power control in the indoor environment suggested by Hackmann et al. [2008] and Souryal et al. [2007].

5.2. Network Layer Competence Maintenance

For the good and stable links, high and stable performance is maintained with a very small cost using link layer maintenance. However, maintaining equally high performance for the unstable links is costly due to control overhead. To maintain E2E performance while optimizing total transmission energy consumption, our algorithm assigns competence bounds to links along an E2E path.

$$E2E\ \ PDR(k) = \prod_{i=1}^{k} PDR(x_i) \tag{7}$$

The relation between E2E PDR and link PDR along a k-hop path is presented in Equation (7). x_i represents the number of transmissions at hop i. This equation also indicates the relation between the bounds of E2E competence and link-level competence.

$$Cost(x_i) = C_{tp_i} \cdot \sum_{i=1}^{x_i} i \cdot (1-p)^{(i-1)} \cdot p \tag{8}$$

In Equation (8), the expected energy consumption of the link layer control at link i is represented as a cost function of the transmission power level and the number of retransmissions x_i. We note that C_{tp_i} is a constant for an attempt of transmission (including multiple retransmissions). The value of C_{tp_i} depends on the transmission power level used.

$$\sum_{i=1}^{k} (Cost(x_i) + overhead_i) \tag{9}$$

The total transmission energy consumption of an E2E path is presented in Equation (9). There are two types of costs for each link. The $Cost(x_i)$ is the energy consumption for transmission on link i given the number of transmission x_i, as shown in Equation (8). The $overhead_i$ is the energy consumption for control overhead on link i, such as energy consumption for feedback packets, which can be measured at each link.

Our goal is to minimize the total cost along a path while meeting the specified performance level. When the path is first established, we give every link on the path the same competence bounds. To optimize total transmission energy consumption, high competent links should have high bounds and low competent links should have low bounds. Mathematically, this problem is presented as follows:

$$\begin{gathered} \min \sum_{i=1}^{k} (Cost(x_i) + overhead_i) \\ s.t.\ E2E\ \ PDR(k) \geq F \\ 0 \leq p_i \leq 1, x_i \in \mathbb{N}. \end{gathered} \tag{10}$$

F specifies the desired E2E PDR. This is a nonlinear optimization problem that can be approached by KKT conditions [Boyd and Vandenberghe 2004]. We skip the construction and calculation details. However, the complexity of this problem after applying KKT is still exponential. Fortunately, we find that functions $PDR(x_i)$ and $Cost(x_i)$ have an approximate linear relation in their small range. We plot the relations in Figure 9. Each curve in this figure represents the relation between PDR and $Cost$ at a fixed p. We can use a linear model as shown in Equation (11) to describe this relation, especially when p is larger than 0.5, which is the range of link qualities of the most useful links:

$$Cost(x_i) = a_i \cdot PDR(x_i) + b_i. \tag{11}$$

In this linear model, a_i and b_i are functions of p_i. Given p_i, values of a_i and b_i are fixed using a least square approximation. Based on this linear model, the complexity of this optimization problem is now linear rather than exponential. As a result, we can tell that when $a_i \cdot PDR(x_i)$ are equal to each other, the total cost is minimized. The

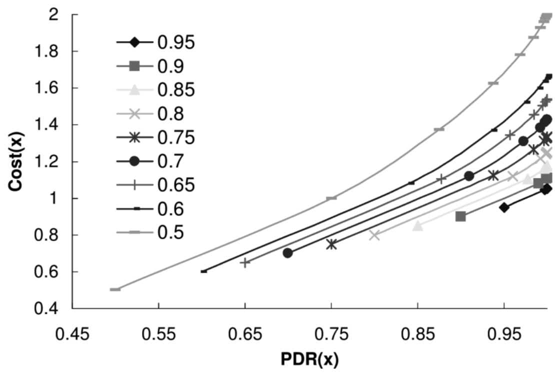


Fig. 9. Approximate linear relation.

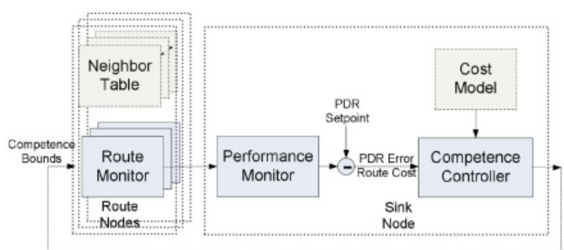


Fig. 10. E2E feedback loop.

minimal cost is $\frac{\sqrt[k]{F\prod_{i=1}^{k} a_i}}{k}$. Correspondingly,

$$PDR(x_i) = \frac{\sqrt[k]{F\prod_{i=1}^{k} a_i}}{a_i}. \tag{12}$$

Based on Equation (12), nodes can calculate their new bounds: desired PDR. To do this, every node needs to know its a_i and $\prod_{i=1}^{k} a_i$. a_i is obtained from a local table storing values of a and b, given p. The latter can be calculated and delivered to nodes via a feedback loop, as shown in Figure 10. In this feedback loop, a control packet is sent from source node to sink node periodically. This packet is used to calculate $\prod_{i=1}^{k} a_i$ hop by hop. The performance monitor at the sink node monitors the E2E PDR and compares it with the specified PDR level. If there is an error, the performance monitor notifies the competence controller about the current error. The competence controller takes $\prod_{i=1}^{k} a_i$ and the path length as inputs and calculates $\sqrt[k]{F\prod_{i=1}^{k} a_i}$. This value is then sent back as feedback to every node along the path via control packets. Nodes then calculate desired PDR bounds according to Equation (12). Finally, link control maintenance can calculate the maximum number of transmissions x_i based on Equation (12). Our control-based design has limitations. For instance, control contention may affect system performance. However, the control contention rarely happens when traffic load is low, which is the case for many wireless sensor network applications. Our evaluation demonstrates that the control-based approach works well in real systems. To address another potential concern, although distributed control consumes resources and introduces delay in large-scale networks, in most existing wireless sensor networks the number of nodes reporting to the same base station is less than a few hundred. When

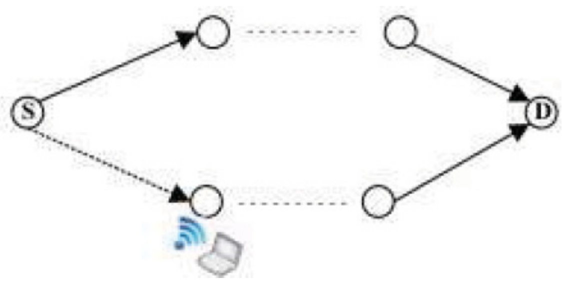


Fig. 11. An example of WSN topology.

multiple routes pass through the same link, the link and network layer's maintenance keep the parameters for every route.

5.3. Stable Delay Routing

The aforementioned competence-enhanced routing increases the network performance in terms of consistent and high-quality communication links in dynamic environmental conditions. In this section, we continue to adapt the network performance in terms of E2E delay to meet stable communication requirements for many applications [Ahmad et al. 2008]. Sparse wireless sensor networks have a limited number of competent links, and for low rate transmission networks, the links delay is mainly affected by the transmission time rather than queuing time. Assuming that the interference and the power levels are independent on each link throughout the path (TDMA-based coordination schemes are used to eliminate interference caused by concurrent transmissions), such that each link L_i has its interference level I_i, transmission power level tp_i, and link quality level p_i. There is a requirement to make E2E delay on this path no more than value T. For example, note the case in Figure 11. Some of the network paths are stable, whereas others are unstable (i.e., suffer from interference). The goal in this example is to send packets from source node to destination node using a stable path. In this example, the stable path delay equals the sum of delay that occurs on the links using this path. In general, the delay on a path that passes through node 1 to node k at time period t is equal to

$$dp(t) = \sum_{i=1}^{k-1} d_i(t), \tag{13}$$

where $d_i(t)$ is the amount of time required to transmit one packet from node n_i to node n_{i+1} in the path at time period t.

Many applications require desired specified E2E delays to achieve high QoS [Chipara et al. 2006]. To meet the E2E requirement, we utilize transmission power control, as it is a well-known mechanism to improve the link performance. The adjustment in power level adapts link quality to the variation of the surrounding environment conditions [Lin et al. 2006].

The cost function of the expected transmission energy consumption from node i to satisfy link L_i requirements is represented in Equation (8), where the power level C_{tp_i} and the number of transmissions x_i do not have to be the same on each link throughout the path.

The delay that occurs on the link L_i can be expressed as the required time for one attempt transmission (i.e., A_{Time}) times the expected number of transmissions required to transmit a packet successfully. Formally, we can model the delay on the link L_i as

$$d_i(t) = A_{Time} \sum_{i=1}^{x_i} i(1-p_i)^{i-1} p_i. \tag{14}$$

The total transmission energy consumption of an E2E path has two parts: the energy consumption for transmission on link i given the number of transmissions x_i, and the $overhead_i$, which is the energy consumption for control overhead on link i occurring from the consumed energy for feedback packets. Our goal is to minimize the total cost along a path while meeting the specified E2E path delay requirement. The total cost along the path can be formulated as

$$\begin{gathered} \min \sum_{i=1}^{k-1} (Cost(x_i) + Overhead_i) \\ s.t. \sum_{i=1}^{k-1} d_i(t) \leq T \\ 0 \leq p_i \leq 1, x_i \in \mathbb{N}, \end{gathered} \tag{15}$$

where T specifies the desired E2E delay. In addition, there is a nonlinear optimization problem. Following the same approximation approach used in solving Equation (11), the path delay is equal to

$$dp(t) = \sum_{i=1}^{k-1} A_{Time} \cdot \frac{Cost(x_i)}{C_{tp_i}}. \tag{16}$$

When $dp(t)$ is equal to the E2E threshold T, then the required energy amount is the lower bound of optimal energy amount. Thus, the minimal cost on each link $Cost(x_i)$ is

$$Cost(x_i) = \frac{T}{A_{Time} \cdot \sum_{i=1}^{k-1} \frac{1}{C_{tp_i}}}. \tag{17}$$

Based on this minimal cost, the corresponding $PDR(x_i)$ is

$$PDR(x_i) = \frac{T - A_{Time} . b_i . \sum_{i=1}^{k-1} \frac{1}{C_{tp_i}}}{\alpha_i . A_{Time} \cdot \sum_{i=1}^{k-1} \frac{1}{C_{tp_i}}}. \tag{18}$$

Based on Equation (18), each node in the path can calculate the desired $PDR(x_i)$ to satisfy the E2E path delay requirement.

Power control algorithms have been designed to adjust transmission power levels to adapt to dynamic environmental changes. Efficient power consumption requires a proper power level, which is used by nodes to transmit packets. The design of network-level power control aims to generate appropriate power levels on each sensor node along a routing path.

Many applications require E2E delay to be stable, and the delay needs to be assured throughout the path. With our route maintenance model, the power level on each transmitting sensor node can be adjusted to satisfy the specified E2E delay requirement. Figure 12 shows the feedback loop. The adjustment process requires each transmitting sensor node to be informed about the values of $\frac{T}{A_{Time} \cdot \sum_{i=1}^{k-1} \frac{1}{C_{tp_i}}}$, whereas α_i can be obtained from a local table that stores the values of α_i given p_i.

When the calculated E2E delay exceeds the threshold T, delay error is calculated. A feedback packet containing the delay error and path-related information is issued;

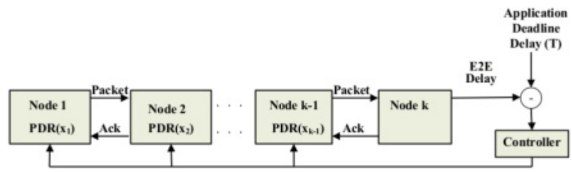


Fig. 12. E2E delay control.

then, based on the feedback, each sensor node along this path calculates the required PDR value using Equation (18).

5.4. Integration

Thus far, we have described multiple techniques at different layers to maintain stable network performances. These techniques include the competence-enhanced stable routing algorithm in Section 4, the link quality control in Section 5.1, the network E2E quality control in Section 5.2, and the stable delay control in Section 5.3. These techniques span multiple layers in the TCP/IP (6LoWPAN) architecture. Specifically, the stable routing algorithm lies in the network layer and the link quality control lies in the link layer, whereas the E2E reliability and delay control lies in the transport layer.

Although all techniques utilize the competence metric, some or all techniques can be applied to different scenarios according to the available information at each node. The stable routing algorithm and the long-term link estimation can directly be incorporated with existing protocols. The competence metric can obtain the stability of a link as long as there are packets sent among neighbors over a long period of time. However, when few packets are exchanged among neighbors, the proposed algorithm may not work better than existing solutions. The link layer control can maintain link quality on an individual link. It can also work together with the transport layer E2E performance control for specified network performances. However, the E2E reliability and delay control algorithms of the transport layer cannot work without the support of link layer control solutions.

The Collection Tree Protocol (CTP) has demonstrated great performances in several wireless sensor testbeds [Gnawali et al. 2009]. The CTP protocol uses data path validation and adaptive beckoning to deal with link dynamics. Since such a design can capture rapid link quality changes and select routing paths accordingly, it can achieve high performances even with unstable links. To achieve even more stable and energy-efficient performances, we can integrate CTP with our design. We can add the competence metric as a link metric together with the ETX. When selecting a routing path, the routing algorithm will consider the competence metric besides ETX. The competence metric is useful when dynamic link qualities occur on many links, causing increasing beacon rates and frequent route changes in the network. In this scenario, our design provides stable and good enough performance with low overhead in comparison to CTP, which pursues high performance with high overhead.

6. EVALUATION

6.1. E2E PDR Analysis for Stable and Unstable Periods

The most widely used link metric is the ETX [Bicket et al. 2005; Woo et al. 2003; Fonseca et al. 2007; MultiHopLQI 2006]. Many popular data collection protocols combine the DV algorithm with link estimation techniques for wireless sensor networks. For example, the MintRoute [Woo et al. 2003] algorithm uses eavesdropping and an EWMA operator to estimate the probability of successful transmission over each link.

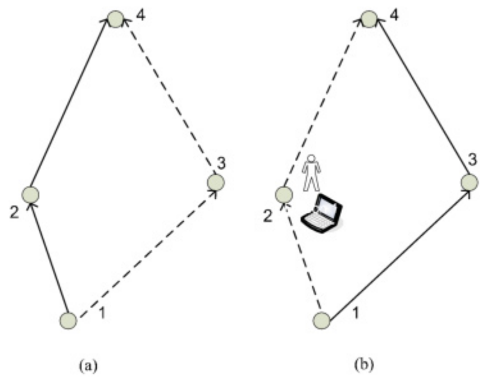


Fig. 13. Experimental topology.

It then translates these probabilities into ETX values and uses a DV to find routes that minimize the E2E ETX. The CTP [Gnawali et al. 2009] augments MintRoute with explicit beacon messages to estimate link quality even when traffic rates are low. MultiHopLQI uses the link quality indicator (LQI) defined by IEEE 802.15.4 [1999] as an instantaneous link quality estimate, helping it react more quickly to changes in link quality [MultiHopLQI 2006]. In Fonseca et al. [2007], a hybrid estimator integrates routing feedbacks and link estimates together to achieve high reliability. These protocols are currently state of the art in data collection for wireless sensor networks and have been shown in empirical studies to have very high PDRs.

Based on DV routing protocols, such as MultiHop LQI or MintRoute [Woo et al. 2003], we have implemented a DV routing algorithm in TinyOS as the baseline, a competence-enhanced DV routing (C-DV), and a DV routing (MC-DV) that is both competence enhanced and maintained. The DV routing algorithm adopts an ETX-based link estimator using the EWMA filter, which is widely used in existing protocols. C-DV adopts routing scheme 1 described in Section 4. We used the fixed transmission power level 20 of the CC2420 radio for the DV and C-DV setup. We did not use radio duty cycling. In MC-DV, the E2E PDR bounds are set as [80%, 100%]. The decay factor α for competence calculation is 0.9. The implementation of MC-DV takes 22772B ROM and 4238B RAM.

We have conducted controlled experiments at night with four nodes. The topologies of these experiments are shown in Figure 13. Node 1 sends 1,000 packets to base station 4 at a rate of 1 pkt/sec. First we ran three algorithms when there was no interference or human activity in this area, as shown in Figure 13(a). Then we ran another test with intentional interference near node 2, as shown in Figure 13(b). Node 2 is hanging 4.5 feet high on the top of a cubicle and beside an office door. A student used Wi-Fi to download files in the cubicle and walked in and out using the office door from time to time.

The E2E PDRs are shown in Figure 14. From this figure, we can see that in the stable periods when there is no interference or human activity, the three algorithms have almost the same PDR. The E2E PDR of path 1-2-4 is 99.3%, and the E2E PDR of path 1-3-4 is 96.2%. All three algorithms select the path 1-2-4 all the time, which had constant good communication quality. In this case, the use of long-term estimation and maintenance do not make a difference. However, in the unstable periods, three algorithms have different PDRs. This is because with interference and shadowing, the E2E PDR of path 1-2-4 was highly variable, ranging from 100% to 20%. However, the E2E PDR of path 1-3-4 has a little variation (around 4%) due to weak interference

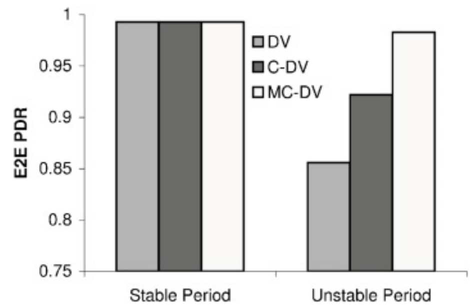
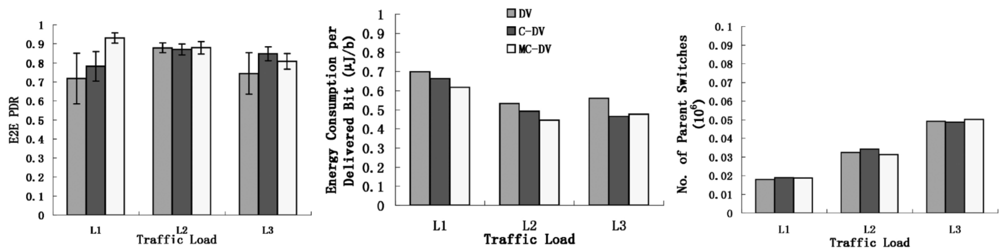


Fig. 14. Experimental result.

(nodes 2 and 3 are located at the opposite sides of an office). DV keeps oscillating between the routing path 1-2-4 and 1-3-4 due to the short-term estimation. As a result, when path 1-2-4 was selected and interference occurred near node 2, packets were lost. The routing path of C-DV converged at 1-3-4 after a few oscillations and then the PDR settles around 94.2%. In the MC-DV experiment, when path 1-2-4 was selected, the route maintenance increased transmission powers and number of retransmissions at nodes 1 and 2. However, the PDR was still bad when interference occurred close by. Then, after MC-DV switched to path 1-3-4, its PDR was improved when transmission powers and retries were increased at nodes 1 and 3. In addition, the routing path converged to 1-3-4 quickly, and the E2E PDR settled at around 99%. From this controlled experiment, we conclude that (1) the long-term estimation used in competence helps choose stable links and improves PDR; (2) route maintenance helps improve PDR on links that are weakly interfered or shadowed; and (3) in networks with only stable links or only strongly interfered unstable links, the benefit of long-term estimation and route maintenance is limited.

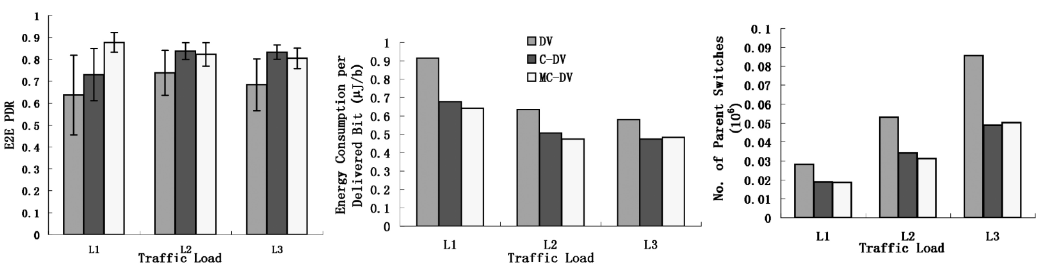
We also conducted nine multihop experiments in the testbed with 48 T-Mote Sky nodes, with each experiment lasting 24 hours. We used three kinds of periodic traffic loads for communications from sources to a sink, which are typical for environmental monitoring. In traffic load 1 (L1), there were 3 source nodes, each of them sending a data packet every 20 seconds. In traffic load 2 (L2), there were 8 source nodes, each of them sending a data packet every 20 seconds. In traffic load 3 (L3), there were 8 sources, each sending a packet every 10 seconds. We note that such traffic loads do not cause message queue overflow. In this experiment, we focus on studying the performance difference between stable periods at night and unstable periods during the day. We divided the data obtained in each 24-hour experiments into two parts, corresponding to a stable network period from 8 PM to 8 AM and an unstable network period from 8 AM to 8 PM, and plotted them in Figures 15 through 19.

In Figure 15(a) and Figure 16(a), we have plotted the observed average E2E PDR. We have also plotted corresponding standard deviations over 12 hours. We have drawn four main observations from these figures: first, DV has a much higher E2E PDR and a smaller standard deviation in the stable periods than the unstable periods, whereas the E2E PDRs of C-DV and MC-DV demonstrate much smaller performance differences in both stable and unstable periods. Previous evaluations have shown that ETX-based routing algorithms can achieve good performances in stable networks [Woo et al. 2003]. Our evaluations have confirmed this. For example, the E2E PDR of DV with traffic load 2 (L2) is 87.9% over 12 hours, including 8 continuous hours above 90%. However, E2E PDRs of DV in the unstable periods drop significantly. This result shows that previous solutions do not work well in highly dynamic networks. Second, the differences between E2E PDRs of C-DV in both stable and unstable periods and



(a) Average E2E PDR (b) Energy Efficiency (c) Parent Switch

Fig. 15. Evaluation in stable periods.



(a) Average E2E PDR (b) Energy Efficiency (c) Parent Switch

Fig. 16. Evaluation in unstable periods.

the differences between E2E PDRs of MC-DV in the same periods are less than 5%. The standard deviations of C-DV and MC-DV are also much smaller than those of DV in all cases. This result demonstrates that C-DV and MC-DV achieve stable and high E2E PDR in highly dynamic networks, outperforming DV. More specifically, with light traffic L1, the average E2E PDR of C-DV is above 80% for more than 80% of the time. The average E2E PDR of MC-DV is above 80% for more than 99% of the time. Third, MC-DV outperforms DV and C-DV with light traffic L1. This demonstrates the benefit of route maintenance for interference-free streams. Two factors that contribute to performance of MC-DV with different traffic are interference among streams and control contention. Interference among streams affects the qualities of links. There were three streams with L1 load, and 8 streams in L2 and L3 loads. In addition, the maintenance introduces extra control packets at both link and network layers. With a heavier traffic load, the links near the base station may become unstable, whereas with light traffic, the interference is free and stable links are selected. On the other hand, control contentions may occur at streams near each other. For example, power controllers on two parallel links increase power alternately when their transmissions interfere, causing degrading PDR. These issues indicate that MC-DV may not work very well with heavy traffic and large-scale networks. Fourth, in our experiments, we found that not all observed E2E PDRs demonstrate obvious improvement. For other sources, we have observed smaller performance improvement than that of DV, which suggests that the improvement of competence-enhanced routing depends on the density of competent links in the network. If a node has no competent links that it can use, the stability of performance will not improve much. Overall, our competence-based link characterization and feedback control–based stabilization are critical for achieving better network performances in dynamic wireless sensor networks.

The 24-hour E2E PDR of all three algorithms under light traffic L1 is shown in Figure 17. From this figure, we can see the average E2E PDRs of DV, M-DV, and

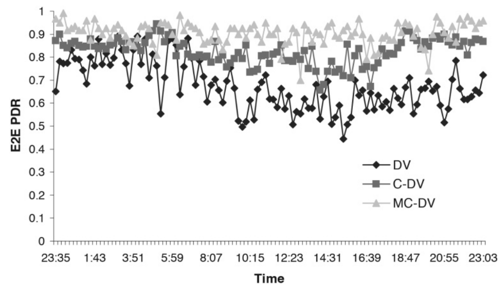
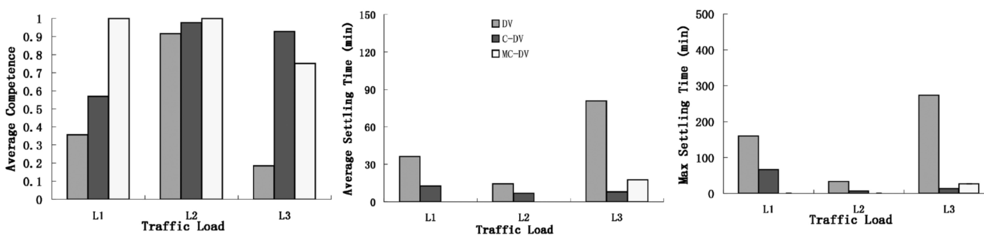


Fig. 17. E2E PDR in 24 hours.



(a) Average Competence (b) Average Settling Time (c) Max Settling Time

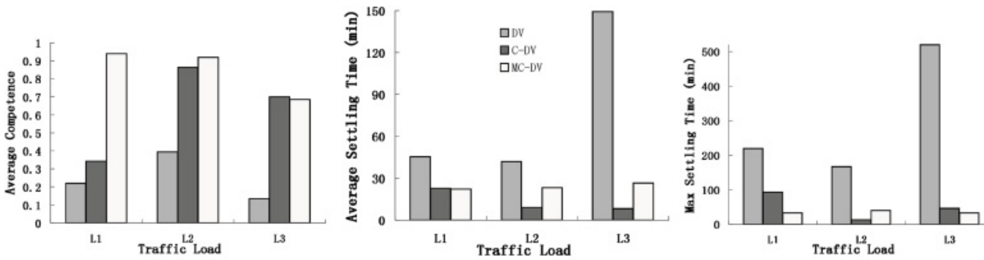
Fig. 18. Evaluation in stable periods.

MC-DV and their variations over 24 hours. The transition between stable period and unstable period can be observed at DV, as its E2E PDR starts decreasing around 8 AM and remains low until late afternoon. C-DV and MC-DV, on the other hand, have much more stable E2E PDRs over time, although we can still observe a small performance decrease during the stable period.

The transmission energy efficiency is presented in Figure 15(b) and Figure 16(b). Transmission energy is estimated based on the total number of transmissions, the packet length, the transmission power level used for each transmission, and the control overhead. Several interesting observations can be made from these figures. First, the energy consumption per delivered bit of all algorithms in unstable periods is higher than stable periods because that many transmissions in unstable periods are wasted without successfully delivering the packets. Then, as shown in Figure 16(b), in unstable periods, the energy consumption per delivered bit of C-DV and MC-DV is much lower than that of DV. This result suggests that C-DV and MC-DV are more energy efficient than DV in both stable and unstable periods.

We have plotted the total number of parent switches in Figure 15(c) and Figure 16(c). We have two main observations. First, the numbers of parent switches of all three algorithms are similar in stable periods. This demonstrates that the characterization of stable links does not increase the number of parent switches in stable periods. Second, in the unstable periods the numbers of parent switches of DV are much higher than these of C-DV and MC-DV. This result shows that C-DV and MC-DV have successfully decreased the traffic oscillation that DV suffers in the unstable periods. We also note that as the traffic load increases, interference caused by control packets in MC-DV may increase.

We have also calculated the average competence on E2E PDR of these three algorithms in both stable periods and unstable periods, and plotted them in Figure 18(a) and Figure 19(a). The value of average competence represents how well E2E PDR stays within the specified range in the long term. From these figures, we can see that average



(a) Average Competence (b) Average Settling Time (c) Max Settling Time

Fig. 19. Evaluation in unstable periods.

competence values of all three algorithms during stable periods are higher than corresponding ones during unstable periods. This confirms that interference and dramatic environmental changes influence the stability of E2E PDR within the specified range. On the other hand, the competence values of MC-DV and C-DV are much higher than DV during both stable periods and unstable periods. Furthermore, decreases of competence values of MC-DV and C-DV during unstable periods are much less than the ones of DV. These results demonstrate that our designs achieve more stable E2E PDR than the previous design. We have calculated the average settling times and maximum settling times of these algorithms, and plotted them in Figure 18(b) and (c) and Figure 19(b) and (c). The trends of average settling times and maximum settling times are similar. From these figures, the settling times are almost twice as long during unstable periods as they were during stable periods. Both C-DV and MC-DV have a much lower average settling time than does DV in all experiments.

6.2. E2E Delay Analysis

Woo et al. [2003] offer a probabilistic approach for quantifying path delay to serve real-time routing. Since the link-level delays are probabilistic rather than deterministic, this approach proposes a multitimescale adaptation routing protocol.

We conducted simulations with real data traces. In this simulation, each node can use different power levels to transmit packets; at the same time, each node faces a different background interference level.

In WSN, the link quality is affected by the surrounding environment conditions. For example, if the interference level increases, then the link quality is affected directly due to this increase. In this section, we are motivated to study the effect of different condition (i.e., interference levels).

In Figure 20, we can see that the measured delay of a path, which consists of three hops. For the stable period, we fixed the interference amount that affects the path nodes at $-76.53dBm$; for the unstable period, we varied the interference amount between $-76.53dBm$ and $-75.6dBm$. The average delay for stable period is around 7ms, and the average delay for unstable period is around 12.7ms. The standard deviation for the stable period appears to be much smaller than the standard deviation for the unstable period. The variation of interference has a major effect on the path delay. Regulated power transmission can lead to more stable delay behavior and simultaneously reduce the delay variation.

To evaluate energy efficiency of our design, we calculated the total transmission energy consumption of all nodes using the following equation:

$$CE = \sum_{i=1}^{k-1} \left(\sum_{j=min}^{max} (NumD_{ij} \times TE_j) \times LD \right), \tag{19}$$

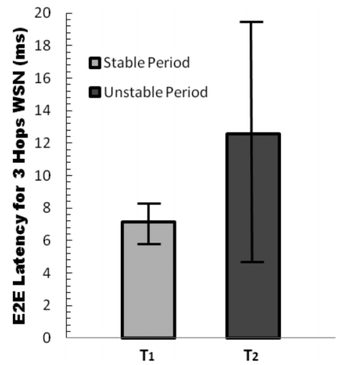


Fig. 20. Delay in stable and unstable periods.

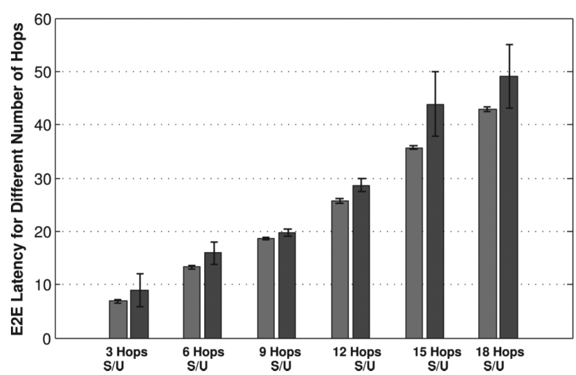


Fig. 21. Delay in stable period and unstable period with PC.

where i is transmitting node id, k is the path length, and j is the power level for the transmitting node. $NumD_{ij}$ is the number of packets transmitted from node i using the power level j, and TE_j is the transmission energy consumed per bit, which is based on Chipcon CC2420 low-power radios. LD is the length of the data packet, and we fixed it at 20 bits. In our simulation, the source node transmitted one packet each second under two different periods. In the first period (the stable period), we did not use any power control, whereas in the second period (the unstable period), we used our transmission power control solution.

Figure 21 shows the delay for the stable period and unstable period for different number of hops. For the 3-hops path in the stable period, the delay was around 7ms with a standard deviation of 0.4, whereas in the unstable period with power control to satisfy $T = 9$ms, the delay was close to 9ms with a standard deviation of 3.1. In the 6-hops path, the stable period delay was around 13.2ms with a standard deviation close to 0.4, whereas in the unstable period with power control to satisfy $T = 15$ms, the delay was 15.8ms with a standard deviation of 2. Using the 9-hops path, the stable period delay was close to 19.6ms with a standard deviation of 0.19, and for the unstable period with power control to satisfy $T = 20$ms, the delay was around 19.7ms with a standard deviation of 0.76. Finally, for the 12-hops path in the stable period, the delay was 25.7ms with a standard deviation of 0.34, and using an unstable period with power control to satisfy $T = 30$ms, the delay was around 28.7ms with a standard deviation of 1.2.

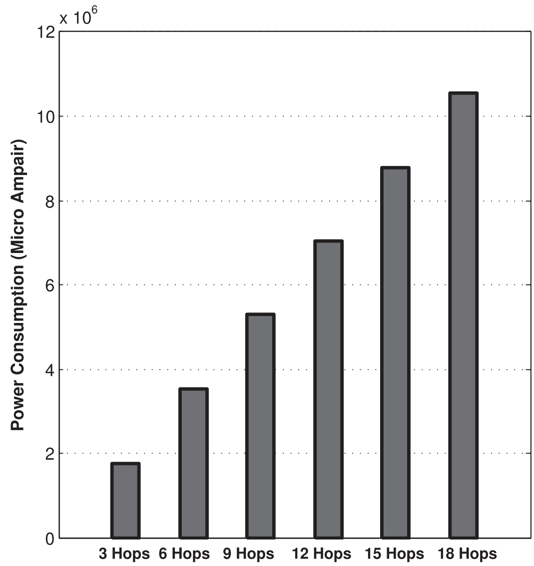


Fig. 22. Transmission energy consumption.

The results reveal that the standard deviations in stable periods are less than the standard deviations in the unstable periods, therefore suggesting that our design regulated transmission power levels of nodes to better meet the E2E requirements.

In Figure 22, we divided a 1-hour period into intervals of 5 minutes and then calculated the power consumption every 5 minutes for all nodes along the path. For the 3-hops path, the power consumption was around 1.8×10^6 microamps. In the 6-hops path, the power consumption was around 3.53×10^6 microamps. For the 9-hops path, the power consumption was around 5.3×10^6 microamps. Finally, for the 12-hops path, the power consumption was around 7×10^6 microamps. We can see that the total power consumption increases as the path length increases.

7. CONCLUSIONS

This article presents a competence metric to characterize the long-term communication quality. To achieve stable performances in E2E communication, we incorporate the competence metric into routing protocol designs. We also propose a feedback control framework that addresses dynamics at the link, network, and transportation layers. Our evaluations with 48 T-Motes have demonstrated that our design achieves satisfactory and stable network performances over time, outperforming existing protocols.

ACKNOWLEDGMENTS

We also thank Joseph Shen and Yongjin Lu for valuable discussions.

REFERENCES

T. Abdelzaher, Y. Diao, J. L. Hellerstein, C. Lu, and X. Zhu. 2008. Introduction to control theory and its application to computing systems. In *Performance Modeling and Engineering*. Springer, 185–215.

A. Ahmad, L. Abdul Latiff, and N. Fisal. 2008. Real-time routing in wireless sensor networks. In *Proceedings of ICDSW.*

P. Asare, D. Cong, S. G. Vattam, B.-G. Kim, A. King, O. Sokolsky, I. Lee, S. Lin, and M. Mullen-Fortino. 2012. The medical device dongle: An open-source standards-based platform for interoperable medical device connectivity. In *Proceedings of IHI'12*.

J. Bicket, D. Aguayo, S. Biswas, and R. Morris. 2005. Architecture and evaluation of an unplanned 802.11b mesh network. In *Proceedings of ACM MobiCom'05*.

S. Boyd and L. Vandenberghe. 2004. *Convex Optimization*. Cambridge University Press.

G. Buttazzo. 2006. Research trends in real-time computing for embedded systems. *ACM SIGBED Review* 3, 3, 1–10.

Q. Cao, T. Abdelzaher, T. He, and R. Kravets. 2007. Cluster-based forwarding for reliable end-to-end delivery in wireless sensor networks. In *Proceedings of IEEE INFOCOM'07*.

A. Cerpa, J. L. Wong, M. Potkonjak, and D. Estrin. 2005. Temporal properties of low power wireless links: Modeling and implications on multi-hop routing. In *Proceedings of ACM MobiHoc'05*.

S. Chakrabarti and A. Mishra. 2001. QoS issues in ad hoc wireless networks. *IEEE Communications Magazine* 39, 2, 142–148.

L. Chen and W. Heinzelman. 2005. QoS-aware routing based on bandwidth estimation for mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications* 23, 3, 561–572.

O. Chipara, Z. He, G. Xing, Q. Chen, X. Wang, C. Lu, J. Stankovic, and T. Abdelzaher. 2006. Real-time power aware routing in wireless sensor networks. In *Proceedings of IWQOS'06*.

D. E. Comer. 2000. *Internetworking with TCP/IP: Principles, Protocols, and Architecture*, Vol. 1. Addison-Wesley.

A. Dunkels, F. Österlind, and Z. He. 2007. An adaptive communication architecture for wireless sensor networks. In *Proceedings of ACM SenSys'07*.

E. Felemban, C. Lee, E. Ekici, R. Boder, and S. Vural. 2005. Probabilistic QoS guarantee in reliability and timeliness domains in wireless sensor networks. In *Proceedings of IEEE INFOCOM'05*.

R. Fonseca, O. Gnawali, K. Jamieson, and P. Levis. 2007. Four bit wireless link estimation. In *Proceedings of HotNets VI'07*.

L. Girod, M. Lukac, V. Trifa, and D. Estrin. 2006. A self-calibrating distributed acoustic sensing platform. In *Proceedings of ACM SenSys'06*.

O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis. 2009. Collection Tree Protocol. In *Proceedings of SenSys'09*.

Y. Gu and T. He. 2007. Data forwarding in extremely low duty-cycle sensor networks with unreliable communication links. In *Proceedings of ACM SenSys'07*.

G. Hackmann, O. Chipara, and C. Lu. 2008. Robust topology control for indoor wireless sensor networks. In *Proceedings of ACM SenSys'08*.

T. He, S. Krishnamurthy, J. A. Stankovic, T. F. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, and B. Krogh. 2004. Energy-efficient surveillance system using wireless sensor networks. In *Proceedings of ACM MobiSys'04*.

T. He, P. Vicaire, T. Yan, L. Luo, L. Gu, G. Zhou, R. Stoleru, Q. Cao, J. A. Stankovic, and T. Abdelzaher. 2006. Achieving real-time target tracking using wireless sensor networks. In *Proceedings of RTAS'06*.

J. L. Hellerstein, Y. Diao, S. Parekh, and D. M. Tilbury. 2004. *Feedback Control of Computing Systems*. John Wiley and Sons.

IEEE 802.15.4. 1999. IEEE 802.15.4, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs).

C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva. 2003. Directed diffusion for wireless sensor networking. *IEEE/ACM Transactions on Networking* 11, 1, 2–16.

S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft. 2006. XORs in the air: Practical wireless network coding. *ACM SIGCOMM Computer Communication Review* 36, 4, 243–254.

M. Kim and B. Noble. 2001. Mobile network estimation. In *Proceedings of ACM MobiCom'01*.

S. Kim, R. Fonseca, P. Dutta, A. Tavakoli, D. Culler, P. Levis, S. Shenker, and I. Stoica. 2007. Flush: A reliable bulk transport protocol for multihop wireless networks. In *Proceedings of ACM SenSys'07*.

S. Kim, S. H. Son, J. A. Stankovic, S. Li, and Y. Choi. 2003. Safe: A data dissemination protocol for periodic updates in sensor networks. In *Proceedings of Distributed Computing Systems Workshops*. IEEE, 228–234.

S. Lin, J. Zhang, G. Zhou, L. Gu, J. A. Stankovic, and T. He. 2006. ATPC: Adaptive transmission power control for wireless sensor networks. In *Proceedings of ACM SenSys'06*.

X. Liu, H. Zhang, Q. Xiang, X. Che, and X. Ju. 2012. Taming uncertainties in real-time routing for wireless networked sensing and control. In *Proceedings of MobiHoc'12*.

C. Lu, J. A. Stankovic, S. H. Son, and G. Tao. 2002. Feedback control real-time scheduling: Framework, modeling, and algorithms. In *Proceedings of JRTS'02*.

C. Lu, X. Wang, and X. Koutsoukos. 2005. Feedback utilization control in distributed real-time systems with end-to-end tasks. *IEEE Transactions on Parallel and Distributed Systems* 16, 6, 550–561.

E. Miluzzo, X. Zheng, K. Fodor, and A. T. Campbell. 2008. Radio characterization of 802.15.4 and its impact on the design of mobile sensor networks. In *Proceedings of WSN'08*.

D. C. Montgomery. 2005. *Introduction to Statistical Quality Control*. John Wiley and Sons.

MultihopLQI 2006. Tinyos MultiHopLQI Collection Protocol. Retrieved January 15, 2015, from http://www.tinyos.net/tinyos-1.x/tos/lib/MultiHopLQI.

J. Paek and R. Govindan. 2007. RCRT: Rate-controlled reliable transport for wireless sensor networks. In *Proceedings of ACM SenSys'07*.

J. Polastre, J. Hill, and D. Culler. 2004. Versatile low power media access for wireless sensor networks. In *Proceedings of ACM SenSys'04*.

J. Polastre, J. Hui, P. Levis, J. Zhao, D. Culler, S. Shenker, and I. Stoica. 2005. A unifying link abstraction for wireless sensor networks. In *Proceedings of ACM Sensys'05*.

D. Puccinelli, O. Gnawali, S. Yoon, S. Santini, U. Colesanti, S. Giordano, and L. Guibas. 2011. The impact of network topology on collection performance. In *Proceedings of EWSN'11*.

P. Santi. 2005. Topology control in wireless ad hoc and sensor networks. *ACM Computing Surveys* 37, 2, 164–194.

H. Schulzrinne and S. Casner. 1993. RTP: A Transport Protocol for Real-Time Applications. Internet draft.

L. Selavo, A. Wood, Q. Cao, T. Sookoor, H. Liu, A. Srinivasan, Y. Wu, W. Kang, J. Stankovic, D. Young, and J. Porter. 2007. LUSTER: Wireless sensor network for environmental research. In *Proceedings of ACM SenSys'07*.

N. Shankaran, N. Roy, D. C. Schmidt, X. D. Koutsoukos, Y. Chen, and C. Lu. 2008. Design and performance evaluation of an adaptive resource management framework for distributed real-time and embedded systems. *EURASIP Journal on Embedded Systems* 2008, Article No. 9.

M. R. Souryal, J. Geissbuehler, L. E. Miller, and N. Moayeri. 2007. Real-time deployment of multihop relays for range extension. In *Proceedings of ACM MobiSys'07*.

K. Srinivasan, M. Kazandjieva, S. Agarwal, and P. Levis. 2008. The beta-factor: Measuring wireless link burstiness. In *Proceedings of ACM SenSys'08*.

M. Wachs, J. I. Choi, J. W. Lee, K. Srinivasan, Z. Chen, M. Jain, and P. Levis. 2007. Visibility: A new metric for protocol design. In *Proceedings of SenSys'07*.

A. Woo, T. Tong, and D. Culler. 2003. Taming the underlying challenges of reliable multihop routing in sensor networks. In *ACM SenSys'03*.

Y. Xue, B. Ramamurthy, and M. Vuran. 2011. SDRCS: A service-differentiated real-time communication scheme for event sensing in wireless sensor networks. *Computer Networks* 55, 5, 3287–3302.

J. Zhao and R. Govindan. 2003. Understanding packet delivery performance in dense wireless sensor networks. In *Proceedings of ACM SenSys'03*.

G. Zhou, Q. Li, J. Li, Y. Wu, S. Lin, J. Lu, C.-Y. Wan, M. D. Yarvis, and J. A. Stankovic. 2011. Adaptive and radio-agnostic QoS for body sensor networks. *ACM Transactions on Embedded Computing Systems* 10, 4, Article No. 48.

Received October 2013; revised July 2014; accepted October 2014