Lessons From A Sensor Network Expedition

Robert Szewczyk[1], Joseph Polastre[1], Alan Mainwaring[2], and David Culler[1,2]

[1] University of California, Berkeley
Berkeley CA 94720
{szewczyk,polastre,culler}@cs.berkeley.edu

[2] Intel Research Laboratory at Berkeley
Berkeley CA 94704
{amm,dculler}@intel-research.net

Abstract. Habitat monitoring is an important driving application for wireless sensor networks (WSNs). Although researchers anticipate some challenges arising in the real-world deployments of sensor networks, a number of problems can be discovered only through experience. This paper evaluates a sensor network system described in an earlier work and presents a set of experiences from a four month long deployment on a remote island off the coast of Maine. We present an in-depth analysis of the environmental and node health data. The close integration of WSNs with their environment provides biological data at densities previous impossible; however, we show that the sensor data is also useful for predicting system operation and network failures. Based on over one million data and health readings, we analyze the node and network design and develop network reliability profiles and failure models.

1 Introduction

Application-driven research has been the foundation of excellent science contributions from the computer science community. This research philosophy is essential for the wireless sensor network (WSN) community. Integrated closely with the physical environment, WSN functionality is affected by many environmental factors not foreseen by developers nor detectable by simulators. WSNs are much more exposed to the environment than the traditional systems. This allows WSNs to densely gather environment data. Researchers can study the relationships between collected environmental data and sensor network behavior. If a particular aspect of sensor functionality is correlated to a set of environmental conditions, network designers can optimize the behavior of the network to exploit a beneficial relationship or mitigate a detrimental one.

Habitat monitoring is widely accepted as a driving application for wireless sensor network research. Many sensor network services are useful for habitat monitoring: localization [1], tracking [3,14,16], data aggregation [9,15,17], and, of course, energy efficient multihop routing [5,13,25]. Ultimately the data collected needs to be meaningful to disciplinary scientists, so sensor design [19] and in-the-field calibration systems are crucial [2,24]. Since such applications need to run unattended, diagnostic and monitoring tools are essential [26].

While these services are an active area of research, few studies have been done using wireless sensor networks in long-term field applications. During the summer of 2002, we deployed an outdoor habitat monitoring application that ran unattended for four months. Outdoor applications present an additional set of challenges not seen in indoor experiments. While we made many simplifying assumptions and engineered out the need for many complex services, we were able to collect a large set of environmental and node diagnostic data. Even though the collected data was not useful for making scientific conclusions, the fidelity of the sensor data yields important observations about sensor network behavior. The data analysis discussed in this paper yields many insights applicable to most wireless sensor deployments. We utilize traditional quality of service metrics such as packet loss; however the sensor data combined with network metrics provide a deeper understanding of failure modes. We anticipate that with system evolution comes higher fidelity sensor readings that will give researchers an even better understanding of sensor network behavior.

The paper is organized as follows: Section 2 provides a detailed overview of the application, Section 3 analyzes the network behaviors that can be deduced from the sensor data, Section 4 contains the analysis of the node-level data. Section 5 contains related work and Section 6 concludes.

2 Application overview

In the summer of 2002, we deployed a 43 node sensor network for habitat monitoring on an uninhabited island 15km off the coast of Maine, USA. Biologists have seasonal field studies with an emphasis on the ecology of the Leach's Storm Petrel [12]. The ability to densely instrument this habitat with sensor networks represents a significant advancement over traditional instrumentation. Monitoring habitats at scale of the organism was previously impossible using standalone data loggers. To assist the biologists, we developed a complete sensor network system, deployed it on the island and monitored its operation for over four months. We used this case study to deepen our understanding of the research and engineering challenges facing the system designers, while providing data that has been previously unavailable to the biologists. The biological background, the key questions of interest, and core system requirements are discussed in depth in [18,20].

In order to study the Leach's Storm Petrel's nesting habits, nodes were deployed in underground nesting burrows and outside burrow entrances above ground. Nodes monitor typical weather data including humidity, pressure, temperature, and ambient light level. Nodes in burrows also monitored infrared radiation to detect the presence of a petrel. The specific components of the WSN and network architecture are described in this section.

2.1 Application Software

Our approach was to simplify the problem wherever possible, to minimize engineering and development efforts, to leverage existing sensor network platforms and components, and to use off-the-shelf products. Our attention was focused on the sensor network operation. We used the Mica mote developed by UC Berkeley [10] running the TinyOS operating system [11].

In order to analyze the long term operation of a WSN, each node executed a simple periodic application that met the biologists requirements in [20]. Every 70 seconds, each node sampled each of its sensors. Data readings were timestamped with 32-bit sequence numbers kept in flash memory. The readings were transmitted in a single 36 byte data packet using the TinyOS radio stack. We relied on the underlying carrier sense MAC layer protocol to prevent against packet collisions. After successfully transmitting the packet of data, the motes entered their lowest power state for the next 70 seconds. The motes were transmit-only devices and the expected duty cycle of the application was 1.7%. The motes were powered by two AA batteries with an estimated 2200mAh capacity.

2.2 Sensor board design

We designed and built a microclimate sensor board for monitoring the Leach's Storm Petrel. We decided to use surface mount components because they are smaller and operate at lower voltages than probe-based sensors. Burrow tunnels are small, about the size of a fist. A mote was required to fit into the tunnel and not obstruct the passage. Above ground, size constraints were relaxed. To fit the node into a burrow, the sensor board integrated all sensors into a single package to minimize size and complexity. To monitor the petrel's habitat, the sensor board included a photoresistive light sensor, digital temperature sensor, capacitive humidity sensor, digital barometric pressure sensor, and passive infrared detector (thermopile with thermistor). One consequence of an integrated sensor board is the amount of shared fate between sensors; a failure of one sensor likely affects all other sensors. The design did not consider fault isolation among independent sensors or controlling the effects of malfunctioning sensors on shared hardware resources.

2.3 Packaging strategy

In-situ instrumentation experiences diverse weather conditions including dense fog with pH readings of less than 3, dew, rain, and flooding. Waterproofing the mote and its sensors is essential for prolonged operation.

Sealing electronics from the environment could be done with conformal coating, packaging, or combinations of the two. Since our sensors were surface mounted and needed to be exposed to the environment, we sealed the entire mote with parylene leaving the sensor elements exposed. We tested the sealant with a coated mote that ran submerged in a coffee cup of water for days.

Our survey of off-the-shelf enclosures found many that were slightly too small for the mote or too large for tunnels. Custom enclosures were too costly. Above ground motes were placed in ventilated acrylic enclosures. In burrows, sealed motes were deployed without enclosures.

(a) Sealed block (b) Cylinder with vents and drainage

Fig. 1. Acrylic enclosures used at different outdoor applications.

Of primary concern for the packaging was the effect it has on RF propagation. We decided to use board-mounted miniature whip antennas and build enclosures out of acrylic. There were significant questions about RF propagation from motes inside burrows, above ground on the surface, within inches of granite rocks, tree roots and low, dense vegetation. When we deployed the motes we noted the ground substrate, distance into the burrow, and geographic location of each mote to assist in the analysis of the RF propagation for each mote. Enclosures built for this deployment can be seen in Fig. 1.

2.4 Application Realization

The network architecture had a multi-level structure as shown in Fig. 2. The first level consisted of motes with sensors. In general, sensor nodes perform general-purpose computing and networking, as well as application-specific tasks. The *gateway* is responsible for transmitting packets from the *sensor patch* through a local transit network to one or more *base stations*. The base stations in the third level provide database services as well as Internet connectivity. The fourth and final level consists of remote servers supporting analysis, visualization and web content. Mobile devices may interact with any of the networks–whether it is used in the field or across the world connected to a database replica.

The sensor patch consisted of the motes with their sensor boards in a single hop network. The single hop network was chosen not only for simplicity but also to evaluate the characteristics of a single radio cell without interfering cells. The gateway was implemented with a relay mote connected to a high-gain Yagi antenna to retransmit data from the sensor patch over a 350 foot link to the base station. The relay node ran at a 100% duty cycle powered by a solar cell and rechargeable battery. The data was logged by a laptop into a Postgres database and then replicated through an Internet satellite connection.

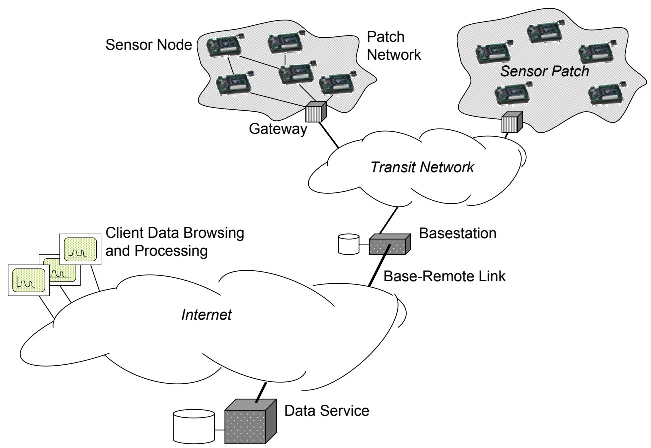

Fig. 2. System architecture for habitat monitoring

2.5 Experimental Goals

Since our deployment was the first long term use of the mote platform, we were interested in how the system would perform. Specifically, this deployment served to prove the feasibility of using a miniature low-power wireless sensor network for long term deployments. We set out to evaluate the efficacy of the sealant, the radio performance in and out of burrows, the usefulness of the data for biologists including the occupancy detector, and the system and network longevity. Since each hardware and software component was relatively simple, our goal was to draw significant conclusions about the behavior of wireless sensor networks from the resulting data.

After 123 days of the experiment, we logged over 1.1 million readings. During this period, we noticed abnormal operation among the node population. Some nodes produced sensor readings out of their operating range, others had erratic packet delivery, and some failed. We sought to understand why these events had occurred. By evaluating these abnormalities, future applications may be designed to isolate problems and provide notifications or perform self-healing. The next two sections analyze node operation and identify the causes of abnormal behavior.

3 Network analysis

We need evaluate the behavior of the sensor network to establish convincing evidence that the system is operating correctly. Our application was implemented as a single hop network, however the behavior in a single hop is equivalent to what occurs in any WSN radio cell. We begin by examining WSN operation and its performance over time in order to evaluate network cell characteristics.

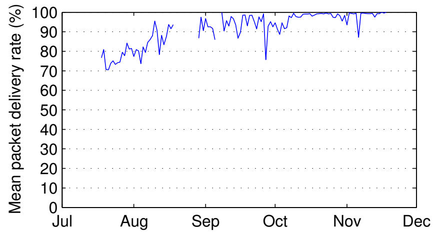

Fig. 3. Average daily losses in the network throughout the deployment. The gap in the second part of August corresponds to a database crash.

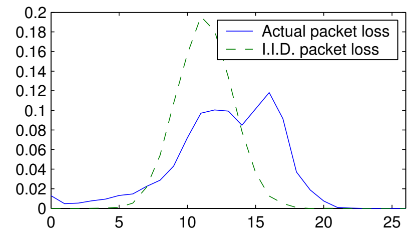

Fig. 4. Distribution of packet losses in a time slot. Statistically, the losses are not independently distributed.

3.1 Packet loss

A primary metric of network performance is packet loss in the network over time. Packet loss is a quality-of-service metric that indicates the effective end-to-end application throughput [4]. The average daily packet loss is shown in Fig. 3. Two features of the packet loss plot demand explanation: (1) why was the initial loss rate high and (2) why does the network improve with time? Note that the size of the sensor network is declining over time due to node failures. Either motes with poor packet reception die quicker or the radio channel experiences less contention and packet collisions as the number of nodes decreases. To identify the cause, we examine whether a packet loss at a particular node is dependent on losses from other nodes.

The periodic nature of the application allows us to assign virtual time slots to each data packet corresponding with a particular sequence number from each node. After splitting the data into time slices, we can analyze patterns of loss within each time slot. Fig. 5 shows packet loss patterns within the network during the first week of August 2002. A black line in a slot indicates that a packet expected to arrive was lost, a white line means a packet was successfully received. If all packet loss was distributed independently, the graph would contain a random placement of black and white bars appearing as a gray square. We note that 23 nodes do not start to transmit until the morning of August 6; that reflects the additional mote deployment that day. Visual inspection reveals patterns of loss: several black horizontal lines emerge, spanning almost all nodes, *e.g.* midday on August 6, 7, and 8. Looking closer at the packet loss on August 7, we note it is the only time in the sample window when motes 45 and 49 transmit packets successfully; however, heavy packet loss occurs at most other nodes. Sequence numbers received from these sensors reveal they transmitted data during every sample period since they were deployed even though those packets were not received.

More systematically, Fig. 4 compares the empirical distribution of packet loss in a slot to an independent distribution. The hypothesis that the two distribu-

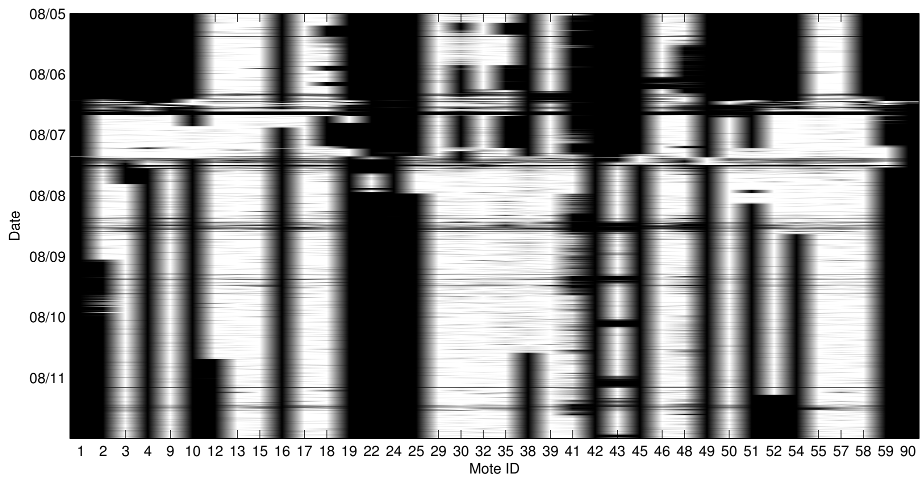

Fig. 5. Packet loss patterns within the deployed network during a week in August. Y-axis represents time divided into virtual packet slots (note: time increases downwards). A black line in the slot indicates that a packet expected to arrive in this time slot was missed, a white line means that a packet was successfully received.

tions are the same is rejected by both parametric (χ^2 test yields 10^8) and non-parametric techniques (rank test rejects it with 99% confidence). The empirical distribution appears a superposition of two Gaussians: this is not particularly surprising, since we record packet loss at the end of the path (recall network architecture, Section 2). This loss is a combination of potential losses along two hops in the network. Additionally, packets share the channel that varies with the environmental conditions, and sensor nodes are likely to have similar battery levels. Finally, there is a possibility of packet collisions at the relay nodes.

3.2 Network dynamics

Given that the expected network utilization is very low (less than 5%) we would not expect collisions to play a significant role. Conversely, the behavior of motes 45 and 49 implies otherwise: their packets are only received when most packets from other nodes are lost. Such behavior is possible in a periodic application: in the absence of any backoff, the nodes will collide repeatedly. In our application, the backoff was provided by the CSMA MAC layer. If the MAC worked as expected, each node would backoff until it found a clear slot; at that point, we would expect the channel to be clear. Clock skew and channel variations might force a slot reallocation, but such behavior should be infrequent.

Looking at the timestamps of the received packets, we can compute the phase of each node, relative to the 70 second sampling period. Fig. 6 plots the phase

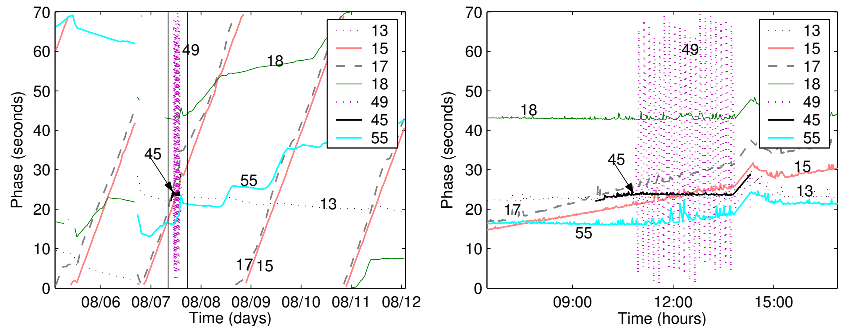

Fig. 6. Packet phase as a function of time; the right figure shows the detail of the region between the lines in the left figure.

of selected nodes from Fig. 5. The slope of the phase corresponds to a drift as a percentage of the 70-second cycle. In the absence of clock drift and MAC delays, each mote would always occupy the same time slot cycle and would appear as a horizontal line in the graph. A 5 ppm oscillator drift would result in gently sloped lines, advancing or retreating by 1 second every 2.3 days. In this representation, the potential for collisions exists only at the intersections of the lines.

Several nodes display the expected characteristics: motes 13, 18, and 55 hold their phase fairly constant for different periods, ranging from a few hours to a few days. Other nodes, *e.g.* 15 and 17 appear to delay the phase, losing 70 seconds every 2 days. The delay can come only from the MAC layer; on average they lose 28 msec, which corresponds to a single packet MAC backoff. We hypothesize that this is a result of the RF automatic gain control circuits: in the RF silence of the island, the node may adjust the gain such that it detects radio noise and interprets it as a packet. Correcting this problem may be done by incorporating a signal strength meter into the MAC that uses a combination of digital radio output and analog signal strength. This additional backoff seems to capture otherwise stable nodes: *e.g.* mote 55 on August 9 transmits in a fixed phase until it comes close to the phase of 15 and 17. At that point, mote 55 starts backing off before every transmission. This may be caused by implicit synchronization between nodes caused by the transit network.

We note that potential for collisions does exist: the phases of different nodes do cross on several occasions. When the phases collide, the nodes back off as expected, *e.g.* mote 55 on August 9 backs off to allow 17 to transmit. Next we turn to motes 45 and 49 from Fig. 5. Mote 45 can collide with motes 13 and 15; collisions with other nodes, on the other hand, seem impossible. In contrast, mote 49, does not display any potential for collisions; instead it shows a very rapid phase change. Such behavior can be explained either though a clock drift, or through the misinterpretation of the carrier sense (*e.g.* a mote determines it needs to wait a few seconds to acquire a channel). We associate such behavior with faulty nodes, and return to it in Section 4.

4 Node-level analysis

Nodes in outdoor WSNs are exposed to closely monitor and sense their environment. Their performance and reliability depend on a number of environmental factors. Fortunately, the nodes have a local knowledge of these factors, and they may exploit that knowledge to adjust their operation. Appropriate notifications from the system would allow the end user to pro-actively fix the WSN. Ideally, the network could request proactive maintenance, or self-heal. We examine the link between sensor and node performance. Although the particular analysis is specific to this deployment, we believe that other systems will be benefit from similar analyses: identifying outlier readings or loss of expected sensing patterns, across time, space or sensing modality. Additionally, since battery state is an important part of a node's self-monitoring capability [26], we also examine battery voltage readings to analyze the performance of our power management implementation.

4.1 Sensor analysis

The suite of sensors on each node provided analog light, humidity, digital temperature, pressure, and passive infrared readings. The sensor board used a separate 12-bit ADC to maximize the resolution and minimize analog noise. We examine the readings from each sensor.

Light readings: The light sensor used for this application was a photoresistor that we had significant experience with in the past. It served as a confidence building tool and ADC test. In an outdoor setting during the day, the light value saturated at the maximum ADC value, and at night the values were zero. Knowing the saturation characteristics, not much work was invested in characterizing its response to known intensities of light. The simplicity of this sensor combined with an *a priori* knowledge of the expected response provided a valuable baseline for establishing the proper functioning of the sensor board. As expected, the sensors deployed above ground showed periodic patterns of day and night and burrows showed near to total darkness. Fig. 7 shows light and temperature readings and average light and temperature readings during the experiment.

The light sensor operated most reliably of the sensors. The only behavior identifiable as failure was disappearance of diurnal patterns replaced by high value readings. Such behavior is observed in 7 nodes out of 43, and in 6 cases it is accompanied by anomalous readings from other sensors, such as a 0^oC temperature or analog humidity values of zero.

Temperature readings: A Maxim 6633 digital temperature sensor provided the temperature measurements While the sensor's resolution is 0.0625^oC, in our deployment it only provided a 2^oC resolution: the hardware always supplied readings with the low-order bits zeroed out. The enclosure was IR transparent to assist the thermopile sensor; consequently, the IR radiation from direct sunlight would enter the enclosure and heat up the mote. As a result, temperatures measured inside the enclosures were significantly higher than the ambient

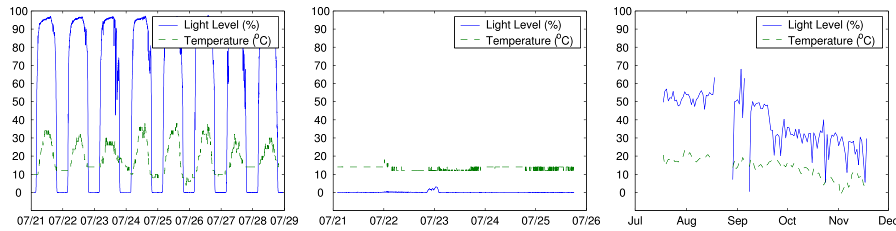

Fig. 7. Light and temperature time series from the network. From left: outside, inside, and daily average outside burrows.

temperatures measured by traditional weather stations. On cloudy days the temperature readings corresponded closely with the data from nearby weather buoys operated by NOAA.

Even though motes were coated with parylene, sensor elements were left exposed to the environment to preserve their sensing ability. In the case of the temperature sensor, a layer of parylene was permissible. Nevertheless the sensor failed when it came in direct contact with water. The failure manifested itself in a persistent reading of 0^{o}C. Of 43 nodes, 22 recorded a faulty temperature reading and 14 of those recorded their first bad reading during storms on August 6. The failure of temperature sensor is highly correlated with the failure of the humidity sensor: of 22 failure events, in two cases the humidity sensor failed first and in two cases the temperature sensor failed first. In remaining 18 cases, the two sensors failed simultaneously. In all but two cases, the sensor did not recover.

Humidity readings: The relative humidity sensor was a capacitive sensor: its capacitance was proportional to the humidity. In the packaging process, the sensor needed to be exposed; it was masked out during the parylene sealing process, and we relied on the enclosure to provide adequate air circulation while keeping the sensor dry. Our measurements have shown up to 15% error in the interchangeability of this sensor across sensor boards. Tests in a controlled environment have shown the sensor produces readings with 5% variation due to analog noise. Prior to deployment, we did not perform individual calibration; instead we applied the reference conversion function to convert the readings into SI units.

In the field, the protection afforded by our enclosure proved to be inadequate. When wet, the sensor would create a low-resistance path between the power supply terminals. Such behavior would manifest itself in either abnormally large (more than 150%) or very small humidity readings (raw readings of 0V). Fig. 8 shows the humidity and voltage readings as well as the packet reception rates of selected nodes during both rainy and dry days in early August. Nodes 17 and 29 experienced a large drop in voltage while recording an abnormally high humidity readings on Aug. 5 and 6. We attribute the voltage drop to excessive load on the batteries caused by the wet sensor. Node 18 shows an more severe effect of rain: on Aug. 5, it crashes just as the other sensors register a rise in the humidity

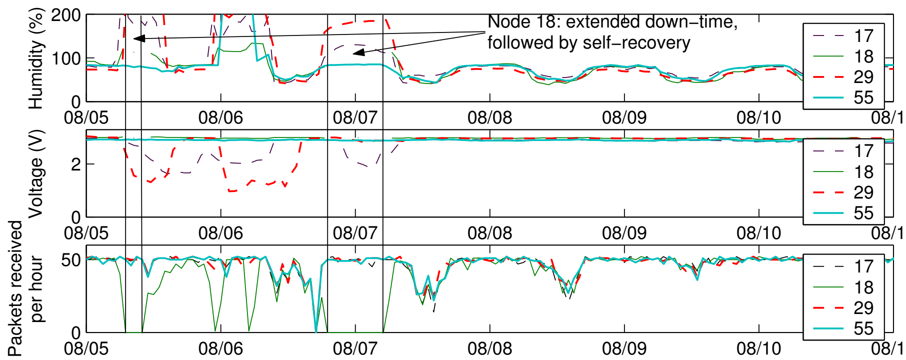

Fig. 8. Sensor behavior during the rain. Nodes 17 and 29 experience substantial drop in voltage, while node 55 crashes. When the humidity sensor recovers, the nodes recover.

readings. Node 18, on the other hand, seems to be well protected: it registers high humidity readings on Aug. 6, and its voltage and packet delivery rates are not correlated with the humidity readings. Nodes that experienced the high humidity readings typically recover when they dried up; nodes with the unusually low readings would fail quickly. While we do not have a definite explanation for such behavior, we evaluate that characteristics of the sensor board as a failure indicator below.

Thermopile readings: The data from the thermopile sensor proved difficult to analyze. The sensor measures two quantities: the ambient temperature and the infrared radiation incident on the element. The sum of thermopile and thermistor readings yields the object surface temperature, *e.g.* a bird. We would expect that the temperature readings from the thermistor and from the infrared temperature sensor would closely track each other most of the time. By analyzing spikes in the IR readings, we should be able to deduce the bird activity.

The readings from the thermistor do, in fact, track closely with the temperature readings. Fig. 9 compares the analog thermistor with the digital maxim temperature sensor. The readings are closely correlated although different on an absolute scale. A best linear fit of the temperature data to the thermistor readings on a per sensor per day basis yields a mean error of less than 0.9^oC, within the half step resolution of the digital sensor. The best fit coefficient varies substantially across the nodes.

Assigning biological significance to the infrared data is a difficult task. The absolute readings often do not fall in the expected range. The data exhibits a lack of any periodic daily patterns (assuming that burrow occupancy would exhibit them), and the sensor output appears to settle quickly in one of the two extreme readings. In the absence of any ground truth information, *e.g.* infrared camera images corresponding to the changes in the IR reading, the data is inconclusive.

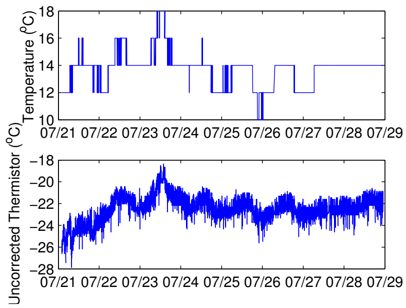

Fig. 9. The temperature sensor (top) and thermistor (bottom), though very different on the absolute scale, are closely correlated: a linear fit yields a mean error of less than 0.8^{o}C.

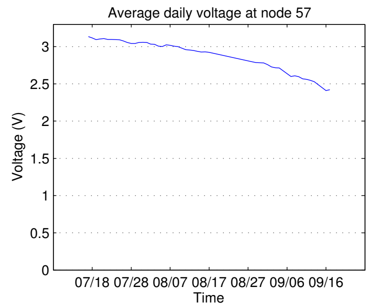

Fig. 10. Voltage readings from node 57. Node 57 operates until the voltage falls below 2.3V; at this point the alkaline cells can not supply enough current to the boost converter.

4.2 Power Management

As mentioned in Section 2, one of the main challenges was sensor node power management. We evaluate the power management in the context of the few nodes that did not exhibit other failures. Motes do not have a direct way of measuring the energy they consumed, instead we use battery voltage as an indirect measure. The analysis of the aggregate population is somewhat complicated by in-the-field battery replacements, failed voltage indicators, failed sensors and gaps in the data caused by the database crashes. Only 5 nodes out of 43 have clearly exhausted their original battery supply. This limited sample makes it difficult to perform a thorough statistical analysis. Instead we examine the battery voltage of a single node without other failures. Fig. 10 shows the battery voltage of a node as a function of time. The battery is unable to supply enough current to power the node once the voltage drops below 2.30V. The boost converter on the Mica mote is able to extract only 15% more energy from the battery once the voltage drops below 2.5V (the lowest operating voltage for the platform without the voltage regulation). This fell far short of our expectations of being able to drain the batteries down to 1.6V, which represents an extra 40% of energy stored in a cell [6]. The periodic, constant power load presented to the batteries is ill suited to extract the maximum capacity. For this class of devices, a better solution would use batteries with stable voltage, *e.g.* some of the lithium-based chemistries. We advocate future platforms eliminate the use of a boost converter.

4.3 Node failure indicators

In the course of data analysis we have identified a number of anomalous behaviors: erroneous sensor readings and application phase skew. The humidity sensor seemed to be a good indicator of node health. It exhibited 2 kinds of

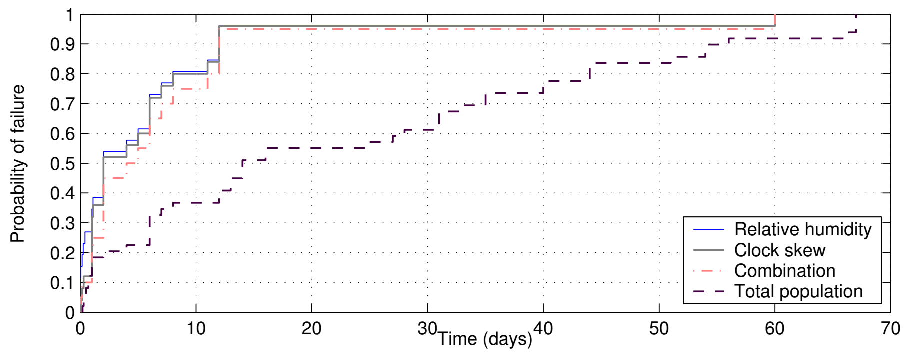

Fig. 11. Cumulative probability of node failure in the presence of clock skew and anomalous humidity readings compared with the entire population of nodes.

erroneous behaviors: very high and very low readings. The high humidity spikes, even though they drained the mote's batteries, correlated with recoverable mote crashes. The humidity readings corresponding to a raw voltage of 0V correlated with permanent mote outage: 55% of the nodes with excessively low humidity readings failed within two days. In the course of packet phase analysis we noted some motes with slower than usual clocks. This behavior also correlates well with the node failure: 52% of nodes with such behavior fail within two days.

These behaviors have a very low false positive detection rate: only a single node exhibiting the low humidity and two nodes exhibiting clock skew (out of 43) exhausted their battery supply instead of failing prematurely. Fig. 11 compares the longevity of motes that have exhibited either the clock skew or a faulty humidity sensor against the survival curve of mote population as a whole. We note that 50% of motes with these behaviors become inoperable within 4 days.

5 Related work

Traditional data loggers for habitat monitoring are typically large in size and expensive. They require that intrusive probes and corresponding equipment immediately adjacent. They are typically used since they are commercially available, supported, and provide a variety of sensors. One such data logger is the Hobo Data Logger [19]. Due to size, price, and organism disturbance, using these systems for fine-grained habitat monitoring is inappropriate.

Other habitat monitoring studies install one or a few sophisticated weather stations an "insignificant distance" from the area of interest. With this method, biologists cannot gauge whether the weather station actually monitors a different microclimate due to its distance from the organism being studied. Using the readings from the weather station, biologists make generalizations through coarse measurements and sparsely deployed weather stations. Instead, we strive to provide biologists the ability to monitor the environment on the scale of the organism, not on the scale of the biologist [8,21].

Habitat monitoring for WSNs has been studied by a variety of other research groups. Cerpa et. al. [3] propose a multi-tiered architecture for habitat monitoring. The architecture focuses primarily on wildlife tracking instead of habitat monitoring. A PC104 hardware platform was used for the implementation with future work involving porting the software to motes. Experimentation using a hybrid PC104 and mote network has been done to analyze acoustic signals [23], but no long term results or reliability data has been published. Wang et. al. [22] implement a method to acoustically identify animals using a hybrid iPaq and mote network.

ZebraNet [14] is a WSN for monitoring and tracking wildlife. ZebraNet nodes are significantly larger and heavier than motes. The architecture is designed for an always mobile, multi-hop wireless network. In many respects, this design does not fit with monitoring the Leach's Storm Petrel at static positions (burrows). ZebraNet, at the time of this writing, has not yet had a full long-term deployment.

The number of deployed wireless sensor network systems is extremely low. The Center for Embedded Network Sensing (CENS) has deployed their Extensible Sensing System [7] at the James Mountain Reserve in California. Their architecture is similar to ours with a variety of sensor patches connected via a transit network that is tiered. Intel Research has recently deployed a network to monitor Redwood canopies in Northern California and a second network to monitor vineyards in Oregon. We deployed a second generation multihop habitat monitoring network on Great Duck Island, Maine in the summer of 2003. These networks are in their infancy but analysis may yield the benefits of various approaches to deploying habitat monitoring systems.

6 Conclusion

We have analyzed the environmental data from one of the first outdoor deployments of WSNs. While the deployment exhibited very high node failure rates, it yielded valuable insight into WSN operation that could not have been obtained in simulation or in an indoor deployment. We have identified sensor features that predict a 50% node failure within 4 days. We analyzed the application-level data to show complex behaviors in low levels of the system, such as MAC-layer synchronization of nodes. We have shown that great care must be taken when deploying WSNs such that the MAC implementation, network topology, synchronization, sensing modalities, sensor design, and packaging be designed with awareness of each other.

Sensor networks do not exist in isolation from their environment; they are embedded within it and greatly affected by it. This work shows that the anomalies in sensor readings can predict node failures with high confidence. Prediction enables pro-active maintenance and node self-maintenance. This insight will be very important in the development of self-organizing and self-healing WSNs.

Notes

Data from the wireless sensor network deployment on Great Duck Island can be view graphically at `http://www.greatduckisland.net`. Our website also includes the raw data for researchers in both computer science and the biological sciences to download and analyze.

This work was supported by the Intel Research Laboratory at Berkeley, DARPA grant F33615-01-C1895 (Network Embedded Systems Technology), the National Science Foundation, and the Center for Information Technology Research in the Interest of Society (CITRIS).

References

1. Bulusu, N., Bychkovskiy, V., Estrin, D., and Heidemann, J. Scalable, ad hoc deployable, RF-based localization. In *Proceedings of the Grace Hopper Conference on Celebration of Women in Computing* (Vancouver, Canada, Oct. 2002).
2. Bychkovskiy, V., Megerian, S., Estrin, D., and Potkonjak, M. Colibration: A collaborative approach to in-place sensor calibration. In *Proceedings of the 2nd International Workshop on Information Processing in Sensor Networks (IPSN'03)* (Palo Alto, CA, USA, Apr. 2003).
3. Cerpa, A., Elson, J., Estrin, D., Girod, L., Hamilton, M., and Zhao, J. Habitat monitoring: Application driver for wireless communications technology. In *2001 ACM SIGCOMM Workshop on Data Communications in Latin America and the Caribbean* (San Jose, Costa Rica, Apr. 2001).
4. Chalmers, D., and Sloman, M. A survey of Quality of Service in mobile computing environments. *IEEE Communications Surveys 2*, 2 (1992).
5. Chen, B., Jamieson, K., Balakrishnan, H., and Morris, R. Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. In *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking* (Rome, Italy, July 2001), ACM Press, pp. 85–96.
6. Eveready Battery Company. Energizer no. x91 datasheet. `http://data.energizer.com/datasheets/library/primary/alkaline/energizer_e2/x91.pdf`.
7. Hamilton, M., Allen, M., Estrin, D., Rottenberry, J., Rundel, P., Srivastava, M., and Soatto, S. Extensible sensing system: An advanced network design for microclimate sensing. `http://www.cens.ucla.edu`, June 2003.
8. Happold, D. C. The subalpine climate at smiggin holes, Kosciusko National Park, Australia, and its influence on the biology of small mammals. *Arctic & Alpine Research 30* (1998), 241–251.
9. He, T., Blum, B., Stankovic, J., and Abdelzaher, T. AIDA: Adaptive Application Independant Data Aggregation in Wireless Sensor Networks. *ACM Transactions in Embedded Computing Systems (TECS), Special issue on Dynamically Adaptable Embedded Systems* (2003).
10. Hill, J., and Culler, D. Mica: a wireless platform for deeply embedded networks. *IEEE Micro 22*, 6 (November/December 2002), 12–24.
11. Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D., and Pister, K. System architecture directions for networked sensors. In *Proceedings of the 9th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-IX)* (Cambridge, MA, USA, Nov. 2000), ACM Press, pp. 93–104.

12. Huntington, C., Butler, R., and Mauck, R. *Leach's Storm Petrel (Oceanodroma leucorhoa)*, vol. 233 of *Birds of North America*. The Academy of Natural Sciences, Philadelphia and the American Orinthologist's Union, Washington D.C., 1996.
13. Intanagonwiwat, C., Govindan, R., and Estrin, D. Directed diffusion: a scalable and robust communication paradigm for sensor networks. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking* (Boston, MA, USA, Aug. 2000), ACM Press, pp. 56–67.
14. Juang, P., Oki, H., Wang, Y., Martonosi, M., Peh, L.-S., and Rubenstein, D. Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet. In *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-X)* (San Jose, CA, USA, Oct. 2002), ACM Press, pp. 96–107.
15. Krishanamachari, B., Estrin, D., and Wicker, S. The impact of data aggregation in wireless sensor networks. In *Proceedings of International Workshop of Distributed Event Based Systems (DEBS)* (Vienna, Austria, July 2002).
16. Liu, J., Cheung, P., Guibas, L., and Zhao, F. A dual-space approach to tracking and sensor management in wireless sensor networks. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications* (Atlanta, GA, USA, Sept. 2002), ACM Press, pp. 131–139.
17. Madden, S., Franklin, M., Hellerstein, J., and Hong, W. TAG: a Tiny AGgregation service for ad-hoc sensor networks. In *Proceedings of the 5th USENIX Symposium on Operating Systems Design and Implementation (OSDI '02)* (Boston, MA, USA, Dec. 2002).
18. Mainwaring, A., Polastre, J., Szewczyk, R., Culler, D., and Anderson, J. Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications* (Atlanta, GA, USA, Sept. 2002), ACM Press, pp. 88–97.
19. Onset Computer Corporation. HOBO weather station. `http://www.onsetcomp.com`.
20. Polastre, J. Design and implementation of wireless sensor networks for habitat monitoring. Master's thesis, University of California at Berkeley, 2003.
21. Toapanta, M., Funderburk, J., and Chellemi, D. Development of Frankliniella species (Thysanoptera: Thripidae) in relation to microclimatic temperatures in vetch. *Journal of Entomological Science 36* (2001), 426–437.
22. Wang, H., Elson, J., Girod, L., Estrin, D., and Yao, K. Target classification and localization in habitat monitoring. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2003)* (Hong Kong, China, Apr. 2003).
23. Wang, H., Estrin, D., and Girod, L. Preprocessing in a tiered sensor network for habitat monitoring. *EURASIP JASP Special Issue on Sensor Networks 2003*, 4 (Mar. 2003), 392–401.
24. Whitehouse, K., and Culler, D. Calibration as parameter estimation in sensor networks. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications* (Atlanta, GA, USA, Sept. 2002), ACM Press.
25. Xu, Y., Heidemann, J., and Estrin, D. Geography-informed energy conservation for ad hoc routing. In *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking* (Rome, Italy, July 2001), ACM Press.
26. Zhao, J., Govindan, R., and Estrin, D. Computing aggregates for monitoring wireless sensor networks. In *Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications* (Anchorage, AK, USA, May 2003).