

ROPE: A Reactive, Opportunistic Protocol for Environment Monitoring Sensor Networks

Rachel Cardell-Oliver
School of Computer Science & Software Engineering
The University of Western Australia
rachel@csse.uwa.edu.au

Abstract

The goal of sensor networks that monitor the environment is to detect and report the temporal and spatial dynamics of that environment and to run unattended for several months. Meeting all three of these requirements in real applications has proved to be a challenging task. This paper presents a novel protocol for data gathering that adapts opportunistically to changes in its environment. Each node compresses its gathered data locally, transmitting a burst of data when communication conditions are good. We report the design and analysis of the protocol and results of preliminary implementation tests using CSIRO Flecks and TinyOS.

1. Introduction

The goal of sensor networks that monitor the environment is to *detect* and *report* the temporal and spatial dynamics of that environment and to run *unattended* for several months. Meeting all three of these requirements in real applications has proved to be a challenging task [2, 11, 13].

Our observations of the behaviour of outdoor sensor networks have identified several important properties about the variability of their performance that have not been considered in previous protocol designs. The capacity of a sensor network to deliver sensed data can vary significantly over time as a result of communication failures. Fortunately, most environment monitoring applications can tolerate delays of hours or even days to deliver gathered data. Furthermore, the demand from environment sensor nodes to report data is usually a very small percentage of the network's normal capacity.

Existing data gathering protocols offer regular transmission slots for each node's data. They assume that communication failures can be overcome by the repeated transmission of individual data items. They also assume that all processing of gathered data takes place centrally, after

transmission, rather than locally on the gathering nodes. However, this approach does not support long-lived networks, because nodes waste valuable energy on data re-transmissions when communication conditions are poor or on transmitting redundant data. We propose instead, that the best approach for data gathering in changeable communication environments, is for nodes to transmit their data in bursts when conditions are favourable. This paper proposes a novel data gathering protocol, ROPE, that takes account of the varying capacity and demand of environment monitoring applications. ROPE consequently maximises the in-field lifetime of a network.

Many protocols have been proposed for data gathering in sensor networks. Most maintain a routing tree from sensor nodes to one or more base stations. Sensor nodes either report their data immediately through the tree [11, 2], or data may be filtered and stored at each node, and then transmitted later [5]. However, the overhead for tree maintenance is high given that the volume of data to be delivered is low. Thus, maintaining a routing tree conflicts with the goal of maximising the lifetime of the sensor network.

There is a growing body of experimental evidence that "the neighbour abstraction [of communication connectivity between nodes] is a poor approximation of reality" [1] in outdoor wireless networks [11, 4, 2, 14]. The standard mechanisms for overcoming communication failures between nodes include handshaking [15], redundant transmissions [11] and redundant data paths [5]. Handshaking with data acknowledgements, together with repeated transmissions (by default, up to 7), was used with the SMAC protocol in a soil-moisture monitoring network [15, 2]. In a habitat monitoring network [11], every data packet was transmitted multiple times (by default, 3), but without acknowledgement. Neither of these schemes addresses the problem of long periods of poor communication, resulting in low data delivery rates and wasted energy; currently deployed sensor networks report data delivery rates around 60% [11, 13, 2].

In the SECOAS firefly MAC protocol, nodes adapt

to communication failures by ceasing their time-division scheduled transmissions when they cannot hear their neighbours and resuming when communication conditions improve [5]. The ROPE protocol also adapts to communication failures, but using a different mechanism. In ROPE a node transmits a burst of data packets only if it receives feedback that the data path is reliable. In both cases, nodes do not waste energy transmitting packets when communication links are very poor.

One of the most energy-expensive operations in a sensor network is transmitting a packet. Thus, the lifetime of a network can be improved significantly if each node reduces the number of packets that need be transmitted [8, 9]. Local data compression is particularly suitable for environment monitoring applications with generous latency requirements, because the more data a node gathers before it is required to report that data, the greater the chance for significant local compression of the data.

In biological protocols used in the SECOAS network [8], nodes compress their data to maximise each individual node's fitness. For example, if a node has low batteries, a nearly full logging space, or hears its near neighbours transmitting similar data, then it reduces its frequency of data gathering, and thus the amount of data it must later transmit. Conversely, if interesting changes are occurring in its local environment, then data gathering frequency is increased. In trials with a sea-bed sensor network, such a protocol achieved 50% compression of gathered data [8].

In lightweight temporal compression [9], each node saves linear fragments of its gathered data, to approximate the full data sequence, up to sensor error bounds. Lightweight compression is simple to implement and able to run on small processors such as used in our applications [7, 10]. It has also been shown to deliver significant 20-to-1 compression.

This paper proposes a novel protocol, ROPE, for data-gathering by sensor networks that monitor outdoor environments. We report the results of preliminary implementation trials using CSIRO Fleck hardware [10] and Mica2 motes [7] with TinyOS software components [12]. ROPE is shown to meet the requirement of reporting spatial and temporal dynamics of environment variables, whilst respecting the constraints of the environment in which data gathering occurs. Its novel combination of features are:

- **Opportunistic transmission times** are chosen to transmit bursts of data when conditions are good, and otherwise wait, taking advantage of the generous latency requirements of environment monitoring applications;
- **Lightweight in-network data processing** is used to minimise the amount of data transmitted, saving network energy, and aiding scientific analysis of the delivered data;

- **Minimal state information** is stored by each node, making the network self-organising and fault-tolerant. For example, path information is only generated when needed, and data packets are not acknowledged individually.

The paper is organised as follows. In Section 2 we identify requirements for a general class of environment monitoring applications, and also the constraints imposed by their outdoor sensor network environments. Section 3 presents the ROPE protocol, showing how it addresses these requirements and constraints. Section 4 reports the results of preliminary implementation tests and Section 5 our conclusions and ongoing work.

2. Requirements and Constraints

The main functional requirement for environment monitoring is to capture as accurately as possible the temporal and spatial dynamics of environment variables. A sensor network achieves this goal using clusters of data gathering nodes. Each node monitors local environment variables, reporting significant changes to one or more gateway nodes or base stations. Base stations transfer the data to permanent storage.

2.1. Resources Required for Reporting Data

The demand from sensor nodes for reporting environment data is characterised by the following parameters.

- **N (nodes):** the number of sensing nodes in the network.
- **S (sensing interval):** the temporal separation between readings of the monitored environment variable.
- **D (data compression):** the effect of filtering on the size of a measured data series. Only significant changes in the monitored variable need be saved.
- **RP (readings per packet):** the number of readings (minimum, maximum or average) in a data transmission packet.
- **L (latency):** the maximum time that can elapse before a filtered series of readings need be reported.

The values of these parameters may change over time as a result of changes in the environment (such as increased temporal dynamics of monitored variables), or changes made by the protocol (such as altering the sensing interval or latency time).

2.2. Capacity Available for Reporting Data

In order to save energy, and thus maximise the lifetime of a sensor network, most network protocols synchronise their nodes in sleeping and waking cycles. The capacity of a sensor network for a given application is characterised by the following parameters. Again, the values of these parameters may change over time because of changes in the environment or they may be adjusted by the protocol.

- **W (waking cycle time):** the temporal separation between waking periods for each node.
- **C (communication slot availability):** the proportion of waking periods in which communication is feasible.
- **R (retransmission ratio):** the average number of times each packet is transmitted in a good communication period, including retransmissions to overcome losses.
- **PW (maximum packets per cycle):** the maximum number of packets that can be transmitted in a single waking period.
- **NW (nodes per cycle):** the number of nodes allowed to transmit per waking period.

The conditions under which a specific sensor network application and environment is able to deliver its data successfully can now be expressed in terms of the demand for reporting, and the network capacity available for reporting. Capacity and demand with subscript s is measured in waking periods per latency period, while subscript p is packets per waking period.

- $\text{Capacity}_s = C \cdot L / W$
- $\text{Demand}_s = N / NW$
- $\text{Capacity}_p = PW$
- $\text{Demand}_p = NW \cdot R \cdot \lceil (D \cdot (L/S) / RP) \rceil$

The requirement for successful environment monitoring is determined by a simple relationship: capacity must be greater than or equal to demand for both s and p measures.

2.3. Maximising Field Life

The highest priority non-functional requirement for environment monitoring sensor networks is to maximise the time the network is effectively gathering and reporting environment data. In order to meet this goal, trade-offs can be made against reliability and latency. Delivery of most (for example 90%), but not every, sensor reading is a sufficient requirement for most applications. Spatial redundancy of sensing nodes and local back-ups, are used to compensate for potential network data loss. A delay of many hours

in reporting readings is also acceptable, since the main reasons for reporting are: to free local node storage, to monitor node health, and to identify any management changes that may be required. We do not, in this paper, address the problem of actuator sensor networks that may require immediate user action in response to the current state of the environment.

The field life of a data gathering network is constrained by the energy budget of its individual nodes. Transmitting a packet requires approximately 5 times more energy than taking a sensor reading and storing it locally [4]. Thus, minimising the number of transmissions required is a critical step for extending the lifetime of battery powered nodes, and hence our emphasis in the ROPE protocol on maximising in-network data compression.

2.4. Robustness and Fairness

Each of the network parameters in Sections 2.1 and 2.2 may have a different value under *normal* and *extreme* conditions. For example, 8 hours, is the normally acceptable latency to report a set of readings, but latency of 36 or more hours may be acceptable in extreme conditions. The dynamics of monitored variables changes over time, and thus so does the compression factor. Communication slot availability also changes over time. In response to extreme conditions, *robust* data gathering protocols adapt by increasing capacity or reducing demand. Additionally, long-lived environment sensor networks need to be *self-stabilising* in that when nodes are added or removed from the network the network automatically re-configures itself.

The low power radios used in sensor network nodes are noisy. Not only is there a probability that each message transmitted may be lost, but there are also long range effects in which node to node connectivity may be lost for periods of several hours or even days [2, 4, 11]. Network connectivity also changes as nodes are added and removed from the network, or fail and recover. Although an informed choice of antennae, radio hardware and the positioning of nodes, can minimise radio errors, protocols for environment monitoring still need to adapt their data gathering strategies to changes in radio connectivity caused by changes in the environment that the network can not control. Significantly, poor connectivity is likely to coincide with environment events of particular interest such as rain storms.

A data gathering network must be *fair*, in that resources and opportunities are shared equitably between the nodes and the network gathers data from all reachable parts of its monitored landscape. Fairness is characterised by each node being able to report its data within one latency period: that is $\text{Demand}_s \leq \text{Capacity}_s$ and $\text{Demand}_p \leq \text{Capacity}_p$. Section 3.3 shows how ROPE

adjusts demand and capacity to maximise the range of conditions for which the protocol is fair.

2.5. Dynamics of Environment Variables

The environment monitored by an outdoor sensor network is a 2- or 3-dimensional, often discontinuous, space of values, such as volumetric water content of the soil. The monitored value at any point in this space itself changes over time. Thus we are monitoring a function of one or more environment values over a real number (\mathcal{R}), 3-dimensional landscape:

$$Environment : \mathcal{R}^3 \rightarrow Time \rightarrow Values$$

When monitoring this environment, we attempt to approximate the underlying dynamics of the space of values by a set of discrete readings over time and space. Sensor nodes are placed in the environment so that there is sufficient redundancy to allow for loss of some data, and so that sensor or node failures can be detected.

3. ROPE Protocol Design

In this section we introduce a new protocol, ROPE, to address the application requirements and environment constraints identified in Section 2.

3.1. Soil Moisture Application

The design of the ROPE protocol was prompted by the requirements of a soil moisture study in Banksia woodland sandy soils over an area of approximately 25 hectares (a square of 500 by 500 meters). Monitoring is performed by a sensor network cluster of up to 25 data gathering nodes connected to a base station and GSM gateway [6]. Each node gathers data from up to 4 Echo20 dielectric soil moisture sensors[3] placed up to 2m spatial distance from the node at depths of between 10cm and 1m below the surface of the soil. Larger scale monitoring projects can be constructed using several independent clusters.

3.2. System Model

The ROPE protocol is designed for clusters of monitoring nodes with low power resources (batteries and limited solar recharge) and short range radios. These nodes process data locally, reporting readings of interest to a base station. The base station has higher power resources (larger batteries) than the nodes and a long range GSM mobile phone link to a persistent database store. Network nodes perform coordinated sleep-wake cycles, synchronised by a high power broadcast by the base node. Sensing nodes wake each cycle and listen for a synchronisation signal from the base.

ROPE Parameter	Normal	Extreme
N (nodes)	25	20
NW	1	1
S (mins)	3	15 m
F	[0.05,1.0]	[0.05,1.0]
F _{avg}	0.25	0.50
RP	[1,6]	[1,6]
RP _{avg}	4	2
L (latency hours)	8	36
W (mins)	15	15
C	0.85	0.15
R	1.25	1.25
PW	1800	1800
Demand _s	25	20
Capacity _s	27	21
Demand _p	13	225
Capacity _p	1800	1800

Table 1. ROPE Capacity and Demand

The network is self-stabilising since if a node misses a synchronisation signal, it estimates the time and increases its listening interval for the next round, until it is again synchronised with the base. When new nodes are added to the network, they listen for a full cycle to achieve synchronisation with the network's sleep cycle. If no signal is heard, a new node continues gathering data, but waits for several hours before trying to resynchronise.

At the beginning of each waking cycle, any node which has sufficient data to report, or any node whose reporting period has expired, contends for a transmission path to the base station. After a few packet times, any nodes not required for data delivery return to sleep until the next global reporting round. If more than one node tries to transmit in the same slot, the first requestor is chosen and the rest defer to one of the following slots.

3.3. Soil Moisture Demand and Capacity

Table 1 shows examples of normal and extreme operation of ROPE for soil moisture monitoring. In both cases, capacity exceeds demand as required. In normal conditions a node generates a minimum of 2 packets in a single slot per latency period, typically 13 packets, and in the worst case 120 packets. The maximum packet per slot capacity of ROPE is much larger than required: 1800 packets in 90 seconds awake time. However, energy is not wasted since ROPE nodes only participate during the time they are transmitting their data.

In extreme conditions, the availability of communication slots can fall to 15% or lower. No protocol can satisfy delivery constraints when communication is infeasible for an ex-

tended period. However, ROPE extends the envelope of successful operation as far as possible by first attempting to increase capacity and then decreasing demand. Capacity and demand for slots can be adjusted by increasing latency, decreasing the wake cycle rate or reducing the number of participating nodes. Changes to the waking rate are initiated by the base station, and broadcast to all nodes in its synchronisation message. The new wake cycle should always be a divisor of the normal cycle so that any node that misses the cycle change, or is added later, is still able to find a synchronisation message and so join the network. We can reduce the number of nodes participating by increasing the allowed latency for a proportion (but not all) of the nodes to a much higher value than their neighbours'. For example, if a node hears its neighbour transmitting in extreme conditions, then it can double its own latency. That node may also slow down its sampling rate if necessary, to reduce the amount of data it must store before transmission. Although such nodes will eventually transmit more data, there is plenty of capacity for doing this.

3.4. In-network Data Processing

The goal of local processing of gathered data at each node is to minimise the amount of data to be transmitted, whilst preserving as many interesting features of the underlying spatial and temporal dynamics of the measured landscape as possible. The generous latency requirements typical of environment monitoring applications increase the opportunity for effective compression of the data gathered during that period.

The temporal data series gathered by each node is represented by a sequence of soil moisture values $v_{t_1}^{n,s}$, $v_{t_2}^{n,s}$, $v_{t_3}^{n,s}$, for sensor s on node n . The time t_i is given by a pair, $day \in \{1, 200\}$ and $minute \in \{1, 1440\}$. Soil moisture readings are usually taken at 3 minute intervals. Table 2 summarises the information contained in ROPE data packets. The node identifier, sequence number of the packet, number of packets remaining, and the day of the first reading are included once per packet. In addition each packet contains one or more sets of readings, each consisting of the time in minutes that the reading(s) were taken and 1 to 4 soil moisture readings each tagged with a sensor identifier.

ROPE uses a lightweight data compression scheme based on [9] that approximates the underlying data series by linear fragments. The accuracy of sensor readings from the Echo20 soil moisture probes is typically $\pm 3\%$, but with soil specific calibration the error is reduced to $\pm 1\%$. Output values from the probes range from 375 to 1000 milliVolts. Thus a $\pm 1\%$ volumetric soil moisture error is equivalent to 30 milliVolts. We record only sensor readings which differ by more than a thresh-

Data	Bits
Node ID (usually location)	16
Data Sequence Number $\in [0,256]$	8
Packets Remaining $\in [0,256]$	8
Day $\in [1,200]$	8
Minute $\in [1,1440]$	16
Sensor ID $\in [1,4]$	3
SM milliVolt $\in [300,1000]$	12

Table 2. ROPE Data Packets

old, $T = 30mV$. For each of its probes, s , node n remembers the ultimate and penultimate sensed values, $v_t^{n,s}$ and $v_{t-}^{n,s}$ (respectively). It also remembers the last logged value, $v_{t'}^{n,s}$, for that probe. The difference between pairs of readings is used to determine whether to log or discard the readings. To account for a cumulation of small errors causing a change exceeding the threshold, we also compare the latest reading with the last stored reading. By default, all probe readings are logged at least once per latency period.

1. if $|v_t^{n,s} - v_{t-}^{n,s}| > T$ then log $v_{t-}^{n,s}$ and log $v_t^{n,s}$
2. if $|v_t^{n,s} - v_{t'}^{n,s}| > T$ then log $v_{t'}^{n,s}$
3. if $t - t' \geq L$ (latency hours) then log $v_t^{n,s}$

In Section 4 we show that this simple compression scheme is highly effective, compressing real data by a factor of up to 20-to-1. In future work we plan to allow nodes to adapt their default reading interval depending on the current conditions in the environment.

3.5. Fault-Tolerant Reporting

Communication links are noisy, and so a percentage of packets transmitted, even in good conditions, will be lost. When communication is very poor because of conditions such as bad weather or low batteries waiting for solar recharge, it is better to delay transmission and try again later. A node that is ready to deliver its locally processed data needs to make three judgements:

1. Does it currently have sufficient energy to transmit its logged data? If not, then wait for solar recharge.
2. Does it have a reliable connection to the base station to start transmission? If not, then wait for another slot.
3. Whilst transmission is in progress, is communication reliability sufficiently high to be worth continuing? If not, stop transmitting packets and wait for a more reliable slot.

In order to make these judgements, a node needs feedback from the base station or its neighbours, on the success

of its transmissions. There are trade-offs to be made here. Acknowledging every packet transmitted by a sensor node potentially doubles the energy cost of delivering that data, depending on the size ack packets. Acknowledging every packet is also susceptible to failure because of asymmetric links. However, if no acknowledgements are used, or nodes always transmit multiple times, then significant energy may be wasted in useless transmissions. Our compromise is to use one feedback packet per 10 data packets transmitted. Each feedback packet contains the number of packets received in the last burst, and the sequence numbers of the missing packets. If the loss rate is too high then the sensor node abandons its connection and tries in another slot. Otherwise, a node can choose whether or not to retransmit the lost packets, allowing for data redundancy, and then continues to deliver the rest of its data. A similar feedback scheme is used when a node requests to send its data, but this time only 3 copies of the request are transmitted for feedback from the base node. A node that has no new data to transmit except its latency period health message, sends that packet during the request phase of the protocol.

In summary, sensing nodes in the ROPE protocol take the following actions when ready to report their data or health:

1. Await a base synchronisation message. Otherwise, make a local estimate of cycle starting time and sleep until the next wait cycle.
2. Delay for a random interval within the request phase.
3. If no other requests are heard whilst waiting, then send a request (3x) to transmit data. If a node only has health data to report then transmit that packet. Otherwise sleep until the next wake cycle.
4. Await a request-health feedback packet from the base station.
5. If the feedback is OK and the node has data to transmit then start transmitting bursts of 10 packets. Otherwise sleep until the next wake cycle. Send a negative acknowledgement if necessary to request another copy of the feedback message.
6. Every 10 packets await a data feedback packet from the base station.
7. If feedback shows the delivery rate is OK, and the node has data remaining, then retransmit any lost packets (optional) and await feedback. Repeat steps 5 to 7. Otherwise sleep until the next wake cycle.

4. Implementation Tests

We have implemented and tested components of the ROPE protocol using CSIRO Flecks and Mica2 Motes with

<i>Trial Conditions</i>				
trial period (hours)	72	72	72	24
fleck ID	10	11	12	6
wake cycle (mins)	15	15	15	5
pkts tx per cycle	10	10	10	50
weather	fair	fair	fair	stormy
line of sight	yes	yes	yes	no
<i>Packet Reception Rates</i>				
mean (%)	99.82	99.74	99.06	84.16
std dev (%)	2.50	2.19	2.93	5.56

Table 3. Communication Trials

TinyOS software components. The Fleck uses the same Atmega128 processor as the Mica motes, but has a different radio: a 433Mhz, Nordic nRF903 radio transceiver with range up to 500m [10]. The Fleck is powered by batteries and a solar cell for recharging the batteries; it operates on supply voltage from 1.3 to 5V. Like the motes, Flecks can be programmed with TinyOS. To date, we have used a prototype sensor board for one Echo20 soil moisture probe, built by Fleck researchers; a full sensor board for 8 soil moisture or soil temperature probes is currently under development.

4.1. Data Transmission Performance

In this section we consider how well nodes are able to identify the quality of their communication links, and how much that quality changes over time. The trials measure:

1. the temporal pattern of single hop packet transmission losses over an extended period;
2. the effectiveness of solar recharging in an outdoor environment, using three different solar panels;

Table 3 shows that, in fine conditions, packet delivery is extremely reliable and so the design decision not to use packet by packet acknowledgements is justified. The final column of Table 3 shows that packet reception can be very noisy in poor communication conditions. This confirms previous results on the significant *grey area* of packet reception performance [4, 14]. The trials with Flecks 10, 11 and 12 were run in perfect conditions of sunny, clear days. These tests were run with three Flecks, each within 20m of the base station. We also tested longer transmission ranges with antennas in line of sight and found that, for a base station placed 50cm above the ground transmission range was up to 60m with above 95% reception rates. When the base station was placed 1.2m above the ground then the transmission range was increased to 200m with above 95% reception rates. The final column of Table 3 shows a test performed in extreme conditions: wet and windy weather, over

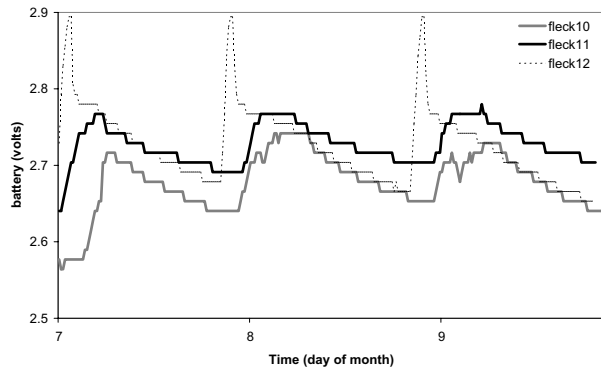


Figure 1. Solar Recharge of Fleck Batteries

approximately 30m without line of sight connectivity between Flecks. Further trials under controlled conditions and longer time periods are planned in order to better understand variations in performance.

The ROPE protocol is designed to minimise the energy used by each node, in order to maximise the overall lifetime of a network. The flecks used in the trials use solar panels to recharge two 2300mAh NiMH batteries. Figure 1 shows the battery recharge for three Flecks over a three day trial period. Three different solar panels were tested for recharging the batteries of each Fleck: two from cheap domestic solar garden lights (flecks 10 and 11 with panel sizes 5.5 x 5.5cm and 10 x 10cm), and a higher quality scientific solar panel (fleck 12 with 11 x 8cm panel). Their three positions provided: 6 hours in full sun (fleck 11), light shade (fleck 10) and only 1 hour per day of full sun (fleck 12). The scientific panel generates much higher current than the garden light panels, and so could be used for applications with high energy requirements. The garden panels, however, are more than sufficient to recharge the batteries for the maximum load generated by the ROPE protocol.

4.2. Measuring the Effectiveness of In-Network Processing

The ROPE protocol uses data compression for energy efficient operation. In order to determine the potential benefits of this approach, we analysed several soil moisture data sets ranging from a trial of 132 minutes in a garden with irrigation to a 46 day field trial in native Banksia woodland. We applied the data compression algorithm outlined in Section 3.4 to these data sets. Both data sets were measured with Mica2 motes with MDA300 sensor boards and two Echo 20 soil moisture probes per node.

As shown in Figure 2 and Table 4 the compression algorithm is extremely efficient. For 132 garden readings at 1 per minute, including a watering event, compression re-

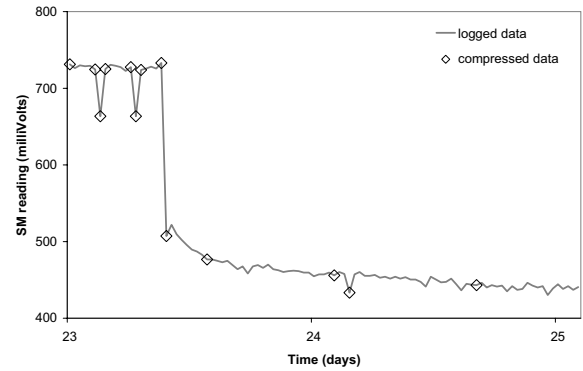


Figure 2. Raw and Compressed Soil Data

number of readings	readings after compression	compression effectiveness
132	32	24%
311	15	4.82%
311	18	5.79%
409	20	4.89%
409	21	5.13%
910	48	5.27%
910	54	5.93%
2208	215	9.74%
2208	287	13.00%

Table 4. Compression Results

duced the original 132 readings to 32. For Banksia field trial data sets, readings were recorded at 30 minute intervals over 8 to 46 days. These data sets were compressed to 5% to 13% of their original size. The trial period in January and February 2005 included two major flooding events, and subsequent drainage of the site, but conditions were otherwise dry and stable. Compression of a sample of over 900 data points over 20 days from the 46 day trial is shown in Figure 2. A few of the data readings were outside the expected range or dynamics of the sensors; this will be further investigated in future work.

5. Conclusions

This paper introduces a novel reactive protocol, ROPE, for outdoor, data-gathering sensor networks. The ROPE protocol adapts the data reporting demands of its environment dynamics to the communication capacity of its environment. The protocol is opportunistic in choosing good transmission times, and making effective use of in-network processing to reduce the network's overall transmission load. Thus, in changeable environments, ROPE is more en-

ergy efficient than existing protocols which maintain permanent routing trees and schedules, and also more robust. We are currently implementing the full ROPE protocol for a single hop network in which the transmissions of each sensor node are able to reach the base node. A multi-hop version of the protocol has been tested in simulation, and we also plan to implement the multi-hop version in field trials.

Acknowledgements

The author is grateful for help from members of the CSIRO Fleck group in Brisbane, Peter Corke, Pavan Sikka, Phil Valencia and Leslie Overs, who provided Fleck hardware and software, a prototype soil moisture sensor board and driver for the Echo20 sensor for this work; John Cardell-Oliver and anonymous Em-Nets referees suggested many improvements to the paper; Keith Smettem and the WA Water Corporation supported the large data set field trials at the Water Corporation's Pinjar site; and Ian Marshall, and members of the SECOAS project, for insightful discussions on adaptive algorithms for sensor networks.

References

- [1] Daniel Aguayo, John Bicket, Sanjit Biswas, Glenn Judd, and Robert Morris. Link-level measurements from an 802.11b mesh network. In *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 121–132. ACM Press, 2004.
- [2] Rachel Cardell-Oliver, Keith Smettem, Mark Kranz, and Kevin Mayer. Field testing a wireless sensor network for reactive environmental monitoring. In *International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Melbourne*, December 2004.
- [3] Decagon Echo-20 Dielectric Aquameter. [Online] Available at <http://www.decagon.com/echo/>, April 2004.
- [4] Alessio Falchi. Sensor networks: Performance measurements with motes technology. Master's thesis, Dept. of Information Engineering, University of Pisa, 2004.
- [5] Antonio Gonzalez, Ian W Marshall, and Lionel Sacks. A self-synchronised scheme for automated communication in wireless sensor networks. In *International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Melbourne*, December 2004.
- [6] K. Mayer and K. Taylor. An Embedded Device Utilising GPRS for Communications. In *International Conference On Information Technology and Applications, Bathurst, Australia*, 2002.
- [7] Mica Motes Specifications. "[Online] www.xbow.com/Products/Wireless_Sensor_Networks.htm Accessed May 2003".
- [8] Chris Roadknight, Laura Parrott, Nathan Boyd, and Ian W Marshall. A layered approach to *in situ* data management on

a wireless sensor network. In *International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Melbourne*, December 2004.

- [9] T. Schoellhammer, B. Greenstein, E. Osterweil, M. Wimbrow, and D. Estrin. Lightweight temporal compression of microclimate datasets [wireless sensor networks]. In *Proceedings of 29th Annual IEEE International Conference on Local Computer Networks*, pages 516 – 524. IEEE, 2004.
- [10] Pavan Sikka, Peter Corke, and Leslie Overs. Wireless sensor devices for animal tracking and control. In *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN'04)*, pages 446–454. IEEE Computer Society, 2004.
- [11] Robert Szewczyk, Alan Mainwaring, Joseph Polastre, John Anderson, and David Culler. An analysis of a large scale habitat monitoring application. In *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, pages 214–226. ACM Press, 2004.
- [12] TinyOS: a Component-Based OS for the Networked Sensor Regime. [Online] <http://webs.cs.berkeley.edu/tos/>, Sep 2003.
- [13] Gilman Tolle and David Culler. Design of an application-cooperative management system for wireless sensor networks. In *Second European Workshop on Wireless Sensor Networks (EWSN) Istanbul, Turkey*, January 2005.
- [14] Alec Lik Chuen Woo. *A Holistic Approach to Multihop Routing in Sensor Networks*. PhD thesis, Computer Science, University of California, Berkeley, 2004.
- [15] Wei Ye, John Heidemann, and Deborah Estrin. Medium Access Control with Coordinated Adaptive Sleeping for Wireless Sensor Networks. *IEEE/ACM Transactions on Networking*, 12:493–506, 2004.

APPENDICES

Multi-hop ROPE

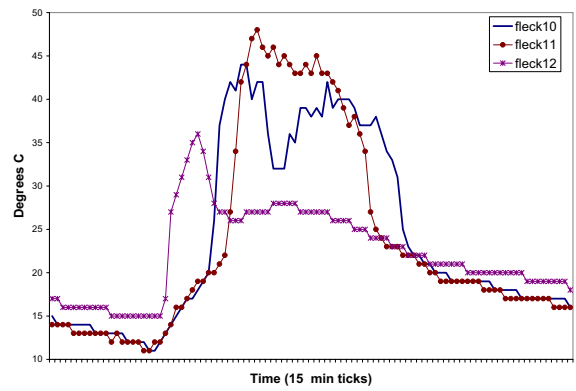
We are currently implementing ROPE for a single hop network in which the transmissions of each sensor node are able to reach the base node. These conditions are satisfied by the high quality radios in Fleck hardware, and our Pinjar field site. We have also tested in simulation a ROPE implementation for a multi-hop network since, even in a single hop network, the transmission range of nodes may be reduced to 18% of normal range in poor weather conditions [4], so creating a multi-hop network for a time.

In multi-hop ROPE, if the base node hears a request packet, then it replies immediately to the requesting node. Neighbouring nodes overhearing the request also listen for the reply. Thus, if a node is out of range of the base, then one of its neighbours will observe that there is no base reply, and so can offer to relay the message. The requestor will hear its neighbour forwarding its request, and so sets that neighbour as its parent. In the case of an asymmetric link, where the requestor does not hear a neighbour, it retransmits its request, asking for a new forwarder. So the request cycle continues until the base station is reached. Each

node knows its location (either GPS at installation, or by estimating its hop count to the base), and so can estimate its “goodness” as a router to the base for the requesting node. The base station, if it can hear a request strongly, is the best neighbour, and so it has highest priority to respond. Routing nodes choose the time of their responses so that better neighbours respond earlier and are chosen as the next link in the path. This gives a local, distributed scheme for creating best paths that is scalable as the network size grows.

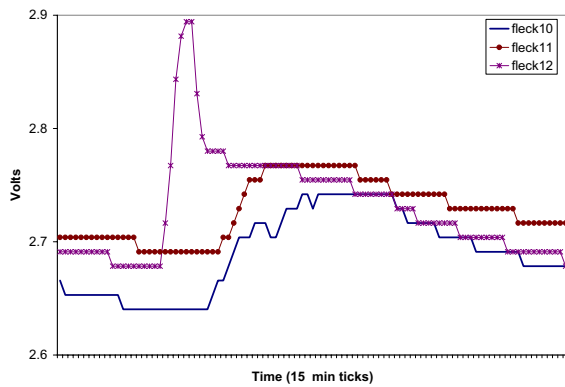
If any node hears further requests whilst a path is being set up, then it sends a wait message to the requestor who defers its request until a future waking cycle, as in the single hop ROPE protocol. All forwarding nodes piggyback information about the number of requests heard so that the base station is aware of possible congestion in the network and can increase the rate of waking cycles if necessary.

Fleck Trial On-Board Temperature



Energy Trial Results

Fleck Trial Battery Voltage



Fleck Trial Battery Solar Charge

