

Field Testing a Wireless Sensor Network for Reactive Environmental Monitoring

Rachel Cardell-Oliver¹, Keith Smettem², Mark Kranz¹ and Kevin Mayer³

¹*School of Computer Science & Software Engineering, The University of Western Australia, {rachel,kranzm01}@csse.uwa.edu.au*

²*Centre for Water Research, The University of Western Australia, smettem@cwr.uwa.edu.au*

³*Faculty of Engineering and Information Technology, The Australian National University and CSIRO ICT Centre, Kevin.Mayer@csiro.au*

Abstract

Wireless sensor network technology promises to reveal fine-grained, dynamic changes in monitored variables of an outdoor landscape. But there are significant problems to be overcome in translating this vision to working systems. This paper describes the design and implementation of a reactive, event driven network for environmental monitoring of soil moisture and evaluates the effectiveness of this solution. A novel feature of our solution is its reactivity to the environment: when rain falls and soil moisture is changing rapidly, measurements are collected frequently, whereas during dry periods between rainfall events measurements are collected much less often. Reactivity allows us to focus on dynamic responses and limit the amount of useless data gathered, as well as improving robustness and network lifetime. The main contribution of the paper is to demonstrate that a reactive sensor network can deliver useful data on soil moisture responses to rainfall. Field trial results on the reactivity, robustness and longevity of the network are evaluated, and future improvements proposed.

Keywords: Reactive monitoring, Sensor Network

1. INTRODUCTION

Environmental monitoring is a significant driver for wireless sensor network research, promising dynamic, real-time data about monitored variables of a landscape and so enabling scientists to measure properties that have not previously been observable. However, there are significant problems to be overcome in translating this vision to working systems. This paper describes the design and implementation of a reactive network for environmental monitoring and evaluates the effectiveness of the network for data gathering using both laboratory and field tests. The system investigated in this paper comprises sensor network of micro-controller nodes (Berkeley Mica2 nodes [12]) communicating with a Superlite GSM modem [14] over the mobile phone network to a database viewable from an internet web page [9].

Considerable advances have been made in recent years in hardware [12] and software [16] for building wireless sensor networks. However, to ensure effective data gathering by sensor networks for monitoring remote outdoor environments, the following problems remain:

- **reactivity:** the ability of the network to react to its environment, and provide only relevant data to users;

- **robustness:** the ability of network nodes to function correctly in harsh outdoor environments;
- **network lifetime:** maximising the length of time the network is able to deliver data before nodes' batteries are exhausted.

A novelty of our design is its reactivity to the environment. Reactivity is needed to ensure that the most interesting data is gathered, given the resource restrictions of sensor networks. For example, a soil moisture sensor network needs to react to rain storms: frequent soil moisture readings should be collected during rain (say, every 10 minutes), but only infrequent readings (say, once a day) are needed when it is not raining. Achieving reactivity is a challenging problem because the nodes that monitor rain storms may be separated from the nodes monitoring soil moisture, and yet all nodes in the network need to minimise the time spent sending, receiving and listening to messages. Furthermore, the start of rain can not be inferred simply by changes in soil moisture, and so both rain and soil moisture sensing is needed in order to observe the phenomena of interest in this environment. Making wireless sensor networks more reactive is an important step towards improving their usefulness, increasing their lifetime and so towards enabling them to become effective environmental monitors, especially so in remote sites typical for many Australian applications.

The main contribution of this paper is the design and test of a sensor network that successfully meets the goal of reactivity, and that demonstrates satisfactory robustness and network lifetime. Improving the performance of the network is the subject of ongoing work. The long term aim of our research is to develop components for sensor networks that can be simply combined to create reactive, long lived networks for a variety of environmental monitoring applications including irrigation in agriculture or urban settings, monitoring of water catchments and dry-land salinity management.

A. Related Work

The first generation of sensor networks to be tested in field trials used periodic monitoring with a pre-set sensing regime of readings taken at regular intervals and relayed through a fixed networking tree to a base station [8], [2]. An important extension of that approach has been to view the sensor network as an event-driven database that can be queried [7], [6]. For example, a base station might request readings from all nodes

in a certain area with temperature above a threshold value. In TinyDB [7] a network is viewed as a collection of producers who generate data. The data is accessed by a user-in-the-loop who sends queries to producer nodes via a base station. Responses are routed back to the base station and thus so to the user. Our reactive network extends the user-in-the-loop query approach by allowing any network node to generate a request for information from other nodes or to advertise its data to interested nodes. Directed diffusion protocols offer a similar functionality to ours [6], but in diffusion the emphasis is on discovering and maintaining a routing tree for data delivery, whilst here we focus on the application logic of reactive information exchange. The goals of our reactive network are similar to the *active sensor* approach [1] that provides for dynamic programming of nodes for application-specific tasks. Whilst [1] focuses on the programming framework needed to enable mobile control scripts to move between nodes in a sensor network, we focus on application specific aspects: which nodes need to communicate, and how effective is this strategy for managing a sensor network in the field without a human-in-the-loop or a pre-set sensing regime.

2. REACTIVE SENSOR NETWORK DESIGN

A. Monitoring Goals

Our goal is to characterize the transient spatial variability of water infiltration and consequences for the water balance and groundwater recharge under Banksia woodland on the sandy Gngangara groundwater mound, North of Perth, Western Australia. The objective is to provide better process understanding for management of the groundwater resource and ecology as well as providing improved parameterisation for the Perth groundwater model that is used as a management tool to assess safe water abstraction levels. A key question is whether recharge occurs as a uniform advective front or as non-uniform preferential flow. Preferential flow results in a considerable volume of the soil mass remaining quite dry, whilst the infiltrating water 'fingers' down through the profile more locally. The phenomenon can be triggered by a combination of factors including: non-wetting behaviour of the surface soil and concentration of rainfall into localised wet spots by vegetation architecture. Because the water contents of sandy soils respond very rapidly to rainfall, it is imperative to have measurements at short time intervals during storm events. Once the rainfall has drained through the soil (usually in a matter of hours), less frequent monitoring is appropriate because water content changes are then small and driven by root water uptake in response to vegetation transpiration demands. Examples of rain events and soil moisture changes over time gathered by our sensor network are shown in Figure 1.

B. Network Architecture

The soil moisture sensor network is based on Mica2 motes and MDA300 sensor boards [12]. It uses the following components:

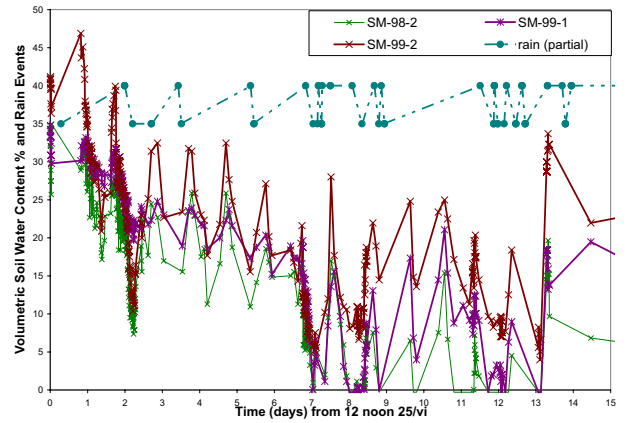


Fig. 1: Rain Events and Soil Moisture Readings over 15 days

- *soil moisture sampling* nodes, each attached to two soil moisture sensors,
- a *rainfall sampling* node,
- a *base* node, linked to a Superlite GSM gateway [14],
- *routing* and *gathering* nodes for transporting soil moisture readings from the sampling nodes to the base node, and rainfall information to the sampling nodes.

The nodes are arranged in branches from the base node, with sampling nodes as leaves. The rainfall node is placed near the base node, both for the security of the rain gauge, and to facilitate data dissemination to all sampling nodes through the base node and routing branches. The current network deployment has a single branch with three sampling nodes, but more generally the architecture supports many branches in a wheel spoke pattern [5], with leaves both at the end of a branch and along its length.

C. Delivering the Data

The Superlite [14] is a single board computer containing a Sony Ericsson GSM module. It is connected to the sensor network by attaching the base node to an MIB510 programming board, and then connecting the programming board to the Superlite using a crossover serial cable. The Superlite, together with the mobile monitoring infrastructure [10] acts as a virtual connection between the sensor network and an online database. Once in the database, the TinyOS messages can be retrieved and decoded using a specially devised SOAP based web service [9]. By utilising GPRS to connect the Superlite to the Internet, we have given the sensor network an inexpensive, always online presence. The library logic on the Superlite is generic in the sense that it manages the Internet connection, as well as relaying messages from the sensor network to the database. The application logic specific to this experiment is only a handful of lines of code [11].

D. Sensor Network Hardware

For this application, we have customised a set of Mica2 433 MHz motes [12] to make them sufficiently robust for outdoor

monitoring. In order to improve radio reception we added a length of coaxial cable to each antenna, allowing it to be raised over a metre from the ground, so providing a significant improvement in the percentage of packets successfully delivered. In the Banksia woodland antennae can be attached to leafless tree branches while the node, battery and soil moisture probes are placed on the ground with some protection from the rain. Soil moisture is measured by two Decagon Echo-20 dielectric sensors [4] attached to each sampling node using the MDA300 sensor board [12]. The Echo-20 probes were chosen because they are a well known standard for measuring soil volumetric water content, and because TinyOS software for reading the probes was available for the MDA300 board [12]. A Decagon Echo rain gauge is also attached to its node via the MDA300 sensor board. We have also customised the standard battery configuration for Mica2 motes to increase network longevity, testing NiMH and LiSO₂ batteries as at James Reserve [17]. Configurations and unit costs for sensor nodes and batteries are detailed in [3].

E. Reactive Sensor Network Software

A reactive network can not simply set its sampler nodes to sleep for a day between soil moisture readings and then awake to check environmental conditions because nodes must be ready to respond frequently as soon as an event of interest occurs, in our case a rainfall event. Furthermore, when a sampling node awakes, all the router and gatherer nodes that it relies on to deliver its message to the base node must also awake. Clock drift between nodes makes the co-ordination of waking cycles a difficult task.

SMAC is a MAC protocol designed to address the problem of energy efficient, co-ordinated sleeping [18], [13] and so is well suited to our application. Nodes synchronise with their neighbours for a cycle of sleeping and waking. In the lowest duty cycle of 1%, nodes sleep for approximately 12 seconds and then awake for the co-ordinated transmission and reception of messages by reliable unicast or by broadcast. The SMAC software stack also provides a reliable physical layer implementation with Manchester encoding.

We have implemented a reactive protocol for gathering soil moisture data and advertising rain events using event-condition-action rules. The rain node checks for rain each SMAC cycle (12 seconds), but only sends a *rate-is-wet* message when it detects 1mm of rain. Two hours after detecting a 1mm rainfall event, the rain node sends a *rate-is-dry* message to indicate that the rain has finished. If there are additional triggers during a rain period, then the timer is reset to 2 hours each time. In this way, sampler nodes will be gathering data at high frequency during rain events, when interesting changes in soil moisture can be expected to occur. The gatherer node waits up to one minute to aggregate soil moisture messages, and then forwards the samples towards the base station.

TinyOS, and the NesC programming language use a modular, interface driven syntax, but there is still a high degree of interaction between components. To program the nodes so that different nodes can use different configurations of the same

code-base requires effective high-level abstractions of basic behaviour. The abstractions required for our application are:

- Sensing activities
 - Soil moisture and health data,
 - Rain-gauge triggered events
- Timing activities
 - Clock synchronisation,
 - Timer setting and triggering,
 - Time-stamping
- Logging activities
 - Writing samples to EEPROM as backup
- Communication Activities
 - All node-node and node-serial port communication

In each TinyOS-NesC package the available activities (commands and events) of the package are exposed by the interface of a single component, as shown in Figure 2. This component is responsible for accessing all other sub-components, and translating between context-sensitive and context-free formats. For example, the Sensing package uses general-purpose interfaces to access the sensor board's Analog-Digital-Converter (ADC), but provides commands to retrieve context-sensitive *Soil Moisture samples* and *Health samples* giving current battery voltage. This component is also responsible for all configuration and initialisation of the package. For example, the Logging package requires memory to be allocated before writing, and the Communication package requires configuration of radio parameters.

It was found that this approach worked well in fine-tuning specific behaviour for different, specialised nodes, while retaining communication consistency between them. This approach also supported the development of the network as a series of logical steps, where adding new features or new specialisations required little effort. For example, adding code to increase sampling during rain involves utilising the RateHandler to notify the Timer and building data aggregation requires a few lines of application code between the receiving and forwarding of samples.

Working with the NesC paradigm assisted us to produce this structure. Defining application specific interfaces makes different node specialisations concise and easy to produce. However, the underlying implementations still use the generalised interfaces, and are thus compatible with existing TinyOS components. Also, since NesC uses an interface-implementation wiring mechanism, it would be easy to substitute new radio or sensing modules, without affecting our overall design.

3. EVALUATION OF THE NETWORK

The goal of the network is to provide useful soil moisture data focussing on dynamic response to rainfall events. To evaluate the success of the network we consider three aspects of performance that contribute to this goal: responsiveness, robustness and longevity.

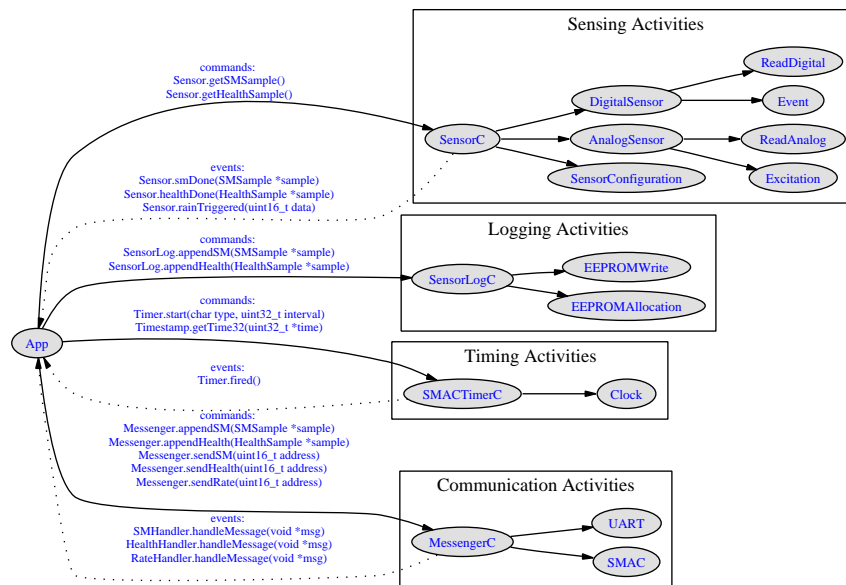


Fig. 2: Reactive Monitoring Software Architecture

A. Responsiveness

The field network was responsive to rainfall events in that it was usually able to respond by modifying the monitoring rate of the sampling nodes. Figure 1 shows that rainfall events correlate reasonably well with high frequency soil moisture monitoring. The correlation is not perfect because of message losses in the network. As a result of this reactive policy a manageable amount of data was returned via the Superlite mobile phone link to an internet database. We were able to monitor soil moisture through a web page as changes took place.

B. Robustness

The overall robustness of a sensor network depends on many factors: the ability of network protocols to recover from errors, the quality of radio transmissions, the quality of the gateway link, how well the nodes withstand harsh outdoor conditions and misadventure from wildlife or humans, and the accuracy of the sensors used for measuring soil moisture and rainfall, as well as each node's lifetime provided by its battery power source. If the network does not perform reliably in any one of these areas, then it may fail to deliver data from the field.

1) *Quality of Mica2 Radio Links:* An important aspect of network robustness is the reliability of radio message transmissions between neighbouring nodes in the network, because one failure in a sequence of transmissions causes the loss of that data.

Despite using reliable SMAC unicast for all message transmissions, the delivery of rainfall and soil moisture messages from one end of the network to the other was far from 100% successful. The MAC layer retransmits a message up to 7 times when no acknowledgement is received. Figure 3 shows the pattern of message delivery success. Each point in Figure 3 shows the number of packets delivered through the network

and GSM to the database per 10 packets expected. The measured loss rate is for messages that have travelled 5 hops through the sensor network, and then over the Superlite GSM link, and so represents the worst-case probability of loss in the network. Of a total of 434 soil moisture messages expected over a period of approximately 270 hours, 277 messages were logged in the database, an overall delivery rate of 63.8%. During this time over 130 health and rate messages were also delivered successfully to the database.

The loss rate of soil moisture messages in field trials was disappointing, because in laboratory trials with the same hardware and software the delivery rate was close to 100%. The loss rates in field trials were time related, with significant changes in reliability during different time intervals. Our laboratory tests confirm the results of [13], where in a wide corridor without human traffic the probability of receiving a transmitted packet is 100% for a receiver node placed 1 to 18m from the transmitter but falls away to almost nothing beyond 18m. However, in outdoor settings the situation is more complex with good reception possible at distances of up to 30m in woodland but bad reception can occur even between nodes only 5m apart. In addition, the delivery of duplicate messages indicating lost acknowledgements suggests that links between nodes are sometimes asymmetric.

In order to minimise the number of messages lost because of poor radio connectivity, the protocol includes a diagnostic phase for each node when it is first turned on in which the node connects to its upstream neighbour and unicasts a packet every 30 seconds. We then monitor the number of packets received at the base node and if necessary move and restart the node until good quality reception is achieved. Since a unicast transmission in SMAC includes a 4 way handshake of RTS-CTS-DATA-ACK this method also tests, in part, the symmetry (or otherwise) of the link between the test node and

its neighbour. The diagnostic phase for each of the node links in the Pinjar network showed each link to be highly reliable. However, link quality varies over time, and over the one month trial period the links showed intervals of high loss as well as intervals with very low losses.

Another method for improving robustness is to have more than one path between critical pairs of nodes communicating in the network. Then when a routing node is unable to reach one neighbour it simply resends its message to another. Although this approach should improve reliability, it is by no means guaranteed to do so, since radio transmission problems between nodes are not necessarily independent. For example, if air moisture is very high, then all transmission success rates will be reduced. The cost of, say, doubling the number of nodes in the network to reduce data loss rates, needs to be balanced against possibly small improvements in packet delivery success. In future work we will investigate the benefits of increasing local buffering of data for long periods to address the observed time dependent problems with radio connectivity.

2) *Quality of Gateway Transmissions:* The current library implementation on the Superlite does not buffer messages from the sensor network, and consequently does not guarantee delivery. A single attempt is made to send each message that it receives over the serial connection, thus a loss of GPRS connection or marginal GSM reception may lead to message loss.

The mobile infrastructure back-end logs all Superlite connections and disconnections, and by analysing this data over the first 12 days of operation we have been able to produce a lower bound on the number of packets lost due to lack of connection. During this 288 hour period, the Superlite connected to the back-end 838 times. 30% of these connection lasted less than 4 minutes, and a further 3.5% lasted between 4 and 5 minutes. Analysis of the logic used in the Superlite application together with past experience indicates that nearly all sessions lasting less than 4 minutes, and approximately half of the sessions lasting between 4 and 5 minutes, terminate as a result of a failed attempt to send data from the Superlite to the back-end. This translates to roughly 226 lost soil moisture,

rate and health messages causing disconnections. All other disconnections may or may not be as a result of a failed attempt to send a message and so we cannot estimate how many of them led to lost messages.

Out of the 837 periods of time the Superlite was not online, for 49% of them the device was offline for less than 3 seconds. Only 8% of the disconnections lasted longer than 2 minutes, and of those, 16% (1.4% of the total) lasted longer than 6 minutes. We suspect that a large proportion of disconnections that lasted for less than 3 seconds were due to the watchdog timer on the Superlite restarting the application. We know from experience and testing that an existing bug in the library allows the watchdog timer to expire with the consequence that the device tries to reconnect to the back-end before the back-end has realised the Superlite is disconnected in the first place. This leads to the backend logging the disconnect and almost immediately thereafter logging the reconnection. The low number of extended off-line periods confirms that the sensor network is located in a marginal GSM coverage area. It is likely that continuously changing environmental conditions aided and abetted the coverage over the 12 day period, leading to unstable GPRS connections.

Although an external whip antenna was used, the use of a better antenna, or a better location for the antenna might be required for future experiments. Secondly, the addition of library code to the Superlite that can buffer unsent messages for retransmission when on-line should also significantly enhance the reliability of the Superlite to back-end connection.

3) *Weather-proofing:* Another aspect of robustness is the ability of nodes and sensors to survive in a harsh outdoor environment. Our field site is native bush land with Banksia scrub, native grasses, and sandy soil, visited by kangaroos, birds and (rarely) humans. We designed and built outdoor housing for the nodes and their batteries using off-the-shelf rain-proof boxes with a rubber seal and with grommets to seal the exit points for the soil moisture probe and antenna cables. Nodes and batteries are securely mounted in these boxes on raised platforms to prevent damage if the node box is disturbed by wind, rain or wildlife. The boxes are placed on the ground, protected by vegetation, and the antenna attached to a tree branch, away from leaves or other moisture sources wherever possible. Soil moisture probes have long cables and so can be placed at some distance from their node if desired, although care must be taken not to leave trailing cables to entangle wildlife. Over a month of operation, that included rain storms of up to 10mm in an hour, water did penetrate some node's housing. Nodes continued to transmit their data, but some sensors returned soil moisture readings of 0.

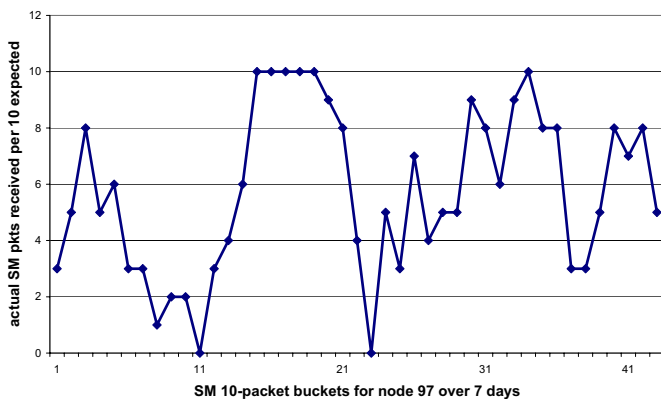


Fig. 3: Packet Reception Success Rates over Time

C. Maximising Network Lifetime

The problem of power management has been recognised as crucially important for wireless sensor networks, but it has also proved difficult to achieve a practical solution [8], [15], [17], [18]. The power management approach designed for Berkeley mote hardware is for motes to alternate between *activity* and *sleep* states. Power draw during the active period will be

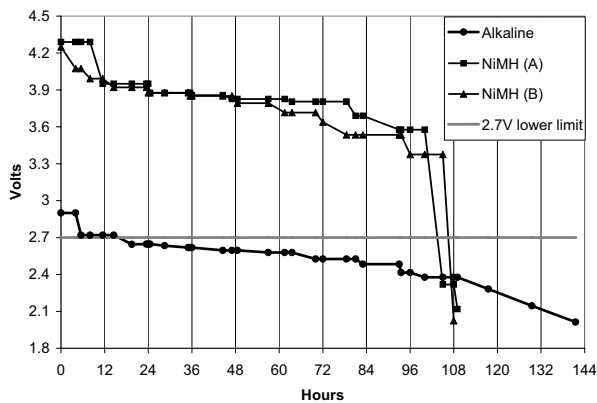


Fig. 4: Mica2 Mote Battery Lifetime at 100% duty cycle

typically from 5 to 20 milliAmps and during sleep the draw is 5 μ Amps. Thus, by choosing a duty cycle that maximises the percentage of time each mote is asleep, it should be possible to extend a mote's lifetime in the field to many months or even years. However, in our mote configuration using MDA300 sensor boards and the SMAC and sensorib code from TinyOS\contrib [13] our measurements show a minimum 8 milliAmp load in both sleep and waking states since in SMAC the radio sleeps, but the CPU does not. The total application lifetime we observed for various battery configurations, confirms our average load measurements.

Figure 4 shows the comparative life of Alkaline vs NiMH batteries at a 100% duty cycle. Although the capacity of a battery (measured in AmpHours) is the most important indicator of the lifetime that can be expected for each node for a given duty cycle of the processor, the decay curve of the battery must also be taken into account since the motes will only give satisfactory sensing and radio transmission performance when battery voltage is above 2.7V. The recommended maximum voltage for the motes' Atmega 128 processor is 3.3V. However, we had no problems with the NiMH batteries initially delivering 4V. Figure 4 shows that the voltage delivered by alkaline batteries falls below 2.7V after 18 hours as opposed to 100 hours for the NiMH batteries, although the overall lifetime of the alkaline batteries was 40% longer than the NiMH configuration. Longer network field-life was achieved using 8000mAh SAFT LiSO₂ batteries. In our field trials SAFT batteries provided 30 days network lifetime for the rain gauge mote, and at least 16 days for soil moisture and router motes. In future work we plan to investigate the use of solar panels for trickle-recharging node batteries.

4. CONCLUSIONS

We have described the design and implementation of a novel reactive sensor network for monitoring soil moisture and evaluated the reactivity, robustness and longevity of the network in the field. The Pinjar network meets the goal of providing useful data on dynamic responses of soil moisture to rainfall. Future work will focus on addressing the limitations of the

current prototype in robustness of packet delivery and network longevity, and in guaranteeing network response to events of interest. We also plan to generalise our event-condition-action framework for programming reactive sensor networks.

ACKNOWLEDGEMENTS

The authors are grateful for funding for this project from the Western Australian Water Corporation and also for the use of their Pinjar field site.

REFERENCES

- [1] Athanassios Boulis and Chih-Chieh Han and Mani B. Srivastava. Design and Implementation of a Framework for Efficient and Programmable Sensor Networks. In *The First International Conference on Mobile Systems, Applications, and Services (MobiSys 2003) San Francisco, CA, May 5-8, 2003*.
- [2] Richard Beckwith, Daniel Teibel, and Pat Bowen. Report from the Field: Results from an Agricultural Wireless Sensor Network. In *First IEEE Workshop on Embedded Networked Sensors. To appear, 2004*.
- [3] Rachel Cardell-Oliver, Keith Smettem, Mark Kranz, and Kevin Mayer. Field Testing a Wireless Sensor Network for Reactive Environmental Monitoring, Technical Report UWA-CSSE-04-003, August 2004. [Online] Available at http://www.csse.uwa.edu.au/research/Technical_Reports/UWA-CSSE-04-003/Report-04-003.pdf.
- [4] Decagon Echo-20 Dielectric Aquameter. [Online] Available at <http://www.decagon.com/echo/>, April 2004.
- [5] Deepak Ganesan, Razvan Cristescu, and Baltasar Beferull-Lozano. Power-efficient Sensor Placement and Transmission Structure for Data Gathering under Distortion Constraints. In *Proceedings of the Symposium on Information Processing in Sensor Networks 2004, Berkeley, California, 2004*.
- [6] John Heidemann, Fabio Silva, and Deborah Estrin. Matching data dissemination algorithms to application requirements. In *Proceedings of the First International Conference on Embedded Networked Sensor Systems*, pages 218–229. ACM Press, 2003.
- [7] Samuel Madden, Michael Franklin, Joseph Hellerstein, and Wei Hong. TAG: a Tiny Aggregation Service for Ad-Hoc Sensor Networks. In *5th Symposium on Operating Systems Design and Implementation*, pages 131–146, December 2002.
- [8] Alan Mainwaring, Joseph Polastre, Robert Szewczyk, David Culler, and John Anderson. Wireless Sensor Networks for Habitat Monitoring. In *Proc. First ACM International Workshop on Wireless Sensor Networks and Applications, September 2002, Atlanta, Georgia, USA, 2002*.
- [9] K. Mayer. UWA Pinjar Web Page. [Online] <http://mobile.act.cmis.csiro.au/kevin/tinybridge/UWAPinjar.aspx>, June 2004.
- [10] K. Mayer. Over the Air Programmable Stand-Alone Controllers Using GPRS Technology, 2002. Department of Engineering. Canberra: The Australian National University.
- [11] K. Mayer and K. Taylor. An Embedded Device Utilising GPRS for Communications. In *International Conference On Information Technology and Applications, Bathurst, Australia, 2002*.
- [12] Mica Motes Specifications. [Online] www.xbow.com/Products/Wireless_Sensor_Networks.htm Accessed May 2003.
- [13] Download s-mac source code for motes. [Online] Available at <http://www.isi.edu/ilense/software/smac/> Accessed June 2003.
- [14] Superlite E Intelligent Terminal Product Brochure. [Online] <http://www.call-direct.com.au/files/SuperliteEit.pdf>, April 2004.
- [15] Robert Szewczyk, Joseph Polastre, Alan Mainwaring, and David Culler. Lessons from a Sensor Network Expedition. In *Proceedings of the First European Workshop on Wireless Sensor Networks, 2004*. Berlin Germany, January 2004.
- [16] TinyOS: a Component-Based OS for the Networked Sensor Regime. [Online] <http://webs.cs.berkeley.edu/tos/>, Sep 2003.
- [17] Mike Wimbrow and Mohammad Rahimi. Email correspondence February to June 2004.
- [18] 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.