Adaptive Sampling and Routing in a Floodplain Monitoring Sensor Network

Jing Zhou
University of Southampton
Southampton, SO17 1BJ
Tel: +44 23 8059 8367
Email: jz@ecs.soton.ac.uk

David De Roure
University of Southampton
Southampton, SO17 1BJ
Tel: +44 23 8059 2418
Email: dder@ecs.soton.ac.uk

Sanjay Vivekanandan
University of Southampton
Southampton, SO17 1BJ
Tel: +44 23 8059 4490
Email: smv@ecs.soton.ac.uk

***Abstract*— We describe the design of a flood warning system which uses a set of sensor nodes to collect readings of water level and a grid-based flood predictor model developed by environmental experts to make flood predictions based on the readings. The reporting frequency of sensor nodes is required to be adaptive to local conditions as well as the flood predictor model to optimize battery consumption. We therefore propose an energy aware routing protocol which allows sensor nodes to consume energy according to this need. This system is notable both for the adaptive sampling regime and the methodology adopted in the design of the adaptive behavior, which involved development of simulation tools and close collaboration with environmental experts.**

***Index Terms*—Adaptive sampling, energy awareness, communication system routing, environmental sensor networks.**

I. Introduction

Pervasive computing technologies provide exciting new opportunities for monitoring the natural environment, such as measurement of water levels and air pollution. Significantly, these technologies make it possible to deploy more devices in order to obtain more data more often, and this greater richness of data is set to create a powerful impact on environmental monitoring and decision-making. Traditional solutions involve dataloggers from which data is collected periodically in person or via telemetry. With wireless communications and energy drawn from local sources such as solar cells, devices can be deployed without the constraints of having to wire them up or make them accessible, and data can be conveyed when needed.

Achieving these benefits within the natural environment brings a number of technological challenges. It is often also the case that deployment is expensive and it may be difficult or impossible to access the devices again later, in contrast to working with handheld pervasive devices. Devices need to withstand harsh environmental conditions. There may also be large numbers of nodes and they need to have coordinated behavior, but at the same time we have to assume that node failure, temporary or permanent, is likely to occur—the natural environment is a place of change.

One of the most substantial challenges, and our focus here, is optimizing the consumption of electrical power. At times when stored power is not being fully replenished, the more often data is sent then the more likely it is that the device will then not have sufficient power to continue its function.

In this paper we describe a flood early warning system, FloodNet. Flood damage represents a major ongoing cost and risk may be increasing due to land-use change, climate change and flood-prone investment. When a flood occurs, the cost of damage has a clear correlation with both the depth of the flooding and the time in advance at which warning is given. By applying pervasive computing technologies on the floodplain we have the potential to obtain better data from which to make predictions, and we can do this in a timely manner in order to improve warning times. Such systems provide an excellent illustration of the benefits and challenges of pervasive computing in the environment. Deployment is facilitated by wireless technologies but we have issues of power for the devices and the need for very long unattended periods of operation. This scenario also emphasizes the energy optimization challenge, because the data is most important during flood conditions and this is when solar energy is typically least available.

The fundamental tradeoff between the need for timely data and the need to conserve energy is the research focus of the project. The goal is to make the system adaptive so that the sampling and reporting rates of the devices vary according to need, conserving power and minimizing the data volume required. Some intelligence is required in this adaptation, because the importance of a device at a given moment will depend upon both its local conditions (e.g. the residual battery power and the link cost) and its role in relaying data from other devices, each of which will vary dynamically according to circumstances. The key element of our strategy is the use of a centralized flood predictor model (see Sect. III) which was developed by environmental experts. From this the priorities for collecting samples from each sensor can be determined. Reducing activity of individual sensor nodes and minimizing the data volume required help prolong network lifetime; i.e. the time until the network fails[1] due to insufficient power. The use of the predictor model within this adaptive sampling regime is a distinctive feature of this work.

[1]In the context of FloodNet, the network failure means that data messages cannot be generated by sensor nodes or cannot be retransmitted due to energy depletion.

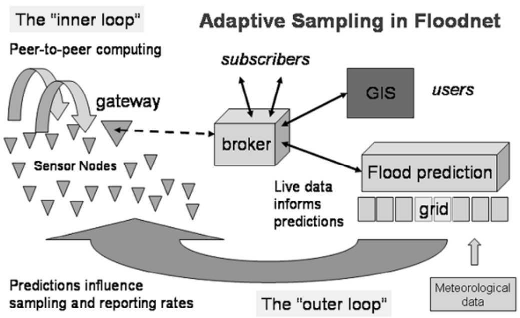

Fig. 1. The adaptive sampling control loops

This new technology in the field makes new demands on the back-end processing. The higher temporal and spatial density of data, and faster turnaround times, present a significantly greater computational task in the flood modelling. For this reason we turn to grid computing for the back-end processing. This combination of Grid and Pervasive computing is another distinctive feature.

The system is illustrated in Fig. 1, which also shows the possible control loops in the adaptive sampling regime. The outermost loop enables the flood predictions to influence the reporting rates of individual nodes, so that closer monitoring can be achieved in anticipation of a possible flooding event. The inner loop is the peer-to-peer behavior of a set of nodes which can communicate with each other but have no external coordination and are therefore described as self-managing or autonomic (a word borrowed from the notion of the autonomic nervous system). Other possibilities include one node, such as the gateway, taking a coordinating role.

Obtaining the required behavior from a system of this complexity is a challenge. Our third distinctive is our methodology for designing the adaptive sampling behaviors in this hybrid Grid-Pervasive system. We worked closely with environmental scientists in order to understand the data prioritization—this is non-trivial, because the significance of the data reported by a node is a function of its location and varies depending on the conditions reported by other nodes locally and from other sites. Rather than adopting established energy-conservation protocols, the adaptive sampling behavior in our system is the result of an extensive co-design exercise with environmental experts.

In the next section we discuss our protocol and other research efforts on energy aware routing. The FloodNet project is outlined in Sect. III. We then describe the design methodology in Sect. IV and in Sect. V we present the energy aware, adaptive protocol that was derived using it, together with evaluation results and discussions on the protocol design. We provide an overview of the implementation of the sensor network and the Grid back-end in Sect. VI. We close by reflecting on the lessons learned.

II. Related Work

The primary feature that distinguishes our adaptive sampling and routing from other research is that the resulting protocol enables the reporting frequency of sensor nodes to be adaptive to both local conditions and the model requirement (i.e. the *data importance* imposed upon by the flood predictor model, see Sect. III). In this section, we describe and compare existing energy aware routing protocols with the FloodNet adaptive routing protocol (FAR). To the best of our knowledge, none of the related work described in this section looks at the effect of the diversity in the reporting rates of nodes on the protocol design.

Stojmenović and Lin [1] proposed power-aware localized routing which only requires localized routing information to minimize the energy consumption and extend battery's worse case lifetime. Due to the absence of mechanisms to provide location information for nodes, we are unable to directly apply their approach to FloodNet. Other energy aware routing protocols using the location information of nodes include GEAR [2] which achieves longer network lifetime by balancing energy usage across the network. GEAR uses energy aware and geographically informed neighbor selection heuristics to forward a data packet towards the target region and applies recursive geographic forwarding or restricted flooding to disseminate the data packet within the region.

To maximize the network lifetime, Chang and Tassiulas [3] proposed formulating the routing problem in wireless sensor networks consisting of static nodes as a linear programming problem for constant data rates as well as arbitrary processes. The major assumption in their work is that global network information is available for decision making on routing. This assumption is also shared by [4]. Hence, each origin node of a commodity can calculate the shortest cost path to its destination node. This is, however, not suitable for the FloodNet scenario in which each node only has a local knowledge of the global network.

LEACH [5], PEGASIS [6] and the Energy Aware Routing protocol [7] extend network lifetime by evenly distributing the energy load among all the nodes in the network. They assume that nodes have a fixed sampling and reporting rate. Moreover, LEACH and PEGASIS assume that nodes are able to transmit with enough battery power to reach the base station. Although there are superficial similarities between LEACH and FAR, and specifically LEACH also considers non-uniform energy consumption of nodes, e.g. cluster-heads often consume more energy for data forwarding, LEACH essentially achieves a longer network lifetime by evenly dissipating energy among all nodes whilst FAR does it by enabling nodes to consume energy according to need.

Schurgers et al. [8] proposed using topology management techniques to coordinate the sleep transitions of all nodes whilst ensuring adequate network connectivity. The sensor network was assumed to be in the monitoring state during most of its lifetime. Power savings were achieved at the cost of an increased path setup latency. This scenario also motivated [9], an energy-efficient MAC protocol that tries to reduce the waste of energy from collision, overhearing, control package overhead and idle listening. In contrast, sensor nodes in FloodNet are required to report data at different but more frequent rates and thus these methods are not appropriate for solving the FloodNet problem.

The Pulse protocol [10] was designed for multi-hop wireless infrastructure access to mobile users by utilizing a periodic flood initiated at the gateways to provide the routing and synchronization information to the network. Substantial energy savings can be achieved by using the synchronization information to allow idle nodes to power off their radios for a majority of the time when they are not required for packet forwarding. The Pulse flood proactively maintains a route from all nodes in the network to the infrastructure access node. This is in contrast with FAR that selects the routing path on-the-fly as data messages traverse the network and thus is robust to temporary failure.

III. Project Background

The FloodNet project has deployed a set of intelligent sensor nodes around a stretch of river in the east of England. This site was chosen for its tidal behavior so that, for test purposes, there are regular variations in water level. The nodes are powered by solar cells in conjunction with batteries and each node communicates with its neighbors using wireless Ethernet. A special node, the gateway, relays the data back to base using GPRS (General Packet Radio Service). This is an ad hoc network, with nodes relaying information across the network to ensure data delivery to the gateway. Various parties can subscribe to the incoming data stream. As well as being stored in a GIS (Geographical Information System), the data is used to inform flood simulations which are used to make flood predictions. The topology of the sensor network is depicted in Fig. 2. One can envisage a number of such deployments at different locations in the river, each reporting back in this way—the current deployment enables us to explore the issues of working with this spatial density of data.

The process of adaptive sampling is mediated by the use of a flood predictor model [11] that comprises a stochastic one-dimentional numerical hydraulic model (ISIS) coupled to an ensemble Kalman filter. The model allows for the real-time collection of water depth data to update the flood predictions regularly with refreshed data. When the model-based probability of the water level exceeding a threshold at a validation location is less than 5%, the requirement for transmission of data from the sensor node at the validation location is lowered. Otherwise, the requirement for transmission of data will arise. The degree of the model requirement is represented by means of the data importance. The predictor model is required to carry out extensive processing in a short period of time (currently 1 hour) to produce the data importance for sensor nodes during the following iteration. Upon each model iteration the network changes its behavior, altering the reporting rate (derived from the data importance) of each individual node according to the data importance placed upon it by the predictor model. In the broadest sense the nodes that first experience a flood event will have a high initial demand. This demand will ease as the model develops a sufficient level of confidence in the prediction for this particular node and as the flood event itself penetrates into surrounding areas, raising the data demand from these areas. With time the more disparate nodes become more active. As such a wave of activity passes from the sensor nodes in the main channel out to the floodplain areas.

IV. Use of Simulators

The fundamental purpose of the sensor network is to provide information to the appropriate quality demanded by the users. Hence to design the system, we focused on the requirements of the users of the information, and in particular with those familiar with the site. Through observing discussions with the users we noted that they were expressing a considerable degree of knowledge about the environmental situation of each node, about priorities and redundancy of data, and about desired behaviors in a range of different circumstances. It was evident that they sought a very sophisticated behavior from the system and brought considerable knowledge.

To capture this knowledge we adopted a design approach based on simulation. Two simulation tools were constructed.

A. High-Level Simulator

The first was a high-level tool which allowed us to express behaviors at a similar level to the statements that the users were making—essentially capturing rules. This enabled us to establish the essential features of the adaptive routing protocol (see Sect. V-A). Although we experimented with an expert-system approach (using JESS, the Java Expert System Shell), we found it useful to build a custom design tool in Scheme [12]. This was effectively an exercise in *metalinguistic abstraction*; i.e. we created a small language which was as close as possible to the ways in which we found the users were describing adaptive sampling behaviors (e.g. the number of reporting rates allowed and the desired interval between consecutive reporting activities).

The simulation approach was based on [13], with a small number of primitive operations to capture the essence of the sensor network and thereby raise the relevant issues. A node was defined to have a number of local state variables and to be able to communicate through a simple broadcast to neighbors within a specified radius and with a specified probability of success. The nodes were required to bootstrap themselves from an initial situation in which they had no knowledge of their neighbors. Behavior was expressed as a set of rules which are triggered on incoming messages according to local state. The basic inter-node messages were:

- HELO — a null message for anyone who is listening so that they know they are receiving from this node identifier;
- HOP — announces to neighbors the number of hops from this node to the gateway;

STATUS announces to neighbors the node status, with parameters including battery level and data importance;

LEVEL relayed message which carries sensor readings.

Some of the "intelligence" was encoded in a *route* function which describes the next node to which an incoming message is to be relayed based on a local routing table which is updated according to incoming messages. Constructing this simulator and performing experiments with it give insights into engineering a more comprehensive discrete-event simulator described in the following section.

B. Discrete-Event Simulator

A second simulator was used to conduct a full simulation of the deployed network. The simulator was implemented in Java and deployed on a single machine. Each sensor node was effected as an object and communicated with one another through message passing mechanisms. The simulator provided a message queue to host the incoming messages of all nodes to be processed and a simulator clock to capture the advance of simulation time at each node. All the messages were stored and processed in non-decreasing order of their timestamps.

The simulator implemented the basic adaptive routing protocol. In addition to a number of parameter choices in the routing protocol, we can also vary certain factors in simulation such as the number of sensor nodes, the pattern of power consumption of sensor nodes, and the frequency of reporting activities of sensor nodes. The simulator enabled us to explore various "what-if scenarios" and different deployment configurations. It also validated the algorithm ready for deployment.

V. THE ADAPTIVE ROUTING

Here we provide an overview of the FloodNet adaptive routing protocol (FAR) resulting from this design process.

A. The Protocol

The predictor model requirement (aka interest), is diffused through the network from the gateway using a flooding technique on an hourly basis (see Fig. 2). This process helps establish routing tables at individual nodes whilst capturing the distance information for these nodes. An interest message comprises the following fields: the unique identifier of the sender, the residual battery power of the sender, the message type, the message timestamp, the distance of the message recipient from the gateway, and the data importance for each node. The data importance will be converted locally by its associated sensor node to one of the three reporting rates (currently once, twice and three times per model iteration) allowed. If an incoming message comprises a fresh interest, then a node keeps a copy in cache. For messages regarding the same interest but from different senders, the node extracts the identifier, the residual battery power, the distance information and the data importance of the sender to fill in its routing table, replaces the sender identifier in the message with its own identifier and increases the distance in the message by 1. Only the first message of an interest will be further broadcast to others within the node's wireless transmission range. Otherwise, the message is dropped for preventing both duplicates and forwarding loops. This process is carried out recursively to ensure that each node will be notified of such an interest.

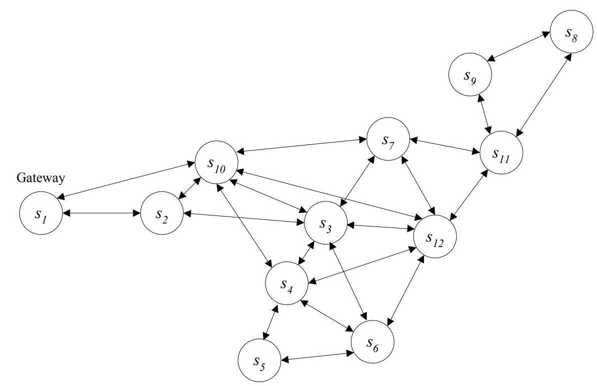

Fig. 2. Interest propagation in FloodNet

Sensor nodes maintain up-to-date information about neighbors in routing tables to assist data message forwarding. The neighbor status comprises the unique identifier, the distance from the gateway, the residual battery power and the data importance of a neighbor. The frequency on which a sensor node broadcasts its up-to-date information is a protocol design parameter that is determined by the accuracy requirement of the component that will utilize the information. As presented later the adaptive routing algorithm will take into account the neighbor status to decide the optimal path for routing, the frequency of local broadcast should therefore correspond with the reporting rate of nodes. Local broadcast is triggered prior to the start of any reporting activity. Sensor nodes receive updates of neighbor status and modify their routing table wherever necessary.

The adaptive routing algorithm we proposed is used for making a decision on which neighbor a sensor node should forward a data message to. We use a metric, *priority*, to denote the degree that a sensor node should be chosen to forward the data message. Ideally, the greater the priority of a node is, the more likely the node would be selected. The priority of sensor node s_1 with respect to s_0 has the following form:

$$p_0^1 = \frac{b_1}{e_{0,1} * \beta^{t_1}}$$

where b_1 represents the residual battery power of s_1, $e_{0,1}$ is the link cost, i.e. the energy required for transmission on $s_0 \leftrightarrow s_1$, t_1 denotes the data importance of s_1, and β (≥ 1) is a tunable parameter[2]. It is assumed that the link cost is known to the involved pair of sensor nodes.

In the cases where a few sensor nodes share the same priority the following rules should be applied:

[2]The discussion about the best value for β can be referred to in Sect. V-B.

- If sensor nodes have the same priority and different data importance, then s_0 shall choose the one with the least importance.
- If sensor nodes have the same priority and the same data importance, then s_0 shall choose the one with the least distance;
- If sensor nodes have the same priority, data importance and distance, then s_0 shall randomly choose one.

We present the details of the routing algorithm as follows.

1) A sensor node sends a data message to a neighbor with the highest priority among all neighbors closer to the destination.
2) If no such neighbors exist, the node sends the data message to a neighbor with the highest priority among all neighbors that are of the same distance from the destination.
3) If all neighbors are farther from the destination, the node sends the data message to a neighbor with the highest priority among all neighbors. Otherwise, the data message is dropped.
4) Step 1, 2 and 3 are repeated until the data message reaches the gateway, or is dropped out.

Each data message carries the following information: the sender identifier, the receiver identifier, the message type, the timestamp, the sensor readings and a list of visited nodes. Hence, forwarding loops are prevented as a sender will always check the list of visited nodes in the data message before retransmitting it.

The priority function has been carefully chosen to conserve the battery power of nodes with more important data. It produces a higher priority, which means more chances of being used as a router for data transmission, for sensor nodes with ample battery power and a light reporting task, whilst giving a lower priority to those with a lower level of battery power and a heavy reporting task. Delivery of data messages has no dependency on any specific node, as each sensor node maintains a routing table listing pointers to multiple neighbors that can further relay data back to the destination, and the optimal path is computed on demand. Hence, the adaptive routing protocol is robust to topological changes due to transient node and link failure.

B. Performance Evaluation

The FAR performance was evaluated through simulation, using the second simulator described in Sect. IV-B. The basic simulation parameters were chosen to model an ad hoc network that consists of 12 sensor nodes with 1 of them residing on the gateway. The initial battery power for all nodes is 1 unit. The exception to this is s_1 on the gateway that was assumed to have ample energy at all times. We assumed constant transmission power for sensor nodes. The energy consumption during idle time was not included in the design and the simulation as we assume a situation under which the energy consumption due to interest propagation, neighbor status maintenance and data message delivery is dominant. The battery consumptions of full power transmitting and receiving are 0.005 unit and 0.0005 unit. The data importance for each node is randomly distributed over [0, 1]. Each model iteration was simulated to last for 1 hour. Having different reporting rates, sensor nodes may generate and send data messages at the 20th, 30th, 40th and 60th minute of each model iteration which correspond to the 4 time units of one simulated hour.

The simulator implemented another protocol we refer to as the minimum energy consumption forwarding (MECF) as we anticipate to see the improvements that FAR brings to its performance by taking into account the residual battery power and the data importance of sensor nodes. The distinction between FAR and MECF is that MECF only forwards data messages to a neighbor that leads to a minimum energy consumption path, i.e. the priority in MECF is calculated on the basis of the neighbor distance and the link cost.

Moreover, the simulator implemented the Energy Aware Routing protocol (EAR) [7] that aimed to increase the survivability of network. The weighting factors used in the experiments on EAR are $\alpha = 1$ and $\beta = 50$, as used in [7]. To study the performance increase of FAR over EAR as a function of network density, we also generated sensor fields (30 units by 30 units) in which different number of nodes are randomly placed. The number of sensor nodes is ranging from 8 to 24 nodes in increments of 4 nodes and each node has a transmission range of 15 units.

We measured the *success ratio* which is defined as the ratio between the number of data messages successfully received by the gateway to the total number of data messages that should be generated by all sensor nodes in the field. We also measured the *network lifetime* which we define as the first time unit at which data messages cannot be generated by sensor nodes or get lost due to insufficient energy. If the battery level of a node is so low that it cannot transmit a data message at full power, then we consider the messages that the node holds but has not sent, are lost. The *node operational time* is defined as the time unit at which the energy of any of the sensor nodes is depleted. For exploring the robustness of FAR, link failure was simulated by a probability of failing links that occurs randomly in each model iteration. Link failure rate is defined as that probability. We executed 20 runs of the simulator for each of different protocols and of a number of different tunable weight β (in FAR) in the simulated ad hoc network[3].

The comparison between MECF and FAR with different β in the FloodNet topology is shown in Table I. It can be observed that FAR outperforms MECF in all cases by producing a longer network lifetime (NL) and a node operational time (NOT). We believe, by carefully choosing a neighbor to forward data messages, FAR enables the energy of sensor nodes to be consumed according to need whilst striking a balance between minimizing the overall energy consumption and maximizing the minimum residual battery power in sensor nodes. Hence, it extends NL and NOT.

[3]It should be noted that being situated on the gateway, s_1 is treated differently from others and all simulation results presented in this section do not include data of s_1.

TABLE I

COMPARISON BETWEEN MECF AND FAR WITH DIFFERENT β IN FLOODNET TOPOLOGY

	MECF	FAR(1.0)	FAR(1.0001)	FAR(1.001)	FAR(1.01)	FAR(1.1)	FAR(1.5)
NL	30.65	32.35	32.55	32	32.35	31.9	31.55
NOT	30.2	31.95	32.25	31.85	31.95	31.6	31.4

TABLE II

COMPARISON BETWEEN MECF AND FAR WITH DIFFERENT β IN RANDOM TOPOLOGIES

	MECF	FAR(1.0)	FAR(1.0005)	FAR(1.005)	FAR(1.05)	FAR(1.5)
NL	38.95	41.1	41.7	41.3	41.2	40.85
NOT	38.85	41.05	41.35	40.95	40.9	40.4

TABLE III

NETWORK LIFETIME AND NODE OPERATIONAL TIME FOR FAR (β = 1.0005) AND EAR

# nodes	12				24			
topology	FloodNet		random		FloodNet-like		random	
protocol	EAR	FAR	EAR	FAR	EAR	FAR	EAR	FAR
NL	29.75	32.55	39.2	41.7	15.85	16.2	26.65	30.3
NOT	29.45	32.25	38.85	41.35	15.8	16.2	26.05	29.8

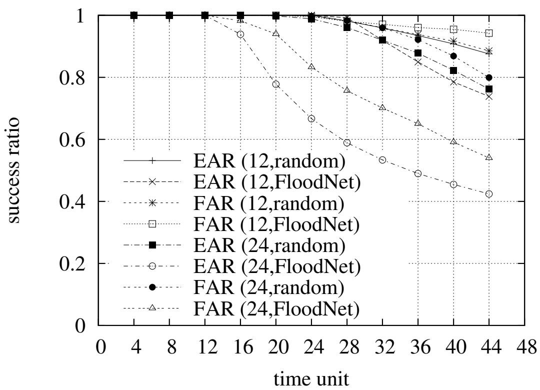

Fig. 3. Success ratio for FAR (β = 1.0005) and EAR

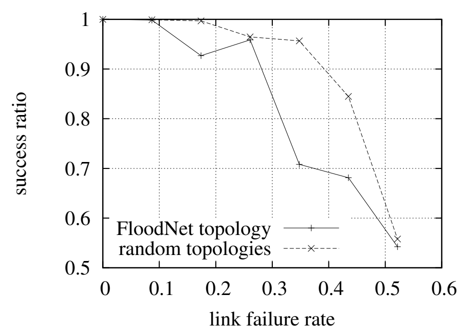

Fig. 4. Success ratio for FAR (β = 1.0005) with different link failure rates

Also, we carried out experiments on MECF and FAR in a number of random topologies ($\geq$ 20) and Table II reveals the result. Both MECF and FAR deliver a better performance in random topologies than in the FloodNet topology. FAR achieves its best performance in the FloodNet topology with β = 1.0001 and in the random topologies with β = 1.0005. This is because such small values for β can prevent the energy of some nodes being depleted far earlier than that of others. The best value for β, however, may need to be tuned to the particular example scenario.

Table III[4] presents the network lifetime and the node operational time that FAR (β = 1.0005) and EAR can achieve respectively. FAR outperforms EAR in all cases that differ in the number of nodes in the network and/or the network topology. In the FloodNet topology with 12 nodes, FAR extends NL by 9.4% and NOT by 9.5% over EAR. Both protocols deliver a longer NL and a longer NOT in random topologies than in the FloodNet topology. The reason for this is that in the random topologies the outgoing links are more evenly distributed among sensor nodes than in the FloodNet topology and consequently, the probability that the energy of the nodes closer to the gateway is depleted much earlier than that of others, is reduced. As the number of sensor nodes increases, FAR produces less increase in NL and NOT over EAR. The corresponding result on the success ratio is depicted in Fig. 3. FAR outperforms EAR by delivering a higher success ratio in all the topologies involved. Furthermore, as the number of the sensor nodes increases, both NL and NOT are reduced (see Table III), thus leading to a lower success ratio.

Figure 4 plots the success ratio as a function of the link failure rate in an ad hoc network of different topologies. FAR delivers a higher success ratio in random topologies than in the FloodNet topology. In the FloodNet topology, the success ratio is above 90% for the link failure rate up to 28%. In a small-sized ad hoc network like FloodNet, the functioning of certain links such as $s_0 \leftrightarrow s_2$ and $s_0 \leftrightarrow s_{10}$, is more crucial to a high success ratio than that of others, as no data messages can be successfully delivered to the gateway if they both fail.

[4]The FloodNet-like topology with 24 nodes corresponds to a combination of two 12-node FloodNet topologies as in Fig. 2.

TABLE IV

FAR (β = 1.0005) WITH DIFFERENT NUMBERS OF REPORTING RATES

topology	random			
# reporting rates	3		6	
protocol	EAR	FAR	EAR	FAR
NL	39.2	41.7	23.75	25.65
NOT	38.85	41.35	23.45	25.5

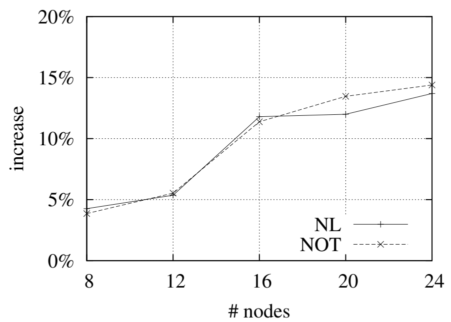

Fig. 5. The effect of network density on performance increase of FAR over EAR in random topologies

In random topologies, when 39% links fail, the success ratio remains above 90%. We believe the robustness of FAR is attributed to its routing algorithm as the latter always tries to find alternative forwarding neighbors even if there is a hole (i.e. no nearer neighbor exists) that could be caused by the failure of either links or nodes.

The impact of the number of reporting rates allowed on the performance of EAR and FAR is demonstrated in Table IV. As can be seen, NL as well as NOT is shortened as the number of reporting rates rises. The case of 3 reporting rates is the same as in the FloodNet scenario, whereas the case of 6 reporting rates denotes that any sensor node may report its readings from once, twice and three times, to six times per hour. Rather than allowing nodes to report at equivalent intervals, we found that scheduling as many sensor nodes as possible to report at the same time can help reduce the number of the local broadcast of neighbor status that should occur prior to the beginning of any reporting activity. For instance, our simulator allowed nodes to send data messages only at the 10th, 20th, 30th, 40th, 50th and 60th minutes of one simulated hour[5]. Therefore, a node with a reporting rate of four times per hour may send its data messages at the 10th, 20th, 40th and 50th minutes of one simulated hour. By doing so, the number of the local broadcast required is reduced, thus energy being conserved.

Figure 5 shows the effect of network density on the performance increase of FAR over EAR in random topologies. FAR achieves a little increase (around 4%) in both NL and NOT in a low density network (with 8 nodes). As the network density becomes higher, FAR will deliver a better performance. An increased network density means more neighbors per node on average. Hence, FAR will have more candidate nodes to choose from to determine a desirable routing path. By examining the simulation trace we found that the improved performance by FAR was obtained at the cost of more energy consumption. This is because the routing decision may have to involve a longer path in order to extend network lifetime whilst satisfying the predictor model requirement.

C. Discussion

A flooding technique was employed in FAR to propagate the predictor model requirement in the sensor network, which resembles interest diffusion in directed diffusion [14]. In directed diffusion, the sink is required to periodically refresh the interest to increase delivery reliability. We acknowledge that the successful delivery of interest messages in our scenario is very important and there are existing methods that guarantee the delivery of messages, for example [15], and might be suitable for FloodNet. However, as interest propagation is intended to find the distance information for each node, applying those techniques for guaranteed delivery may require non-trivial modifications to accommodate this requirement. We found through simulation that flooding is sufficient for interest propagation in FloodNet of its current scale and form. Definitely, we will need to investigate the issue of guaranteed delivery as FloodNet evolves. Note that due to the inherent nonscalability of the flooding technique and the routing table driven approach, the proposed solution may not be scalable enough for direct application to large scale sensor networks.

As a reactive routing protocol, FAR computes routes on demand and only needs to maintain routes to sensor nodes required. It was mentioned earlier that sensor nodes in the FloodNet scenario were deployed at a stretch of river, and thus (moving) obstacles in the signal path may lead to the loss of communication links. Also, some nodes may be temporarily inactive. We believe it is better to use a reactive protocol in FloodNet since proactive routing protocols require the knowledge of node activity to make decisions.

FAR relies on the concept of localized routing algorithms [16] in which sensor nodes only communicate with nodes within some neighborhood. Routing decisions are made on the basis of the priority derived from local conditions and model requirements. Achieving a desirable global objective is essential for the success of such algorithms. Unlike many other energy aware protocols, FAR takes into consideration the requirement that the reporting frequency of nodes should be adaptive to both local conditions and the predictor model requirement. Hence, the criteria for determining the best routing path should not depend upon the energy consumption on the path only[6]. This also explains why our definition for the network lifetime differs from that of others, for example [3] defined it as the time until the network partition occurs due to battery depletion whilst [4] modelled it as the earliest time that a message cannot be sent.

[5] In the real deployment of FloodNet, the intervals at which nodes send data messages should also be determined by taking into account the requirements specified by the environmental experts.

[6] A set of different power aware metrics to determine the optimal paths in mobile ad hoc networks were presented in [17].

Singh et al. [17] argued that it might be a better solution to route packets along nodes having sufficient energy, or nodes under light loads, to increase node and network life. This implies that an optimal path in terms of energy awareness is not necessarily the shortest-hop path. As mentioned in Sect. V-B we did observe that routing in FloodNet occasionally takes a longer route, which potentially increases the energy consumption. Essentially, our approach trades off the overall energy consumption for energy aware, adaptive behaviors of sensor nodes. This is why although the link cost between sensor nodes is considered in the priority function, it is not as significant as other factors, such as the residual energy, the data importance, and the distance, in contributing to the selection of the next-hop node for data delivery.

VI. Implementation

A. Sensor Nodes

The sensor nodes constitute a testbed for a variety of environmental monitoring applications and have relatively high computational capacity, in anticipation of this capability being available in smaller low cost nodes in the future. The nodes are based on a dual computer design. Each node consists of a Single Board Computer (SBC) based on the Intel®XScale PXA processor, which is used to run the local code and a "Sense" board that interfaces to the water depth sensor. The two computers are interconnected by the Inter-IC (I^2C) bus. The SBC is installed with ARM Linux (http://www.arm.linux.org.uk/) and it is the most power-intensive component of the node. For this reason, it is typically powered down to conserve the limited power resource.

It is the role of the Sense board to take periodic readings from the water depth sensor. These readings are stored in a local buffer on the PIC processor (http://www.microchip.com/). The Sense board also wakes up the SBC at regular intervals. As the Sense board is based on a PIC processor, it consumes significantly less power than the SBC and it remains in a powered up state. When the SBC is activated, it uses the I^2C bus interface to access the readings that the Sense board has collected and it will possibly transmit data back towards the gateway node.

B. Flood Modelling

Flood modelling is achieved using the ISIS software tools (HR Wallingford) for modelling rivers and their catchments. Due to the time constraints in the adaptive behavior regime, it is necessary to produce predictions in a guaranteed timeframe. It is not always possible to achieve this by running on shared grid resources. Hence we have implemented a dedicated cluster, which currently consists of 4 nodes. The cluster is used for the operational flood prediction, while additional grid resources can be brought to bear to explore what can be achieved with greater computational power. The dedicated cluster uses the Condor software (http://www.cs.wisc.edu/condor/), which is designed to provide high throughput computing on large collections of distributively owned computing resources, in combination with grid middleware from the UK Open Middleware Infrastructure Institute (http://www.omii.ac.uk). One of the nodes also supports the Storage Resource Broker (http://www.sdsc.edu/srb/index.php/Main_Page), a client-server middleware that provides a uniform interface for connecting to heterogeneous data resources over a network and accessing replicated data sets. The computations are coordinated from Matlab.

One question that naturally arises from this work is to what extend to the processing of the predictions can occur in the sensor network itself; i.e. to what extent the sensor network can itself be a Grid. We observe that the "outer loop" can also take into account information that is not known to the nodes, for example cloud cover forecasts which help make decisions relating to power availability.

VII. Conclusions and Future Work

Our design methodology has led to an interesting protocol. To be energy aware, many existing routing protocols focus on the residual battery power in nodes or the overall energy consumption of nodes along the path back to the destination in selecting the optimal path for data delivery. These approaches are viable for applications in which individual sensor nodes have an identical behavior (i.e. reporting frequency) and consume energy at more or less the same rate. The challenge that this project posed to us is that sensor nodes may differ in reporting rate, and therefore, it is very likely that the energy of some nodes is depleted much earlier than that of others in the network, thus leading to data message loss.

We addressed this issue by developing an adaptive routing protocol, FAR, for this and related projects. FAR takes into account the distinct behaviors of individual nodes and uses priority as well as a set of rules in determining the routing path. This metric evaluates the degree that a sensor node should be used to forward data messages based on the distance, the residual battery power, the link cost, and the data importance of the node. The rules give the guidance to FAR in situations where priority alone is not sufficient for making decisions. Hence, the protocol allows data messages to be carefully routed across nodes with ample energy and light reporting tasks while conserving energy for those which have a low level of energy and heavy reporting tasks. FAR is robust to temporary failure as the routing algorithm will always make an effort to find alternative forwarding neighbors even if there is a hole. Simulation results confirm the anticipated behaviors and improvements of FAR: it can extend the network lifetime and the node operational time over EAR in the investigated environments, and the performance increase of FAR over EAR grows as the network density becomes higher.

At the time of writing, a second phase of nodes is about to be deployed. The completion of our experiment will be when we have collected sufficient environmental data across varying conditions that we can confirm with the expert end-users that the adaptive sampling is performing as required. We will report this aspect of the work in a future paper.

Although nodes have well-defined local behavior (inner loop) and participate in the outer control loop, we have not explored node behaviors in which coordinated decisions are made across subsets of nodes which are isolated from the outer control loop; for example, through leader election. This is a topic that we are able to explore through the simulator.

We also need to conduct further work on isolation of nodes that have a disruptive behavior. One approach that we are able to explore through simulation is for a set of nodes to isolate a disruptive node by agreeing to communicate at different times.

Acknowledgement

FloodNet is funded by the UK Department of Trade and Industry under the Next Wave Programme, as part of the EnviSense Centre for Pervasive Computing in the Environment. The project partners are University of Southampton, IBM, Multiple Access Communications Ltd, Halcrow Ltd and ABP Marine Environmental Research Ltd. We are particularly grateful to Craig Hutton of the Geodata Institute and to Jeff Neal and Peter Atkinson of the School of Geography for specifiying the adaptive sampling requirements. We also acknowledge the input of the Equator project (EPSRC Grant GR/N15986/01).

References

[1] I. Stojmenović and X. Lin, "Power-aware localized routing in wireless networks," *IEEE Trans. Parall. Distrib. Syst.*, vol. 12, no. 10, pp. 1–12, Oct. 2001.

[2] Y. Yu, R. Govindan, and D. Estrin, "Geographical and Energy Aware Routing: a recursive data dissemination protocol for wireless sensor networks," UCLA Computer Science Department, Technical Report UCLA/CSD-TR-01-0023, May 2001.

[3] J.-H. Chang and L. Tassiulas, "Maximum lifetime routing in wireless sensor networks," *IEEE/ACM TRANSACTIONS ON NETWORKING*, vol. 12, no. 4, pp. 609–619, Aug. 2004.

[4] Q. Li, J. Aslam, and D. Rus, "Online power-aware routing in wireless ad-hoc networks," in *Proceedings of the 7th annual international conference on Mobile computing and networking*, Rome, Italy, 2001, pp. 97–107.

[5] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Hawaii International Conference on System Sciences*, vol. 8, Jan. 2000, pp. 3005–3014.

[6] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power-Efficient GAthering in Sensor Information Systems," in *IEEE Aerospace Conference Proceedings*, vol. 3, Mar. 2002, pp. 1125–1130.

[7] R. C. Shah and J. Rabaey, "Energy Aware Routing for Low Energy Ad Hoc Sensor Networks," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, vol. 1, Orlando, FL, Mar. 2002, pp. 350–355.

[8] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, "Topology management for sensor networks: exploiting latency and density," in *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking and computing*, Lausanne, Switzerland, 2002, pp. 135–145.

[9] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *Proceedings of IEEE INFOCOM 2002*, vol. 21, no. 1, 2002, pp. 1567–1576.

[10] B. Awerbuch, D. Holmer, and H. Rubens, "The pulse protocol: Energy efficient infrastructure access," in *Proceedings of IEEE INFOCOM 2004*, Hong Kong, China, Mar. 2004.

[11] J. Neal, P. Atkinson, and C. Hutton, "Real-time flood modelling using spatially distributed dynamic depth sensors," in *Geosensornetzwerke - von der Forschung zur praktischen Anwendung*, C. Brox, A. Krüger, and I. Simonis, Eds., vol. 23. Institut für Geoinformatik, Universität Münster, Germany: IfGI prints, 2005, pp. 147–159.

[12] H. Abelson, R. Dybvig, C. Haynes, G. Rozas, N. Adams IV, D. Friedman, E. Kohlbecker, G. Steele Jr., D. Bartley, R. Halstead, D. Oxley, G. Sussman, G. Brooks, C. Hanson, K. Pitman, and M. Wand, "Revised report on the algorithmic language Scheme," *Higher-Order and Symbolic Computation*, vol. 11, no. 1, pp. 7–105, Aug. 1998.

[13] S. Adams and D. De Roure, "A simulator for an amorphous computer," in *Proceedings of the 12th European Simulation Multiconference on Simulation - Past, Present and Future*, R. N. Zobel and D. P. F. Möller, Eds. Manchester, United Kingdom: SCS Europe, 1998, pp. 153–160.

[14] C. Intanagonwiwat, R. Govindan, and D. Estrin, "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, Massachusetts, USA, 2000, pp. 56–67.

[15] I. Stojmenović, "Geocasting with guaranteed delivery in sensor networks," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 29–37, Dec. 2004.

[16] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: scalable coordination in sensor networks," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, Seattle, Washington, USA, 1999, pp. 263–270.

[17] S. Singh, M. Woo, and C. S. Raghavendra, "Power-aware routing in mobile ad hoc networks," in *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, Dallas, Texas, United States, 1998, pp. 181–190.