CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE
Concurrency Computat.: Pract. Exper. 2008; **20**:1303–1316
Published online 22 November 2007 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/cpe.1279

An experiment with reflective middleware to support grid-based flood monitoring

Danny Hughes[1,*,†], Phil Greenwood[1], Gordon Blair[1], Geoff Coulson[1], Paul Grace[1], Florian Pappenberger[2], Paul Smith[2] and Keith Beven[2]

[1]*Computing Department, Infolab21, Lancaster University, Lancaster LA1 4WA, U.K.*
[2]*Environmental Science Department, Lancaster University, Lancaster LA1 4YQ, U.K.*

SUMMARY

Flooding is a growing problem, which affects more than 10% of the U.K. population. The cost of damage caused by flooding correlates closely with the warning time given before a flood event, making flood monitoring and prediction critical to minimizing the cost of flood damage. This paper describes a wireless sensor network (WSN) for flood warning, which is capable of not only integrating with remote fixed-network grids for computationally intensive flood modelling purposes but also performing on-site grid computation. This functionality is supported by the reflective and component-based GridKit middleware, which provides support for both WSN and grid application domains. Copyright © 2007 John Wiley & Sons, Ltd.

Received 4 July 2007; Accepted 21 September 2007

KEY WORDS: grid; WSN; middleware

1. INTRODUCTION

Flooding is a growing problem in the U.K., which affects a large number of people physically and economically. The problem was dramatically highlighted by the widespread floods of Autumn 2000, the total cost of which was estimated to be in the order of £1 billion. Following these floods,

*Correspondence to: Danny Hughes, Computing Department, Infolab21, Lancaster University, Lancaster, LA1 4WA, U.K.
†E-mail: danny@comp.lancs.ac.uk

major initiatives [1] have been undertaken to improve the U.K.'s flood readiness. These include improving flood defences, raising public awareness and, significantly for this project, improving flood warning systems.

Traditionally, hydrologists have approached flood prediction by deploying sensors (such as depth and flow-rate sensors) at sites prone to flooding. Data from these sensors are then collected via telemetry (e.g. GSM-based) or even recovered manually by field-workers. These data are then used as the input for flood-prediction models. Two main classes of flood-prediction model are commonly used. The first, referred to as *spatial* models [2], provides detailed, site-wide predictions, albeit with limited accuracy at any given point. These models are computationally complex and must be executed on cluster computers or grids. The second class, referred to as *point-prediction* models [3], provides accurate depth predictions, but only for a single point in the flood plain. They are computationally simple, so that may be executed in a timely fashion on standard desktop hardware or even on some embedded hardware.

Traditional flood-monitoring approaches impose a rigid separation between the on-site wireless sensor networks (WSNs) that are used to collect data and the off-site computational grid that is used to analyse these data. Essentially, the sensor networks are computationally 'dumb', being composed of nodes that are capable only of recording and transmitting sensor data. In order to better support timely flood warnings, we argue that more on-site 'intelligence' is required. The 'GridStix' sensor platform presented in this paper uses powerful embedded hardware, heterogeneous wireless networking technologies and next-generation grid middleware to implement an adaptable WSN that doubles as a lightweight grid, allowing nodes to not only ship data to remote fixed grids but also perform 'local' grid computations with significant benefits for adapting the behaviour of the WSN, supporting diverse sensors and providing more timely flood warnings.

The remainder of this paper is structured as follows. Section 2 discusses how local computation can be exploited in WSNs. Section 3 then introduces our 'GridStix' platform. Section 4 describes the important role of adaptation. Section 5 quantitatively evaluates the benefits of adaptation for this scenario. Section 6 discusses related work, and, finally, Section 7 offers conclusions and outlines our current deployment status and directions for future work.

2. EXPLOITING LOCAL COMPUTATION

Our prototype flood-prediction system uses on-site grid computation to improve support for flood monitoring. This section discusses how local computation can be used to (i) inform system adaptation, (ii) support richer sensor modalities and (iii) provide more timely warnings to local stakeholders.

2.1. Adapting WSN behaviour

Local computation can be used to drive the adaptation of WSN behaviour based on awareness of environmental conditions such as predicted flood risk and power monitoring. For example, based on the execution of point prediction models, we can switch to a more reliable network topology at times when node failure seems more likely (i.e. when imminent flooding is predicted). Adaptation

DOI: 10.1002/cpe

may also be informed by input from computationally intensive spatial prediction models executed in the remote fixed grid. For example, where spatial models predict unusual environmental conditions (e.g. when a bridge becomes blocked), redundant sensors in this region may be activated to increase the resolution of monitoring.

2.2. Supporting richer sensor modalities

The availability of local computation allows support for richer sensor modalities such as image-based flow prediction [4]. Image-based flow prediction is a novel technique for measuring water flow rates that uses off-the-shelf digital cameras. It is cheaper, more resilient and more convenient to deploy than the commonly used ultrasound flow sensors, but can only be used where significant computational power is available. Flow-rate measurements are calculated based on a series of images taken by a digital camera deployed overlooking the river. Naturally occurring tracer particles are identified on the water surface and tracked through subsequent images, from which the surface velocity of the water is calculated and cross-section velocity is inferred. The data set produced by this method, a sequence of high-resolution images, is too large for off-site transmission to be feasible using GSM or GPRS technologies; therefore, the method is impractical in current sensor network deployments. However, organizing computationally capable sensor nodes into a local grid allows analysis to be performed on-site, and the results of this analysis, which are significantly smaller, may then be transmitted off-site.

2.3. Providing more timely flood warnings

On-site flood modelling allows timely flood warnings to be distributed to local stakeholders. These flood warnings are based on the results of point prediction models executed by the local grid of GridStix nodes and disseminated to local stakeholders in a range of formats including on-site audio/visual warnings, a public Web site and SMS alerts. Each of these media has associated benefits and drawbacks. For example, SMS warnings are an effective method of publishing timely alerts to local stakeholders. However, SMS warnings require that users register for the service in advance and are therefore ineffective for stakeholders who might be unaware of a flood risk. Local audio/visual flood warnings may be effective without the need for stakeholders to proactively participate; however, their effectiveness is dependent upon stakeholders being within audio/visual range. In any case, increased on-site intelligence allows the system to continue providing flood warnings even if infrastructure linking the site back to the lab were to fail.

3. THE GRIDSTIX PLATFORM AND DEPLOYMENT

3.1. Overview

In order to achieve the 'local grid' functionality discussed in the previous section, a powerful and versatile sensor platform is required (in terms of both hardware and software). This section describes such a platform—GridStix.

3.2. Hardware platform

In order to support the proposed functionality, a sensor node must be capable of interfacing with a variety of sensors including traditional sensors such as depth sensors and more novel sensors such as the digital imaging hardware that is used to support image-based flow prediction. A suitable device must also be capable of supporting a variety of wireless communications technologies to provide redundancy and allow sensor nodes to switch physical network technologies as conditions require. Finally, the device must have sufficient computational and storage resources to support the GridKit software platform, which is described in detail in Section 3.3.

WSNs make use of devices with constrained local resources, such as the Berkley Motes [5]. Such devices have extremely modest power requirements and can therefore operate for long periods on small batteries. However, such constrained platforms do not offer sufficient computational power to support on-site flood prediction, nor do they offer sufficient support for diverse networking technologies and sensor modalities. For this reason, more powerful embedded hardware has been selected for use in the GridStix platform.

Each GridStix node is based on the *Gumstix* [6] embedded computing platform, so named as each device is roughly the same size as a pack of gum. Despite their small size, each of these devices is equipped with a 400 MHz Intel XScale PXA255 CPU, 64 MB of RAM and 16 MB of flash memory[‡]. These hardware resources support the execution of a standard Linux kernel and Java Virtual Machine, making them inherently compatible with the GridKit middleware. Furthermore, the PXA255, which performs comparably to a 266 MHz Pentium-class CPU (for non-floating point operations), is capable of executing a point prediction model multiple times per minute.

The Gumstix devices also provide a variety of hardware I/O mechanisms, enabling connection to a variety of sensors. For example, the camera is connected via a standard wired Ethernet connection on the NetCF [6] expansion board, while depth sensors are connected via the ADC of the RoboStix [6] expansion board. In this way, it is possible to connect several sensors to a single device. In terms of networking, each device is equipped with 802.11 b g^{-1} and Bluetooth network hardware, which is used to provide an *ad hoc* communications infrastructure. Furthermore, a small number of the devices are equipped with GPRS uplink and DVB satellite downlink for transmitting and receiving data from off-site. Of course, the above capabilities come at the expense of increased power consumption: While a Berkley Mica Mote consumes only 54 mW during active operation [5], our devices consume around 750 mW during active operation using Bluetooth networking and around 2400 mW during active operation using WiFi networking; thus, it would not be feasible to power them for long periods using batteries alone.

To address these power issues, solar arrays are employed. On the basis of the empirical measurements taken at the flood site, we have found that four 15CM^2, 2.5 W mono-crystalline solar panels combined with a 12 V 7AH battery are sufficient to continually power a GridStix even during the dark British winter months. Finally, to minimize the effects of harsh weather conditions, flood water, vandalism, uncooperative grazing animals, etc., we house the devices in durable, water-tight

[‡]Gumstix offering a 600 MHz PXA 270 CPU, 128 MB of RAM and 32 MB of flash is now available; however, this paper describes and evaluates GridStix based on the 400 MHz PXA255 CPU only.

Figure 1. A deployed GridStix node.

containers and enclose all external wiring in resilient piping. A first-generation GridStix node is shown in Figure 1 (current versions have a larger solar array and more resilient cable-housing).

3.3. The GridKit middleware platform

The GridKit middleware platform [7] provides the key functionality that is required to support distributed systems such as Grid, P2P and WSN. GridKit is based on the OpenCOM [8] component model and the various facets of system functionality are implemented as independent component frameworks. This component-based approach allows developers to build rich support for distributed systems or, conversely, to build stripped-down deployments suitable for execution on embedded hardware such as Gumstix. Importantly for this application, GridKit offers rich support for application-level overlay networks through the 'open overlays' component framework and for runtime reconfiguration, allowing system functionality to be adapted to best suit the demands of the environment.

DOI: 10.1002/cpe

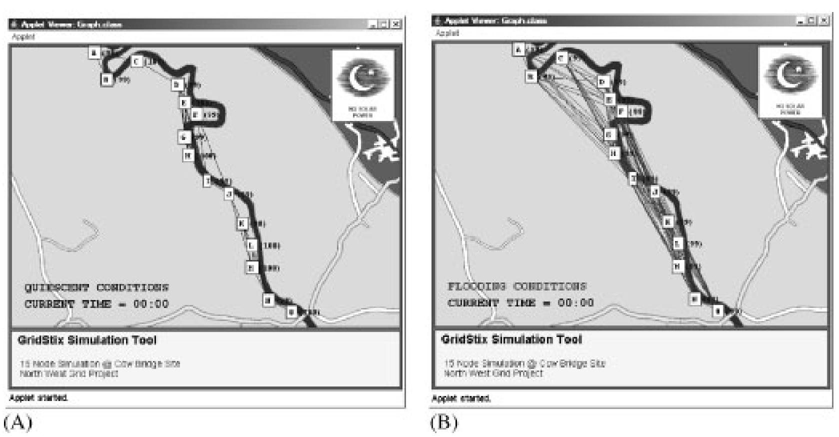
The overlays framework is an OpenCOM component framework that is deployed on each GridStix. The framework accepts 'plug-in' components that offer various types of overlay-related behaviour from underlying transport protocols such as TCP or UDP to spanning-tree overlays (for disseminating data from sensor nodes and the gateway), to application-specific overlays such as distributed hash tables, which can be used to support distributed storage. The full range of overlays implemented using the open overlays framework is described in [9]. Overlay plug-ins are themselves 'mini' component frameworks, each of which is composed of three distinct components that, respectively, encapsulate the following areas of behaviour:

(i) *Control*, in which the node co-operates with its peer control element on other nodes to build and maintain an overlay-specific virtual network topology;
(ii) *Forwarding* behaviour that determines how the overlay will route messages over the aforementioned virtual topology;
(iii) *State* information that is maintained for the overlay, e.g. nearest neighbours.

As these elements expose a standard interface, overlays may be freely composed. Furthermore, they may be re-configured at runtime to modify system functionality. A detailed description of how such reconfiguration is achieved is provided in [9].

3.4. Deployment

Fifteen GridStix nodes are currently deployed to perform flood monitoring on a 3 km stretch of the River Ribble in the Yorkshire Dales. All of these nodes are equipped with pressure-based depth sensors and a subset is equipped with ultrasound-based flow measurement equipment and cameras for image-based flow measurement. A single node is equipped with GPRS uplink and DVB satellite down-link. Figure 2 shows a satellite overview of the site (visualized in a purpose-built simulator) overlaid with Bluetooth and 802.11 b g^{-1} networks. Figure 2(A) shows a Bluetooth *ad hoc* network, while Figure 2(B) shows an 802.11 b g^{-1} *ad hoc* network.



(A) (B)

Figure 2. (A) Bluetooth physical network and (B) WiFi physical network.

DOI: 10.1002/cpe

This chosen site is prone to flooding for much of the year and thus offers good potential for evaluating the system under real-world conditions. Flooding at the site affects the nearby village of Long Preston and therefore additionally presents us with a motivation for evaluating warning systems for local stakeholders. Furthermore, the site is largely rural, which minimizes the risk to deployed hardware due to theft and/or vandalism. We anticipate expanding the deployment to a maximum of 20 nodes before the 2007 winter season.

4. SUPPORTING ADAPTATION

4.1. Overview

This section examines situations in which adaptation is possible and then goes on to consider the factors that can be used to inform such adaptation. Three discrete classes of adaptation are identified: (i) adaptation at the level of the physical network; (ii) adaptation at the overlay network level; and (iii) adaptation of the CPU performance.

4.2. Potential for physical network adaptation

Our flood-prediction system makes use of four wireless networking technologies, Bluetooth, IEEE 802.11 b g^{-1}, GPRS and DVB satellite downlink, each of which has very different performance characteristics.

4.2.1. On-site networking

- The 802.11 b g^{-1} hardware (Marvell 88W8385) supports speeds of up to 55 Mbps and supports a range of up to 1 km. It offers resilient and high-performance *ad hoc* networking, but consumes a significant amount of power—up to 2 W.
- The class 2 Bluetooth radio (Infineon PBA31308) supports speeds of up to 922 kbps at a range of up to 100 m. It offers QoS that is significantly lower than 802.11 b g^{-1}, but significantly higher than GPRS. It consumes a maximum of 0.4 W.

4.2.2. Flood site to lab networking

- The GPRS hardware (TelecomFM CellFax) supports uplink speeds of up to 29 kbps and downlink speeds of up to 58 kbps. Range is not an issue as we assume that the entire deployment site is within the bounds of GPRS coverage. GPRS offers lower performance than either 802.11 b g^{-1} or Bluetooth and consumes a maximum of 4.8 W.
- The DVB satellite router (Manhattan BSM 2) offers no uplink, but offers downlink speeds of up to 1 mbps and consumes a maximum of 6.4 W. Used together with the TelecomFM GPRS modem, this provides an asynchronous connection providing 1 Mbps downstream, 29 kbps upstream and consuming a maximum of 10.2 W.

DOI: 10.1002/cpe

Each of our communication technologies clearly has advantages and disadvantages. For example, 802.11 b g^{-1} offers good QoS and long range; however, it consumes significantly more power than Bluetooth. Similarly, the GPRS uplink can be augmented with DVB satellite downlink, which significantly improves downstream performance, but at the expense of significantly increased power consumption.

4.3. Potential for application-level overlay adaptation

As previously discussed, application-level overlays are employed to provide communications support for our flood-prediction system. There are a range of overlays that can be used, and each has advantages and disadvantages.

Off-site data dissemination is supported by the use of *spanning tree-based overlays*. These are commonly used in WSNs to disseminate data from a large number of sensors to a small number of logging or bridging nodes which form the 'root' of the tree. Prime examples of spanning trees are shortest path (SP) and fewest hop (FH) trees. FH trees are optimized to maintain a minimum of hops between each node and the root. They minimize the data loss that occurs due to node failure, but are sub-optimal with respect to power consumption. SP trees, on the other hand, are optimized to maintain a minimum distance in edge weights from each node to the root. As a result, they tend to consume less power than FH trees, but are more vulnerable to node failure. Both forms of tree can be efficiently created in a centralized environment using Dijkstra's algorithm [10], or in a decentralized environment using Bellman–Ford [11]. Examples of SP and FH spanning trees are shown in Figure 3.

SP and FH are just two common spanning tree types and many others are also available. We are currently investigating the performance of a range of spanning trees for off-site data dissemination. Nevertheless, these two examples serve to illustrate the trade-off that often exists between overlay performance and power consumption.

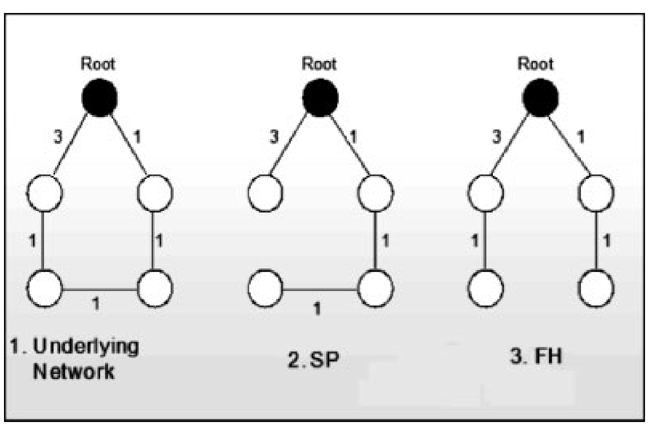


Figure 3. FH and SP spanning trees.

DOI: 10.1002/cpe

4.4. CPU power adaptation

The XScale PXA255 CPU used in the GridStix platform supports software-controlled frequency scaling, which allows the CPU to be set at a variety of speeds from 100 to 400 MHz. Processing power increases with clock speed, but at the cost of increased power consumption. Of course, power consumption is also affected by the way the CPU is used by applications; therefore, power management could also be implemented by explicitly controlling the manner in which local processes are scheduled (e.g. by modifying the frequency with which local point prediction models are executed). Nevertheless, the XScale's support for software control of clock frequency provides convenient, coarse-grained adaptation of CPU power consumption/performance.

5. EXPLOITING ADAPTATION

In the above, we have presented a number of ways in which the behaviour of GridStix nodes can be adapted. However, this is only half of the story—for adaptation to be useful and meaningful, it must be suitably informed by relevant real-world triggers. We now present three adaptation scenarios that demonstrate how awareness of real-world conditions can be used to maintain optimal system operation in changing environmental conditions. The first scenario considers adapting to increased flow rates, the second considers adapting to changing risk of node failure and the third considers adapting to changing prediction criticality. These scenarios are not exhaustive; however, they do demonstrate how local computation can be used to optimize WSN behaviour and thus produce a more useful and more robust flood prediction system.

5.1. Adapting to increased flow rates

As discussed in Section 2, the local grid functionality offered by the GridStix nodes is used to support distributed image processing. Distribution of the image analysis process is necessary in order to perform the fine-grained level of image analysis required to derive accurate flow rates. As the process requires the transmission of a large number of high-resolution images, it requires the high performance offered by 802.11 b g^{-1} networking. However, as described in Section 4.2, 802.11 b g^{-1} networking consumes significantly more power than Bluetooth networking. Fortunately, very coarse-grained changes in flow rate can be detected by a single node (with a local connection to the camera). When such a change is detected, the network reconfigures from a low-power Bluetooth physical network, which maximizes battery life, to a high-performing 802.11 b g^{-1} physical network (visualized in Figure 2(A) and (B), respectively). We measured the average per hop power consumption of each node deployed in the field as they relayed messages to the root (and from thence to the lab); in both cases, we used an SP spanning tree as described in Section 4.3. This is shown in Figure 4.

As can be seen in Figure 4, Bluetooth consumes significantly less power than 802.11 b for all nodes in the system, and thus, by maintaining a Bluetooth overlay during quiescent conditions and reconfiguring to use an 802.11 b overlay during high flow conditions, we minimize power consumption while maintaining support for sensor types that require high networking performance.

DOI: 10.1002/cpe

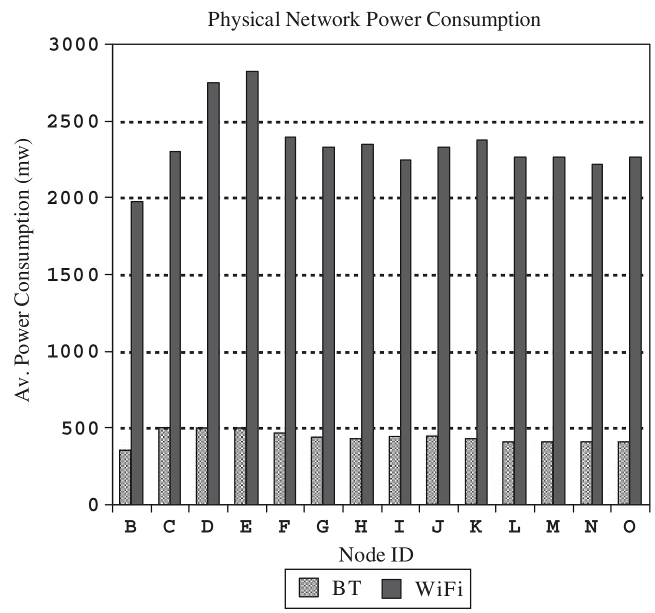


Figure 4. Power consumption of Bluetooth and WiFi physical network.

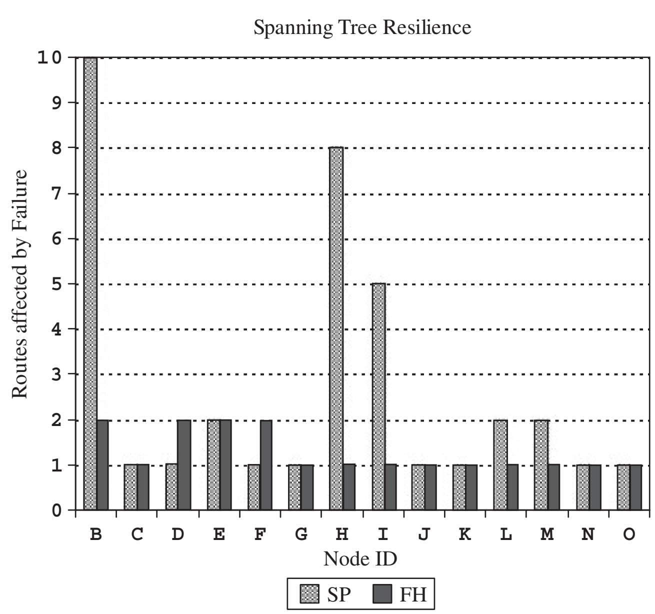


Figure 5. Resilience of fewest hop and shortest path spanning trees.

DOI: 10.1002/cpe

5.2. Adapting to changes in failure risk

Sensor nodes are at significant risk of damage or destruction due to being swept away by flood water or due to collision with debris. This risk may be assessed using on-site flow-rate measurements. The impact that a node's failure has on WSN as a whole is highly dependent on application level overlay that is being used to disseminate data offsite. Consider the spanning trees introduced in Section 3: SP trees consume less power than FH trees; therefore, during normal operation, off-site dissemination of sensor data should be performed using an SP spanning tree. However, when on-site flow measurements indicate an increased risk of node failure, the system should switch to an FH spanning tree, which is significantly more resilient to node failure. In this way power consumption is minimized during normal operation, while resilience is preserved at times of high risk. Figure 5 shows the resilience of FH and SP spanning trees in terms of the number of valid routes that are affected by node failure.

As can be seen in Figure 5, FH trees are significantly more resilient to failure than SP trees, though, as discussed in Section 4.3, they consume significantly more power. Thus, by maintaining an SP spanning tree overlay during quiescent conditions and reconfiguring to use an FH overlay during high-flow conditions, we minimize power consumption while maximizing resilience to failure in high-risk conditions.

5.3. Adapting to changes in prediction criticality

As previously described, local point predictions are used to provide timely warnings for local stakeholders. When such a warning is in place, the computation of local flood warnings becomes more time critical. Where initial flood warnings were accurate, local predictions can be used to show likely paths of inundation, and, in the case of erroneous warnings, local predictions can be used to lift the flood warning. The latter is particularly critical as flood preparation and particularly evacuation is an expensive activity. During normal system operation, the timely execution of flood warnings is not particularly critical; therefore, nodes can scale down their CPU speed to 200 MHz to minimize CPU power consumption (to 300 mW). However, when stakeholder flood warnings are in place and the computation of flood warnings becomes more critical, the nodes can increase their CPU speed to a maximum of 400 MHz (520 mW). In this way, the system conserves power during normal operation, while maintaining the ability to provide timely flood warnings in critical situations. The effect of CPU frequency scaling on power consumption is shown in Figure 6.

6. RELATED WORK

A number of grid-related projects have addressed the issues of WSN-grid integration in general and WSN-based flood prediction in particular. A prime example of the former is the Equator remote medical monitoring project [12], and a prime example of the latter is Floodnet [13]. However, all these systems (to the best of our knowledge) employ a 'dumb' proxy-based approach to integrating WSNs with the grid and thus cannot take advantage of the local computational power that we employ to drive the adaptation of WSN behaviour, to support richer sensor modalities such as image-based flow prediction and to provide timely flood warnings to local stakeholders.

DOI: 10.1002/cpe

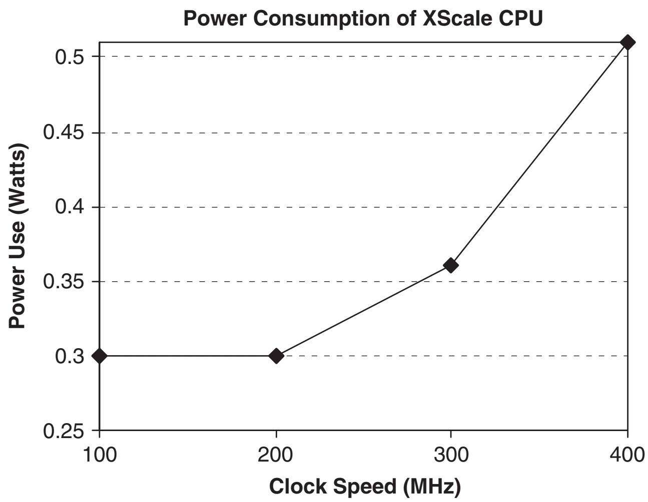


Figure 6. XScale power consumption.

The key difference between our system and existing work on WSN-grid integration is that our work aims to promote the sensors to *first-class* grid entities. This allows a greater degree of integration and flexibility than those approaches that treat sensor networks as conceptually distinct from the grid. In particular, for our flood-prediction scenario, it allows us to more effectively support WSN adaptation, to support richer sensor modalities and to enable proactive behaviour such as informing local stakeholders of pending flooding.

7. CURRENT STATUS AND FUTURE WORK

This paper has described a WSN that is capable of supporting not only remote off-site flood modelling based on grids in the fixed network but also local on-site flood modelling using a lightweight grid built on our GridStix platform. The current deployment on the river Ribble is nearing maturity and we intend to expand this (up to a maximum of 20 nodes) before the end of 2007.

In future research, we are especially planning to work on improving our system's adaptation mechanisms. Currently, our adaptation policies are manually implemented, but we plan in the future to investigate the extent to which nodes can 'learn' appropriate adaptation behaviour. For example, if nodes were capable of autonomously selecting appropriate power management strategies, it would significantly reduce the time to deployment for novel environmental monitoring applications. To accomplish this, sensor nodes could successively load different power management policies and, based on the relative success of these policies, select the most appropriate one for a given environmental monitoring scenario or set of environmental conditions.

Currently, the information used to inform system behaviour originates exclusively from within the system itself. However, external sources might also provide valuable information on which

DOI: 10.1002/cpe

adaptation could be based. For example, local weather predictions, particularly predicted hours of sunlight, could be used to better inform battery-life models (due to fluctuations in the power captured by solar panels).

A final area of planned future work is to investigate how our WSN's functionality may be expanded from an exclusively monitoring role to additionally encompassing flood-response support. For example, real-time on-site visualization of flood models would be useful for the emergency services who could use these data to inform the placement of sand bags and other flood defences. Similarly, the digital cameras deployed to perform image-based flow measurement could be switched to providing real-time remote imaging for flood responders. This adaptation of node roles not only necessitates modifications to local functionality but also imposes new requirements for the supporting physical and application-level networks.

While the Ribble deployment provides a realistic environment for evaluating system performance, it has a number of significant limitations, such as a limited scale of deployment, the unpredictability of flood events and the time required to perform tests. We are therefore concurrently pursuing lab-based and simulation-based testing of the system to gain more insight into its generic applicability. In particular, we are currently gathering empirical data from the Ribble deployment, and, based upon these basic performance characteristics, a simulator is being constructed that will allow the testing of various overlays and reconfiguration strategies using site-specific topographical information, past weather conditions and past flooding data. This will be used to prototype potential deployment technologies and investigate the performance of large-scale deployments that could not be easily tested in the real world.

REFERENCES

1. The Environment Agency. Lessons learned: Autumn 2000 floods. http://www.environmentagency.gov.uk/commondata/acrobat/126637 [16 October 2007].
2. Pappenberger F, Beven K, Hunter N, Bates P, Gouweleeuw B, Thielen J, De Roo A. Cascading model uncertainty from medium range weather forecasts (10 days) through a rainfall-runoff model to flood inundation predictions within the European Flood Forecasting System (EFFS). *Hydrology and Earth System Science* 2005; **9**(4):381–393.
3. Beven K, Romanowicz R, Pappenberger F, Young PC, Werner M. The uncertainty cascade in flood forecasting. *Proceedings of the ACTIF Meeting: Innovation, Advances and Implementation of Flood Forecasting Technology,* Tromsø, 2005; **33**: [CD-ROM]. Available at: http://www.actif-ec.net/conference2005/proceedings/PDF%20docs/Session_04_Uncertainty/Beven_Keith.pdf.
4. Creutin JD, Muste M, Bradley AA, Kim SC. River gauging using PIV techniques: A proof of concept on the Iowa river. *Journal of Hydrology* 2003; **277**:182–194.
5. Crossbow. Mica mote data sheet. www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA.pdf [16 October 2007].
6. Waysmall Computers. Gumstix embedded computing platform specifications. http://gumstix.com/spexboards.html [16 October 2007].
7. Coulson G, Grace P, Blair G, Porter B, Cai W, Duce D, Cooper C, Younas M, Sagar M, Li W. Open overlay support for the divergent grid. *Proceedings of the U.K. E-Science All Hands Meeting*, Nottingham, U.K., September 2005.
8. Coulson G, Blair G, Clark M, Parlavantzas N. The design of a highly configurable and reconfigurable middleware platform. *ACM Distributed Computing Journal* 2002; **15**(2):109–126.
9. Coulson G, Grace P, Blair GS, Mathy L, Duce D, Cooper C, Yeung W, Cai W. A component-based middleware framework for configurable and reconfigurable grid computing. *Concurrency and Computation*: *Practice and Experience* 2005; **18**(8):865–874. http://dx.doi.org/10.1002/cpe.981.
10. Dijkstra EW. A note on two problems in connection with graphs. *Numerische Mathematik* 1959; **1**:S269–S271.

11. Bellman R. On a routing problem. *Quarterly of Applied Mathematics* 1958; **16**(1):87–90.
12. Rodden T, Greenhalgh C, DeRoure D, Friday A. Extending GT to support remote medical monitoring. *Proceedings of U.K. E-Science All Hands Meeting*, Nottingham, U.K., September 2005.
13. DeRoure D. Improving flood warning times using pervasive and grid computing. Available at: http://envisense.org/floodnet/ingenia/ingenia.htm [16 October 2007].