Smart Bridges, Smart Tunnels: Transforming Wireless Sensor Networks from Research Prototypes into Robust Engineering Infrastructure[☆]

Frank Stajano[a], Neil Hoult[b,1], Ian Wassell[a], Peter Bennett[b], Campbell Middleton[b], Kenichi Soga[b]

[a] *University of Cambridge, Computer Laboratory, Cambridge, UK*
[b] *University of Cambridge, Department of Engineering, Cambridge, UK*

Abstract

We instrumented large civil engineering infrastructure items, such as bridges and tunnels, with sensors that monitor their operational performance and deterioration. In so doing we discovered that commercial offerings of Wireless Sensor Networks (WSNs) are still geared towards research prototypes and are currently not yet mature for deployment in practical scenarios.

We distill the experience gained during this three-year interdisciplinary project into specific advice for researchers and developers. We discuss problems and solutions in a variety of areas including sensor hardware, radio propagation, node deployment, system security and data visualization. We also point out the problems that are still open and that the community needs to address to enable widespread adoption of WSNs outside the research lab.

1. Introduction

Large civil engineering infrastructure items such as bridges, highways, tunnels and water pipes are expected to last for decades or even centuries. Over the course of their lifetimes, these structures deteriorate and require timely maintenance in order to prevent further degradation that might lead to accidents, the need for replacement or, in the worst case, collapse. Traditionally, early detection of such deterioration is achieved by visual inspection, either during routine maintenance visits or when a maintenance team is sent to the site to investigate a known or suspected problem. But such inspections are time-consuming and costly and therefore infrequent. An alternative is to equip infrastructure with sensors that are permanently wired up to report back to a central system; but this solution is not adopted very extensively because of the difficulty and cost of running data and power cables

[☆] Revision 66 of 2010-02-03 18:30:09 +0100 (Wed, 03 Feb 2010)

[1] Dr Hoult, at Cambridge while this research was carried out, is now at Queen's University, Department of Civil Engineering, Kingston, Ontario, Canada.

to each individual sensor in challenging environments such as a subway tunnel or a long suspension bridge.

The purpose of our research project, which at the time of writing has been running for almost three years, is to develop a system for continuous monitoring of such infrastructure using wireless sensor networks which, compared to wired systems, are easier and cheaper to deploy and also offer the opportunity for straightforward expansion. Very few other groups, to our knowledge, have deployed wireless sensor networks (WSNs) with the goal of using them for *long term* monitoring of civil engineering infrastructure; among those few are Feltrin et al. [5] and there the main parameter being measured is vibration. Their system uses the limited processing power of the nodes in an efficient manner and their network feeds back only the most critical data.

Although many papers have been written about WSNs, experience papers reporting on real-world deployments are a minority: they include at least Mainwaring et al. [19] who monitor seabirds' nesting environment and behaviour, Arora et al. [1] who deploy a perimeter control WSN and Werner-Allen et al. [23] who monitor an active volcano. Closer to our scenario are Krishnamurthy et al. [16] who monitor equipment for early signs of failure and especially Kim et al. [14] who monitor the structural health of the Golden Gate bridge. But our favourite is Barrenetxea et al. [2], which describes the authors' experience in deploying several environmental monitoring networks over glaciers in the Alps: they offer a wealth of valuable insights on how to prepare for deployment and how to extract maximum value from the exercise. We adopted their "hitchhiker's guide" structure of presenting our experience as advice to a reader who might wish to do something similar. Content-wise our papers are complementary, since we focus primarily on radio propagation and security issues which Barrenetxea et al. did not explore.

How nice it would be if we could just go out and buy a commercial off the shelf (COTS) WSN system and use it for monitoring our structures straight away. Unfortunately, though, the available commercial systems are typically only kits of building blocks and a non-trivial integration effort is required, together with the development of any missing parts, before arriving at a complete and usable monitoring solution.

This experience paper identifies some of the challenges and issues encountered when installing wireless sensor networks in the field, with specific but not exclusive reference to civil engineering deployments, and discusses how these challenges can be addressed. Inspired by the format of the instructive and well-presented paper by Barrenetxea et al. [2], we share our experience in the form of small independent vignettes, each accompanied by specific advice that tells you how to avoid falling into the same traps. These will hopefully be of use to both the application-oriented engineer working on a new deployment and the lab-based researcher developing an improved generation of WSN kit. Whilst we have made great strides in this area, much work still remains to be done and so we have also highlighted areas of ongoing and future research.

The structure of the rest of the paper is as follows. We first examine (section 2) the main problems of WSN deployment and we present the contributions we offer. We then introduce the testbeds where we trialled, and continue to trial, our WSN systems (section 3). In the main section of the paper (section 4) we

then look at the challenges of WSN deployment and the guidelines we distilled from our experience, with particular emphasis on installation planning, network optimization, radio propagation and security. As well as investigating issues of general applicability we also focus more specifically on the deployment of WSNs on civil infrastructure and on how to manage the data collected from such systems. Finally, in section 5, we draw our conclusions and highlight the areas that still need further work. References and comparisons to related work are found throughout the paper, including earlier in this section. After most "Principles" we also point out the original papers in which we discussed the relevant issues in greater detail.

2. The Problem and Our Contributions

Civil infrastructure in the UK ranges from masonry arch bridges constructed in Roman times[2] to Victorian[3] tunnels to modern structures that push the limits of materials and computer-based design. A similar situation exists around the world and all this infrastructure suffers from the common problem that it is deteriorating with time and in some cases is forced to carry increasing loads. Some sections of the London Underground, for example, are over 100 years old and the buildings on the surface above, as well as other underground structures around these tunnels, have changed drastically in that time. This means that not only are the tunnel linings potentially degraded due to deterioration but that the loads these tunnels are carrying are different than what the Victorian engineer envisaged. The cost associated with replacing any section of the Underground network both in terms of capital investment and associated user delays would be tremendous. Monitoring is used to get a better understanding of the problem, allowing for a more cost effective solution, and to monitor the performance after a repair has been effected. Although most bridge infrastructure tends to be newer, one only has to note the recent collapses in Minnesota, USA [8] and Québec, Canada [12] and the subsequent loss of life to see that structures under 50 years old are also in need of observation.

So, civil engineers need monitoring facilities. However, as we said, you can't simply buy a WSN and start monitoring what you want. In the course of our project we have been performing the required system integration effort, developing the missing hardware and software components (including new sensors) and interfacing them to COTS WSN nodes. We have been deploying networks of such nodes on bridges and underground tunnels. Our WSN systems have been collecting sensor data for nearly three years, providing useful insights into the value of continuous monitoring over an extended period as opposed to spot checks with human-driven inspection. We have thoroughly investigated a variety of issues connected with radio propagation and system security. We have developed tools to render and display the sensor data for the benefit of the people responsible for the structural health of the entity being monitored. Above all, we have interacted on an ongoing basis with bridge, highway and tunnel operators and have learnt from them what

[2] The Roman Empire ruled over Britannia between AD 43 and 410 and several bridges still survive from that era [21].

[3] Victorian era: 1837–1901.

matters, what doesn't and the relative value of the benefits that an ideal system should provide.

The use of WSNs for the monitoring of civil infrastructure is an increasingly popular research area but much of this research takes place in a laboratory setting and is primarily concerned with monitoring vibration [3]. Other wireless sensors have been developed, including ones for acoustic emission detection in concrete structures [6], but they have only been used in short-term test deployments. Several groups have installed networks on actual civil engineering structures, such as the Golden Gate Bridge [15] or the Geumdang bridge [18]; but, again, such monitoring has been relatively short-term and has focused on vibration monitoring. These deployments represent a significant contribution to research in this area: the Golden Gate deployment consisted of 64 nodes, which is more than we have currently tried; but the temporary nature of their installation meant that issues relating to robustness and security did not have to be addressed.

Our approach is interdisciplinary and application-driven and it is based on experience in the field rather than just computer simulations. The questions that our WSN systems must answer are chosen by the research interests of the civil engineering academics and validated by discussion with the infrastructure owners and operators. By applying the WSNs to practical problems we can analyse whether the COTS equipment is suitable or whether we are pushing its boundaries.

With this background, the scientific contributions of this "experience" paper fall into two main categories: firstly, the engineering solutions we developed to bring the system to a usable state; secondly, the first-hand knowledge and insights about further open problems that still need to be addressed.

There is clearly a wide gap between the WSN prototype that works in the lab and the one that works in the field, attached to real sensors, exposed to the inconsistencies and threats of a real environment, expected to stay up for months on end, required to be easy to deploy by a maintenance crew, robust to occasional failure of the hardware and software[4] and so forth. We hope that our real-world experience will help inspire and direct further research and development into dependable and usable wireless sensor networks.

3. Our Three Civil Infrastructure Testbeds

In order to gauge the effectiveness of COTS WSN hardware for use in civil infrastructure monitoring, the systems were deployed at three sites in the UK.

The first site, shown in figure 1(b), is in the north anchorage chambers of the Humber Bridge—a major suspension bridge, shown in figure 1(a), that crosses a river estuary in East Yorkshire. Each of the four underground anchorage chambers[5] has dehumidification units to ensure that the exposed unprotected steel strands of the main suspension cables of the bridge do not corrode whilst exposed to the air in the chamber. A 12 node WSN was installed in the north chambers to verify whether

[4] Especially considering the difficulty, inconvenience, delay and cost of sending someone down the tunnel shaft to reboot a misbehaving node.

[5] The bridge runs north-south and is supported by two main cables. There are therefore two north and two south chambers, each anchoring one end of a main cable.

(a) Humber Bridge

(b) North-west anchorage chamber

(c) Ferriby Road Bridge

(d) Jubilee Line tunnel

Figure 1: Our three deployment sites. (Cfr. section 3.)

these dehumidification units were operating correctly. The main challenge with this installation was radio propagation, because some of the nodes are not in the same chamber as the gateway but are located in an adjacent chamber, connected to the first chamber only by a 1 m wide by 2 m high by 15 m long corridor, lined with reinforced concrete. Thus the distance from the gateway to the farthest node is approximately 60 m with a considerable amount of reinforced concrete separating the two.

The second site, shown in figure 1(c), is a reinforced concrete bridge, known as the Ferriby Road Bridge, located several hundred metres to the north of the Humber Bridge. In this case a WSN was used to measure deterioration of the bridge. A principal visual inspection performed in 2002 noted that there were numerous cracks in the concrete on the underside of the bridge. The same inspection noted that some of the bridge's bearings were not perfectly vertical but slightly inclined. In both cases these problems may have existed for many years without any change or they may be getting progressively worse and will eventually require maintenance. Whilst a visual inspection is useful for detecting these problems, it is hard for the same inspector, let alone a different inspector, to judge the change over time. A 7 node WSN was installed, with three nodes measuring change in crack widths, three nodes measuring change in bearing inclination and a final node measuring temperature. This deployment required the development of sensor nodes for the measurement of both displacement and inclination. Radio transmission was also an issue at this site since, although the maximum distance from the gateway to a node was only approximately 35 m, the layout of the nodes in a single plane led to radio wave propagation issues. Security is a greater concern at this site because, unlike the other two deployments where getting close to the WSN nodes requires some degree of physical intrusion and trespassing, here the network is easily accessible to the public, who can drive on the road under the bridge. The nodes are actually installed 8 m above the ground so tampering with the hardware still requires some effort; but on the other hand it is easy for an attacker to get within radio range, for example by parking a vehicle near the bridge, and from there perform a variety of over-the-air attacks.

The third site, shown in figure 1(d), is a London Underground tunnel on the Jubilee Line. There we installed a network of 26 nodes to measure changes in displacement, inclination, temperature and relative humidity in the 180 m long stretch of concrete-lined tunnel. The radio wave propagation environment within a tunnel is entirely different from those of the other two deployments. One of the key areas of research was to measure this propagation environment and develop models to predict the received signal strength at a WSN node deployed in a specified position. These models can then be coupled with an optimization algorithm based on parameters such as transmission strength, battery life and degree of route redundancy between the sensor nodes and the network gateway node; this in order to allow WSN users to optimize the placement of relay nodes ahead of installing the system. The same sensors that were used on the reinforced concrete bridge were also used here—suggesting that, once developed and field-tested, sensors of this kind might constitute a useful addition to the COTS range for use in many civil engineering applications.

4. Principles for Successful WSN Deployment

If WSNs are to be used pervasively on civil infrastructure, they will be deployed by maintenance crews, not by academics or their research students: they must therefore be straightforward to install (section 4.1). Radio connectivity is not automatically guaranteed (section 4.2) and neither is the security of the resulting installation (section 4.3). Certain problems are typical of large civil engineering installations (section 4.4) but also have relevance elsewhere. Finally, data interpretation and presentation offer their own challenges (section 4.5). For each of these aspects we shall now offer specific recommendations—including, where applicable, pointers to papers in which we have described our experience and solutions in greater detail.

4.1. Installation Planning and Network Optimization

Principle 1. *Multi-dimensional optimization: you must choose a goal function.*

A few nodes can only be placed at specific locations, for example an inclinometer that monitors whether a specific concrete panel is moving over time. The location of a relative humidity or temperature sensor, on the other hand, is not as critical; and repeater nodes, as opposed to sensor nodes, can go essentially anywhere, so long as they enable communication—indeed, a priori nothing tells us whether we need any relay nodes, and if so how many. To navigate the space of possible solutions on where to put the nodes, we clearly need an idea of what solutions should be considered as better than others.

Perhaps we want to minimize the number of repeater nodes to be deployed. Maybe we prefer redundancy, to be sure that all the values of interest will continue to be monitored even in case of failure of a certain proportion of the nodes, or in case of failure of a certain proportion of the *links* between the nodes. Perhaps our goal is instead to ensure that mesh formation will take place as soon as possible, unlike what happened in our first deployments. Or to ensure, with a given battery capacity for each node, the longest possible uptime between battery replacement visits. At a higher level, we might imagine optimizing for "whole lifetime cost", taking into account the cost of the nodes, the cost of the installation (each possible relay position might contribute a different cost according to how hard it is to access it, how time-consuming it is to install the relay there, and with what kind of fasteners), the cost of maintenance (having many additional relay nodes may increase the installation cost but decrease the maintenance cost if spreading out the wireless traffic means that the batteries last longer and require less frequent visits for replacing them; or perhaps not, if more nodes mean more batteries to replace) and so forth.

It is obvious that we can only meaningfully discuss a program for the optimal placement of nodes after defining a goal function. It is less obvious what the goal function should be, especially until you've taken part in a deployment. It is also worth noting that some of the goal functions suggested above, while in theory desirable, require rather sophisticated modelling before they can be quantitatively used to decide in advance where nodes ought to be placed.

We have created a deployment tool (described in greater detail in Hirai and Soga [7]) that can determine the location of relay nodes in the network by optimizing

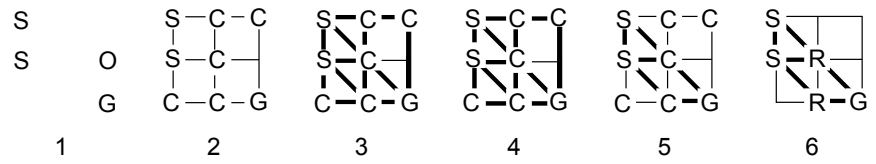

Figure 2: Simple deployment example with 9 nodes. S = sensor; C = relay candidate; G = gateway; R = relay (selected); O = obstacle; *thick line* = routing. (Cfr. principle 1.)

a goal function such as the initial deployment cost or the total communication distance. The approach considers, as would be the case in most civil engineering WSN deployments, that the locations of the sensors and the gateway are fixed—the sensor positions determined by what needs to be sensed and the gateway's by the availability of electrical power and of a connection to the outside. As illustrated in Figure 2 on a toy example, the method follows a six step process:

1. Initialize algorithm with given locations of the sensors, the gateway and the obstacles
2. Prepare relay candidates
3. Prepare routing candidates
4. Prepare routing pair candidates
5. Select the most preferable routing pair candidates
6. A solution of relay deployment can be specified as a result of the selection of routing pairs

The problem can either be solved using the GNU Linear Programming Kit (GLPK) to find the optimal solution or a local search (LS) based algorithm to find a quasi-optimal solution. The choice of solution technique is discussed in Principle 3 but the advantage of the method proposed here is that it creates two routing pairs that account for obstructions between nodes as well as optimizing for a given goal function. The use of two routing pairs means that there is a redundant path from any sensor to the gateway thus increasing the reliability of the WSN.

From our experience we believe that, if WSNs are to be competitive with respect to wired sensor networks, it is worth striving towards the goal of rapidity of deployment—hence the next principle.

Further reading: [7].

Principle 2. *Planning for shortest deployment time: ensure the multi-hop network will achieve end-to-end connectivity within a reasonable time.*

In the Humber Bridge anchorage room deployment, using the meshing software provided by the manufacturer, it took up to *one hour* for nodes to join the mesh. We had to put the nodes in place and then wait for an hour to determine if they could actually see one another; if they could not, we had to move one or more of

the nodes and repeat the process. This was quite time consuming and as a result the initial installation took two days, one extra than originally anticipated.

To overcome the mesh formation problem, a potential solution is to predetermine the mesh topology (even if sub-optimal) and preload it in the nodes. We have developed this approach for use with the Crossbow MICAz platform. Using broadcast messages, the nodes are programmed with preset routing tables. Once these tables are installed, the nodes cycle through each of the preset transmission routes, using each one for the same percentage of time in an effort to equalize battery use. This approach has currently only been tested in a laboratory setting but potentially offers the ability to have "instant" meshing in a real deployment. Preset routing has the additional advantage of hardening the network against certain types of attacks [22]. However it is also possible that the network will not be able to transmit data for long periods of time if one of the routes becomes unavailable due to radio connectivity or equipment failure. Thus one must carefully consider the trade-off between fast network setup, network stability and resistance to routing attacks on one hand, versus system availability on the other. A hybrid solution is possible whereby an initial configuration is predetermined and preloaded (giving fast network formation on deployment) but the optimal route is then reached by incremental improvements in a continuous feedback loop that continues even in steady state (removing one defence against routing attacks but improving availability).

Principle 3. *Assume access time to the site will be limited: you must plan in advance where to put the nodes.*

The London Underground site could only be accessed for a few hours per night while the Jubilee Line trains were not running. Deployment times exceeding those few hours required repeated visits to the tunnel, with escalating costs for personnel not based in London. In such situations, relay node placement by trial and error becomes too expensive as soon as the network grows beyond a few nodes.

Therefore, as mentioned earlier, we have developed a deployment tool that optimizes the relay node locations of the network based on goal functions such as initial deployment cost (as represented by the total number of nodes) or total communication distance (which represents power consumption as this tends to be dominated by radio transmission).

One important consideration for actual WSN deployments is the choice of optimization technique as the deployment tool may need to be used on site once the location of obstructions has been determined. Thus an optimization technique that runs quickly and minimizes the time required for deployment is preferred.

In our research, described more fully in [7], we compared the GLPK, which yields an exact solution, to an LS based method, which is approximate. As can be seen from the results in Table 1 on page 10, the GLPK finds the optimum solution more quickly than the LS based method. However, if only a quasi-optimal solution is required, the LS based approach is much faster than GLPK because the number of iterations can be reduced significantly. Although the difference in total computation time for this simple example does not seem significant, for larger problems, such as those seen in actual deployments, the computation time can easily become a significant percentage of the total time available for the deployment. For example,

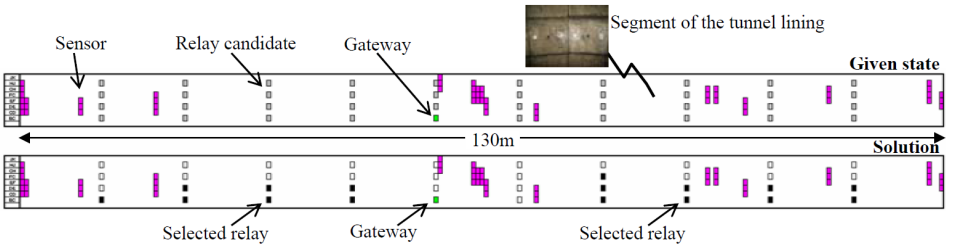

Figure 3: Map of the tunnel lining. (Cfr. principle 3.)

the London Underground deployment was modelled as illustrated in Figure 3, with all the potential relay node positions being considered. The first interesting result was that the optimum solution could not be found at all using the GLPK due to the memory limitations of the Java virtual machine. The second interesting result is based on Table 2 on page 11 which compares computation times for a given number of iterations using either the number of relays (NUM) or the total transmission distance (DIST) as the goal function. One can see that, if the number of relays is used as the goal function, there is no benefit to using a high number of iterations as the solution converges almost immediately to the optimal value of 16. Thus, if NUM is the goal function to be used, the relay node locations can be obtained quite quickly on site once the location of obstructions has been determined. Using the total transmission distance (DIST) as the goal function, on the other hand, required a significant number of iterations to reach the optimal solution. However it can also be seen that a significant reduction in computation time can be obtained if one is willing to accept a sub-optimal but reasonably close solution. We consider that our deployment tool, when used in approximate (LS) mode, provides a reasonable compromise between accuracy and response time and is therefore suitable for on-site usage.

Method	Iterations	Value (m)	Computation time (s)
GLPK	N/A	408.23	0.653
LS	100,000	408.23	0.919
LS	10,000	408.43	0.097
LS	1,000	408.58	0.012
LS	100	409.42	0.000
LS	10	409.83	0.000

Table 1: Numerical experiments with the simple tunnel. The value of the goal function is the minimum communication distance, averaged over 5 trials. Computation times refer to a 2.61 GHz PC with 3.43 GB RAM running Windows XP. (Cfr. principle 3.)

One of the necessary components of the deployment tool is a reliable path loss model. Our accurate empirical path loss models for tunnels on the London Underground (further details in section 4.2) are comprised of two parts. Firstly, using a regression line fit to the measured data, we have established the mean path loss with distance characteristic as a function of operating frequency, wall material, tunnel course and antenna position. Secondly, we have established fading probability

Goal	Iterations	Σ distance (m)	# relays	Computation time (s)
DIST	10,000,000	2030.4	27.8	3338.6
NUM	10,000,000	2038.0	16.0	4196.0
DIST	1,000,000	2030.6	27.6	359.4
NUM	1,000,000	2038.0	16.0	453.8
DIST	100,000	2031.2	29.4	33.9
NUM	100,000	2038.0	16.0	44.5
DIST	10,000	2032.5	31.8	2.9
NUM	10,000	2038.0	16.0	4.9
DIST	1,000	2034.9	32.2	0.0
NUM	1,000	2038.0	16.0	0.1
DIST	100	2037.1	24.4	0.0
NUM	100	2038.0	16.0	0.0
DIST	10	2037.9	17.0	0.0
NUM	10	2038.0	16.0	0.0

Table 2: Numerical experiments with London Underground tunnel example. The goal function alternates between minimizing the total distance or the number of relays. All results are averaged over 5 trials. Computation times refer to a 2.61 GHz PC with 3.43 GB RAM running Windows XP. (Cfr. principle 3.)

distributions as a function of these parameters that enable us to compute the additional path loss (known as fade margin) in excess of the mean path loss that is required in order to achieve a desired data packet error rate (PER) between nodes.

Knowledge of an accurate value of the path loss between nodes, combined with the minimum receiver sensitivity and antenna gain, enables us to establish minimum transmit power levels for each link. This, combined with knowledge of the node's DC input power as a function of the transmit power, will eventually permit the optimization process to address the minimization of total network power consumption.

Further reading: [7].

4.2. Radio Propagation

Principle 4. *Radio is like voodoo: it affects you even if you don't understand or believe it.*

Although one might suspect that a tunnel would provide the most challenging environment in terms of radio propagation, the London Underground deployment proved to be our most successful: every node achieved connectivity immediately after the network was commissioned. In contrast, as we mentioned, when the Humber Bridge anchorage deployment was first installed, only the nodes in the north-east chamber (i.e. the one with the gateway) connected to the network. Those that were in the north-west chamber, at the end of the $2 \times 1 \times 15$ m long corridor, connected to the network very occasionally and only when people were present in the anchorage. This problem was caused by the presence of deep signal fading on the link through the corridor connecting the two chambers.

The perceptive reader might ask: could we not have detected these fades beforehand? The answer is: not easily. One could, and we did, try taking readings with a spectrum analyzer. Whilst this would identify regions of fading, those regions would shift once the equipment and personnel were removed (hence the occasional transmissions from the north-west anchorage chamber in the presence of people).

One could also try modelling the radio propagation environment, as we shall discuss in greater detail for the tunnels, but the geometrical complexity of the space would make it almost impossible to pinpoint regions of fading.

So, how did we get the entire network working? Brute force. We placed high gain antennas on the gateway and on the node at the other end of the corridor. We also ensured that the antenna-to-wall offset was as large as possible since, if the antenna is placed too close to the wall, it can have a significant effect on transmission distance. Similar issues were encountered at the Ferriby Road Bridge and a similar solution was used.

The take-home point: use antennas with as high a gain as you can (beware: higher gain antennas are usually much larger) and keep the antennas as far away from the wall as possible.

Further reading: [26].

Principle 5. *Radio propagation modelling: to minimize the number of nodes to be deployed you need an accurate, efficient and robust propagation model.*

We mentioned the possibility of fading and its negative impact on network connectivity. We also expressed an interest in trying to predict where fades will occur.

Although the Humber Bridge anchorage chambers were too geometrically complex to model, there are other structures, such as tunnels, that due to their linear and relatively constant geometry lend themselves better to electromagnetic modelling. As part of our research we constructed a finite difference time domain (FDTD) model that allows for the path loss within a tunnel to be modelled. In most previous work, tunnels have been treated as large-scale waveguides. However such an approach gives inaccurate path loss predictions at close ranges and also for antennas located close to the tunnel wall [4]. Ray tracing [20] has also been used for tunnel path loss prediction, but it lacks the ability to cope with the often complex tunnel environment and as a consequence can exhibit poor accuracy. By directly solving Maxwell's equations in the time domain, the Finite-Difference Time-Domain (FDTD) method [27] is well suited to the study of the electromagnetic propagation in a complex environment. Our Modified FDTD technique provides a tunnel path loss prediction model having a unique combination of flexibility, accuracy and computational efficiency.

The model takes advantage of the linear nature of tunnels to reduce a computationally expensive 3-D problem down to a much more manageable 2-D problem. Although the model will not tell you precisely where fading will occur, it does give you valuable insights. Firstly it gives you a prediction of measured signal level as a function of distance. Secondly, it lets you determine the degree of fading such that you can introduce a sufficient margin of transmit power to reduce link loss/failure to an arbitrarily low probability. This information can then be fed into a planning

and deployment tool (see principles 1 and 3) allowing would-be WSN installers to increase their chances of obtaining full network connectivity.

Further reading: [24].

Principle 6. *Radio propagation measurements: you must calibrate your radio propagation model with physical measurements that can only be obtained on site.*

Having gone to the trouble of building a 2-D Finite-Difference Time-Domain model, how can you calibrate it and check if it is accurate? You need to obtain real path loss data on site, which we did at several different locations on the London Underground network.

Measurements were taken in an abandoned section of cast iron tunnel near Aldwych Station, as well as in the stretch of the Jubilee Line tunnel where we installed one of our WSNs. Readings were taken at various locations in the tunnel including transmission from centre to centre (indicated as CC in Table 3) as well as for other cases such as side to centre or side to side (indicated as SS in the table). Interestingly, many researchers focus only on centre-to-centre measurements and modelling, despite the fact that the centre of a tunnel is potentially the worst location for a sensor if longevity is a parameter of concern. The high level conclusions of this study are summarized in Table 3.

Investigated Factors	**General Path Loss Performance**		
1. Antenna Position	Centre to Centre (CC)	$>$	All other cases (SS)
2a. Operating Frequency (CC)	868 MHz	$>$	2.45 GHz
2b. Operating Frequency (SS)	868 MHz	$\approx$	2.45 GHz
3. Material	Cast Iron (CI)	$>$	Concrete (Con)
4. Course	Straight (S)	$\approx$	Curved (C)

Table 3: Path loss performance as a function of various factors. In this table, $a > b$ is a shorthand notation to indicate that case a has a better path loss performance than case b (even though this technically corresponds to a numerically *smaller* path loss figure for a), while $a \approx b$ indicates that the two cases have similar performance. (Cfr. principle 6.)

One can see from Table 3 that a straight cast iron tunnel with the nodes placed in the centre and transmitting at 868 MHz would be the ideal case. But obviously the constraints of the real world, such as the passage of trains, may impose practical limitations.

The 2-D FDTD model was then calibrated and validated with this data and the results can be seen in Figure 4 on page 14. It can be seen that there is excellent agreement between the predictions and the measurements for both the centre to centre and side to side transmission cases. This lends further credence to the use of this model as a part of a deployment tool.

Further reading: [25].

Principle 7. *Once a node's position is fixed and you experience fading, you must be able to overcome it.*

So you modelled everything as accurately as possible before deployment, you made sure you had big (high gain) antennas and that they were as far away from

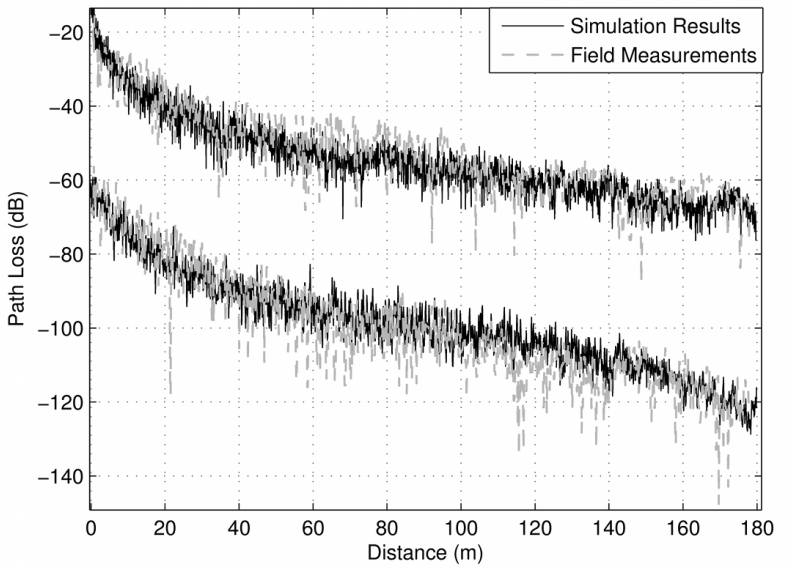

Figure 4: Comparison between the Modified 2D FDTD simulation results and the corresponding field measurements in Aldwych Tunnel. From top down: CC Case (with +20dB offset); SS Case (with -20dB offset). (Cfr. principle 6.)

the wall as possible, then you stuck your sensors to the structure, and yet—since no model, however accurate, can ever *exactly* replicate reality in all respects—you still find a few nodes that will not connect. There are two more potential strings to your bow: spatial and frequency diversity.

Spatial diversity fights fading through the use of multiple antennas separated in space. One feature of the 2.4 GHz transmission band is that the distance between locations experiencing fading tends to be fairly small and so an antenna separation of even 100 mm may be enough to overcome the problem. The potential pitfall with this technique is that you need a switch to allow you to use one antenna or the other, which adds cost and complexity.

The location of a fade is also dependent on the transmission frequency: consequently a node experiencing fading on one channel may work fine on a different one. This is illustrated in Figure 5 that shows path loss versus frequency and distance for a MICAz mote. Since the readings were taken in a static tunnel environment it is also the case that, at any particular distance, the path loss at each channel remains virtually constant in time. Thus, if a mote is installed in an area of poor or inadequate signal power, the potential for poor link performance will persist. Also note that the poor (or good for that matter) channels vary with mote separation distance. Consequently, it can be seen that one way to improve link performance is for the motes to be capable of using different channels, i.e., to be frequency agile.

Thus a possible solution to the fading problem would be to have nodes that hop through the transmission channels, allowing data to be transmitted at least some of the time. Alternatively, the nodes could determine the best channels to use to

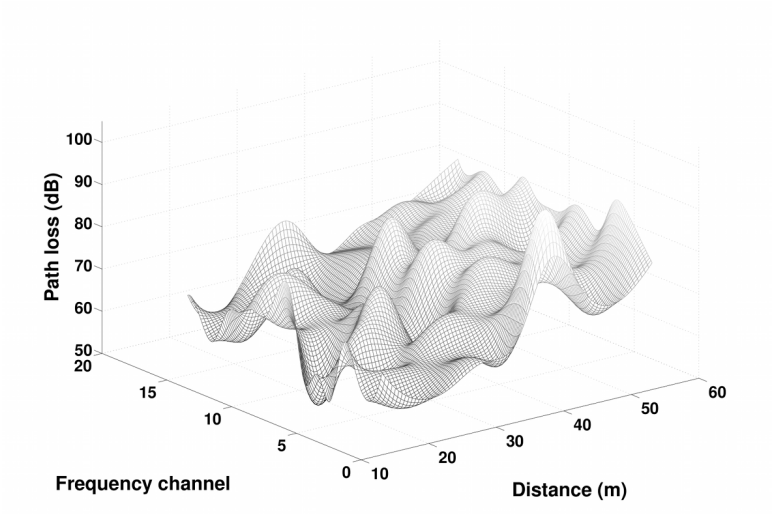

Figure 5: Path loss on sixteen 802.15.4 frequency channels at 30 m. (Cfr. principle 7.)

communicate between each other, but this would increase the complexity of routing. Further reading: [17].

4.3. Security

Principle 8. *Risk assessment: you must talk to the stakeholders and find out what they want to protect.*

It is unfortunately common for technical people to engage in elaborate designs for, say, encryption of storage and communications for all nodes, before even knowing which assets the system owners consider valuable. Don't do that. Ask the system owners what assets (whether tangible or intangible) are important to them and what threats they want to be secure against—even if some will argue that the users' own perception of risk may not always match reality.

We interviewed two senior representatives, one from the Humber Bridge Board and another from the London Underground. Their answers were informative and sometimes unexpected, for example when the bridge manager told us that he saw absolutely no reason to keep the sensor data confidential.

We do recommend debriefing the people in charge of the system on what they believe should be protected, rather than relying only on the techies' guesswork[6]. Having done that, however, it is also valuable to examine the system through the eyes of an attacker, in order not to miss vulnerabilities that the system owners may not have imagined.

We originally expected that the stakeholders would want to keep health monitoring data confidential in order to avoid bad press in case of problems. Interestingly,

[6] Which may be inspired by their love for elaborate solutions that may or may not address a real problem.

this was not the case at all for the bridge manager, who didn't think the readings would ever be embarrassing. The tunnel manager was more cautious and wanted to ensure he knew of any potential safety problem, and had a chance of addressing it, before news got out to the public. To some extent this different attitude may be ascribed to the fact that the bridge is in the open and in plain view, whereas the state of the tunnel is difficult to inspect, both for members of the public and for the operators themselves.

Further reading: [22].

Principle 9. *As far as sensor data is concerned, you should probably pay more attention to integrity and availability than to confidentiality.*

Both the bridge and tunnel operators clearly stated that the whole point of installing the wireless sensor network would be defeated if the data reported by the sensors could be tampered with. Ensuring that the sensor readings were genuine was much more important than preventing outsiders from being able to overhear what the sensors said.

Violating the integrity of sensor data might cause false positives (a report of a problem where there is none) and false negatives (a report that all is fine, hiding an actual problem). Which of these could be exploited by malicious attackers to cause damage to the structure? Causing false negatives allows the deterioration to progress, to the point where pieces of the structure fall off; but this is hard for an attacker to exploit because they'd have to depend on deterioration occurring naturally—they wouldn't be able to cause it to happen on demand. The ability to cause false positives, on the other hand, does not endanger the structure per se but forces the operator to waste resources in following up the alerts; eventually, it may cause the operator to ignore the WSN altogether, therefore effectively disabling it.

While it is reassuring to find out that integrity violations cannot easily result in physical damage to the structure being monitored, if the readings of the WSN are considered of any use and if denial of service through false alarms is to be prevented, then the end-to-end integrity of the readings must be ensured. This justifies employing cryptography in such scenarios. The requirements for battery life impose the use of symmetric cryptography (message authentication codes, or MACs) rather than the more computationally expensive digital signatures. The key distribution problem for WSNs is complex but has been extensively studied in the literature. The main trade-off is between the simplicity of a single shared key for the whole network (where the physical compromise of one node allows the attacker to forge authentication codes for all the other nodes too) and the more complicated approach of a different key for each node (where the physical compromise of one node affects only that node but which is incompatible with schemes where sensor data aggregation and decimation takes place in intermediate nodes before reaching the gateway unless even more complicated schemes are adopted). The single-key approach, supported by the open source cryptographic library TinySec (see footnote 9) was deemed sufficient for this application given the risk assessment with the stakeholders, which highlighted some concern for over-the-air attacks but a low likelihood of physical node compromise.

While cryptographic safeguards will ensure that integrity violations are detected, other approaches must be taken to ensure the availability of correct readings even in the presence of such violations. In both the anchorage and tunnel deployments we achieved enhanced availability of the readings through the use of redundant sensors and monitoring systems. In the anchorage, for example, the use of multiple temperature and humidity sensors in each chamber ensured that, if one sensor failed or was compromised, the data from the other sensors could be interpolated to recover an approximation to the value of that sensor. If the entire wireless network was compromised, the results of the WSN could be checked against the existing wired system. (The wired system was less convenient as it could only be accessed in the anchorage itself, but by the same token it was harder to compromise; it was therefore suitable as a redundant backup.). For the tunnel deployment three separate sensor systems (a WSN, a wired strain gauge system and a fiber optic strain sensor system), each using a different sensor type and data logger, were installed. The fact that the transducers were of different types and that they were installed in slightly different locations meant that care had to be taken when comparing and integrating the results from each system; however the use of three independent sensor subsystems yielded a highly resilient monitoring solution.

Further reading: [22].

Principle 10. *Your risk rating must be a function of the use to which the network will be put and of its side effects on the environment.*

One of the two main questions of our security investigation[7] was whether it was possible to cause damage to the structure (as opposed to the monitoring functionality) by tampering with the WSN, especially through over-the-air attacks rather than physical ones.

It soon became clear that this essentially depended on whether the WSN was used only for monitoring or whether it was part of a full feedback loop that also included actuators. For the bridge and tunnel scenarios the intended use of the WSN was indeed only monitoring, but for other situations, such as water distribution which some of our project colleagues separately investigated, where a leak in a mains pipe could flood a neighbourhood in a few hours[8], it is reasonable to imagine that leak detection sensors would be directly connected to valves, following a "turn off the water supply first, investigate the causes later" philosophy. In such circumstances, maliciously altering the appropriate bits of a packet would have direct consequences on the physical world.

Executive summary: check whether the WSN controls any actuators, directly (high risk) or indirectly (medium risk) or not at all (low risk).

Further reading: [22].

[7] The other was whether an expert of the application domain, in this case a bridge or tunnel engineer, who adopted a COTS WSN system and turned on all the security features it provided, would end up with a secure system. For this, see principle 12.

[8] Not to mention the loss of precious drinking water.

Principle 11. *You must assess whether a vulnerability of the wireless sensor network can be used to attack other networks connected to it.*

Another subtle and hidden way in which the addition of a WSN to a structure might introduce additional vulnerabilities is if by penetrating the WSN the attacker becomes able to infiltrate the back-end where the data is stored, analyzed and rendered and, depending on the quality of the internal isolation between subsystems, the rest of the computer system for that site. Given the number of security holes we discovered in the commercial system we examined, this possibility should not be taken lightly.

We found a crucial flaw, for example, in the XServe middleware component supplied with the Crossbow MICAz motes. XServe is a piece of software that connects the WSN to the back-end server and can be run on either the gateway or the back-end machine itself. We discovered that, when XServe runs as a web server, we could exploit a bug in one of its scripts and execute arbitrary commands on the computer where the program runs. As we wrote to the manufacturers before publishing this vulnerability, the exposure results from a simple programming error but the underlying cause is that the software is too feature-rich, which means it offers a very wide attack surface.

When evaluating the security of a WSN you will be wise to assume that the components of the WSN will contain some vulnerabilities (see principle 12); you must then assess how the WSN talks back to your main back-end system, and how that back-end system is connected to the rest of your IT system.

Assume the WSN is not trustworthy and treat the connection to it as you would any external unsanitized input.

Further reading: [22].

Principle 12. *Evaluation: you must perform independent penetration testing of the COTS equipment you use, even if it claims to offer a "secure" mode of operation.*

We performed penetration testing on the equipment that our team had chosen and we discovered a variety of attacks which we demonstrated on actual hardware (commercially bought nodes and gateway). Some of our attacks were fairly elaborate, involving selective jamming of packets, counter overflows and routing table manipulation; however the most devastating were the simplest. Here are brief sketches of three examples: a relatively complex one based on selective jamming and two of the simple but devastating ones. See Stajano et al. [22] for details.

Example 1. Whilst brute-force physical jamming of radio signals using high power transmitters is impossible to prevent, we demonstrated that a much subtler *selective jamming* attack can be carried out with no other equipment than an ordinary mote, albeit suitably reprogrammed. We developed a technique whereby the attacking mote can jam individual packets based on sender, receiver or other fields in the header, without disturbing the packets that don't match the pattern. The key to this attack was to rewrite the low-level radio access firmware so as to enable the attacking mote to sense the beginning of each packet and then very quickly switch from receive mode to transmit mode if that packet had to be jammed (this is not ordinarily possible because the radio only issues an interrupt after receiving

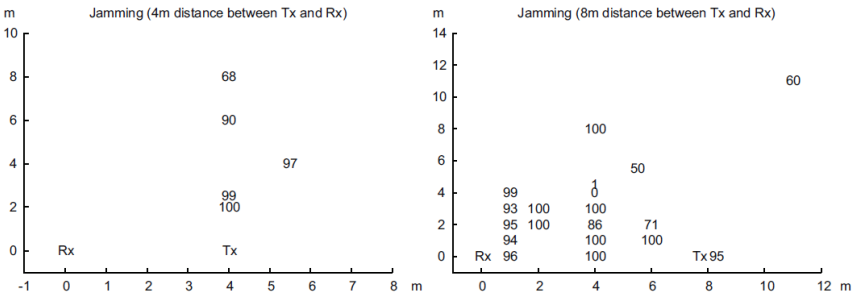

Figure 6: Success rate of selective jamming depending on the position of the attacking mote with respect to the victim transmitter and receiver. The Tx and Rx labels are the transmitting and receiving motes. The numbers 0–100 in the graphs denote the percentage of packets that were jammed when the attacking mote was in that 2D position relative to Tx and Rx. The unexpected patterns are attributed to fading. (Cfr. principle 12.)

the entire packet). This technique could be used to make an infrastructure owner believe that a specific mote had failed; to inject modified packets in the system as if they had come from the victim node; and, more subtly, to manipulate the routing tables of the nodes, thereby opening the door to more complex attacks including the establishment of routing loops to cause battery exhaustion. Figure 6 illustrates the effectiveness of these jamming attacks for a variety of jammer node locations. The experiment demonstrated that relatively sophisticated selective jamming attacks are possible using nothing more than an ordinary mote.

Example 2. The Stargate gateway, a ROMmed-up Linux box, had old and unpatched system software: though released in 2007, it was still vulnerable to exploits that had been publicly reported (and fixed) back in 2002. Such exploits allow an attacker to obtain a root account on the gateway, resulting in either a complete compromise of the sensor network or a complete denial of service.

Example 3. An optional feature of the nodes is over-the-air programming. If this is active, you can reprogram a whole network of motes by uploading a new code image into them. Amazingly, this facility is not authenticated, meaning that any attacker who can send traffic to the network is in a position to cause the nodes to execute any code they like.

Furthermore, even though the inter-node communications were in theory "secure" thanks to the presence of an open-source link layer encryption library called TinySec, in practice that library wouldn't even compile for the nodes we bought, which were equipped with a newer radio chipset. (We did, as part of our security work, port TinySec to the new chipset and distributed our changes as open source[9].)

We recommend that, as part of the necessary system integration effort, users perform thorough penetration testing on the WSN hardware and software components they adopt, or have someone competent and trustworthy do that for them.

Further reading: [22].

[9] Our port of TinySec to the MICAz hardware is at `http://www.winesinfrastructure.org/`.

4.4. Deployments in Civil Engineering Environments

Principle 13. *Deployment in harsh environments: you must ensure your sensors keep working and don't fall off.*

The original vision of WSNs for civil infrastructure monitoring of being able to stick sensors on with a dab of glue or even scatter them to the wind (like "Smart Dust" [13]) is not possible today and may never be possible. The reality is that the sensors must be packaged so that they will survive for years unattended.

For example, one might imagine that a tunnel is a relatively clean and protected environment. However, after only a few weeks in the tunnel, our boxes became coated in a thick black layer of brake dust from the trains. Thus they needed to be placed in boxes with ingress protection ratings that ensure that this brake dust, as well as other contaminants, will not come in contact with the electronics inside.

Also, given the currently available technology, the size of the batteries alone makes the units too heavy and bulky to safely stick on with glue and dangerous to scatter to the wind. In the anchorage deployment we attempted to attach all the wall mounted sensors (some sensors were hung on strings) using glue. Glue was chosen because the sensors could be removed without damage to the concrete wall if the network was ever decommissioned and also because if the sensors fell off the wall they would be unlikely to cause damage (except to themselves) so there was not a safety concern in this instance. Of the seven wall-mounted sensors, three fell off within 24 hours of their initial installation and had to be reattached. These failures were due to the porous nature of concrete, which means that it absorbs liquid. Thus if not enough glue is applied to the sensor before it is attached to the wall, too much of the glue will be absorbed by the concrete and adhesion will not occur. A second potential issue is that the concrete also absorbs moisture, which can result in the wall being too damp for the glue to set properly. The sensors that initially fell off were reattached with more glue in areas where the concrete was confirmed to be dry. It is also worth noting that the concrete walls in the anchorage were actually relatively clean in comparison to most structures that are exposed to the elements and so gluing is even less likely to work in other applications due to dirt interfering with the adhesive bond. At the Ferriby Road Bridge we attempted to attach the inclinometers to the bearings of the bridge using high-strength epoxy. The first issue arose when, on a cold March day, the epoxy would not set resulting in a week's delay in installing the sensors. Then, over the course of a year, one of the sensors (out of the three attached with epoxy) that had been particularly difficult to install in the first instance simply fell off. In both the anchorage and the Ferriby Road Bridge deployments it was fortunate that the sensors fell only a short distance so that no damage to people or equipment occurred.

The reality is that the sensors need to be attached to the structures using mechanical fasteners because the risk of damage to vehicles or pedestrians is simply too great to risk any other type of attachment. Unfortunately unlike gluing, mechanical fastening takes time (to drill holes, insert screws etc.) and so you must also build this into your deployment plan and timing.

Figure 7: We designed and built an LPDT sensor box with two Linear Potentiometer Displacement Transducers (LPDTs) to measure crack width to a resolution of 10 μm. The box is fastened to the concrete bar below and the potentiometer cursors are attached, with brackets, to either side of the crack that lies under the box. One of them measures the crack width and the other the thermal expansion of the concrete, for calibration. (Cfr. principle 14.)

Principle 14. *If sensors that measure what you want don't exist, you must make your own.*

The Crossbow hardware we used in our deployments has, as most others, a modular design based on a so-called "mote" (the basic processing module with a radio interface) that can be instrumented with a selection of optional sensor boards that measure environmental parameters such as temperature and relative humidity; but we also needed to measure changes in geometry (i.e. movement of the sides of a crack). Thus we had to develop our own bespoke sensor boards to measure displacement and inclination as described in figures 7, 9, 10 and 8, 9, 11 respectively.

Current COTS systems are unlikely to have the exact sensors that you require, as even the current environmental sensor boards may not be applicable. For example, Crossbow provides several boards that measure both temperature and light. While you may want to measure both of these parameters, typically the temperature sensor should be shaded to give an accurate measurement, which would then have a negative impact on the light reading! Thus whilst you may hope to find an all in one COTS system, the likelihood is that you will need to create sensor interfaces specific to your application.

Further reading: [9].

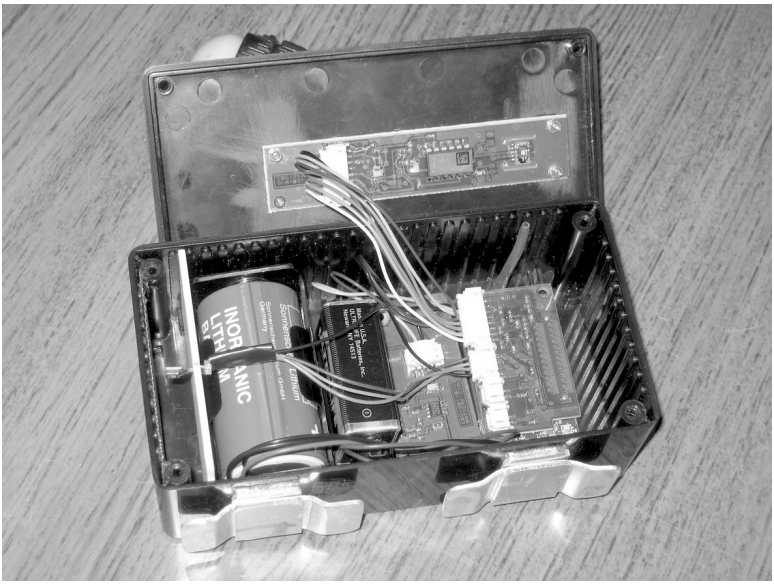

Figure 8: We designed and built an inclinometer sensor box to measure changes in the angle of a structural element (with respect to the vertical) to a resolution of 0.001 degrees. To achieve this precision we used a 16-bit ADC instead of the mote's built-in 10-bit one to sample the analogue output of the MEMS inclinometer element. (Cfr. principle 14.)

Principle 15. *Sensor failures: you must be prepared for the unexpected to happen.*

Most electronics eventually fail; however, if you want a viable WSN system, you need to minimize the chances of this happening and maximize the lifetime of the sensors.

In our case, many of the inclinometers that we deployed in the London Underground mysteriously failed after about 6 months when we changed the batteries. After removing the sensors from the tunnel (which required both physical access and staff time) we discovered that the soldering on the connector between our interface board and the Crossbow mote had failed. There are two potential causes for this: the use of lead-free solder (owing to recent environmental regulations) which, as is well known, doesn't flow as well as the traditional solder, resulting in a higher proportion of dry joints; or, simply, poor soldering workmanship. We have since corrected both of these problems by sending the boards out to a company who specializes in populating circuit boards and uses lead-based solder to do it[10].

These might sound like minor considerations but these are the things that separate a system that works for a few months from a system that works for years.

[10] We could afford not to comply with the regulations because these were research prototypes rather than products for sale.

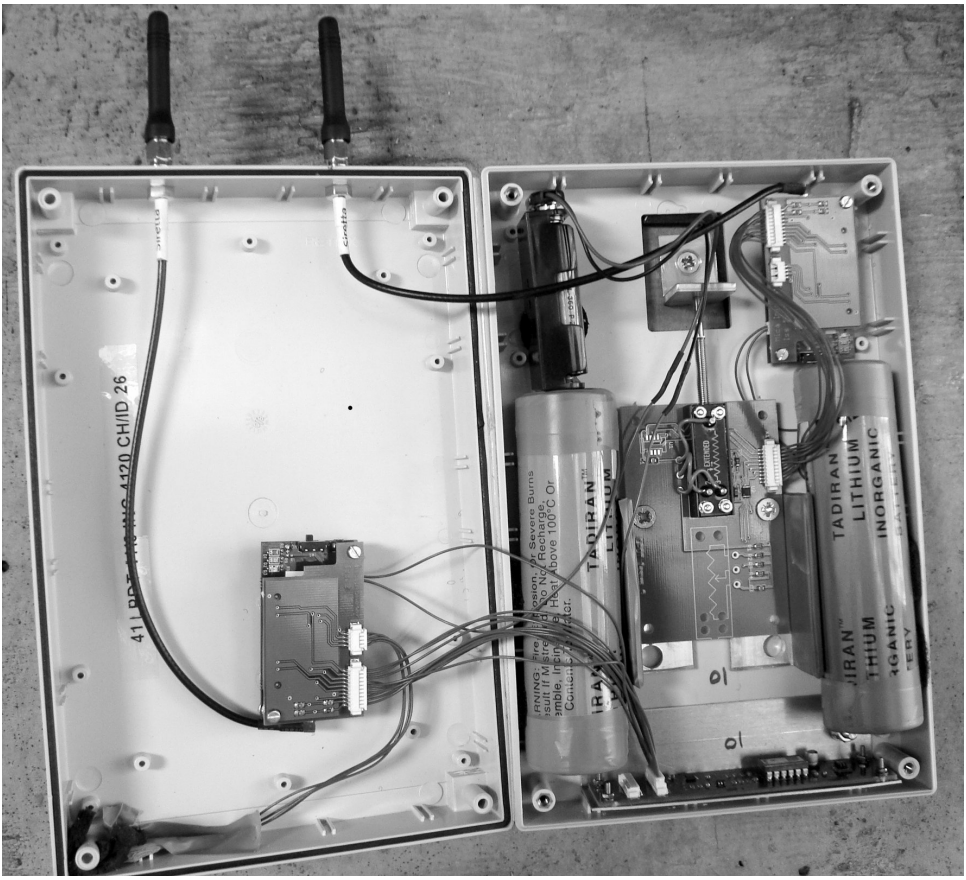

Figure 9: This latest revision of our sensor box incorporates both the LPDT board (center right) and the inclinometer board (bottom right), each as an independent unit attached to its own high capacity (38 Ah, double D-size) lithium battery and Crossbow Iris mote (center left and top right). We replaced our earlier choice of the Crossbow MICAz mote with the Crossbow Iris mote because of the superior radio performance of the latter. Our own electronics were also revised: see figures 10 and 11 for the most recent circuit diagrams. Note that, in this new design of the LPDT board, the second linear potentiometer is optional (and indeed not even mounted here), as we found the thermal expansion effects to be negligible and not worth monitoring in this particular application. (Cfr. principle 14.)

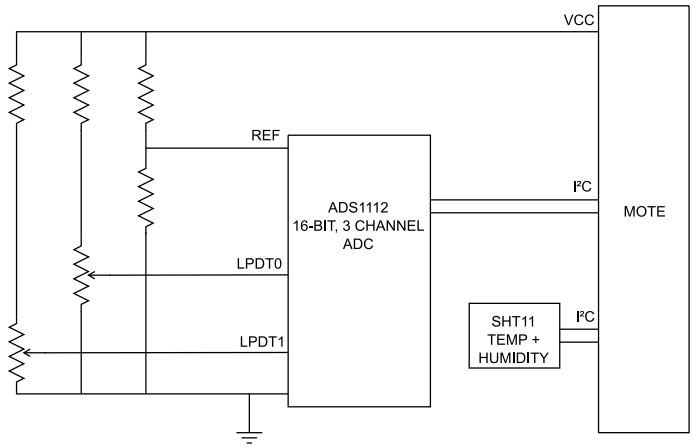

Figure 10: Circuit diagram of LPDT sensor board (cfr. figures 7 and 9). We use a 16-bit ADC rather than the 10-bit one of the mote because the higher resolution is necessary for measuring strain, rather than just crack width. We digitize the output of the two potentiometers and also, for calibration, the supply voltage received from the mote. We also sample temperature and humidity with another single-chip sensor. (Cfr. principle 14.)

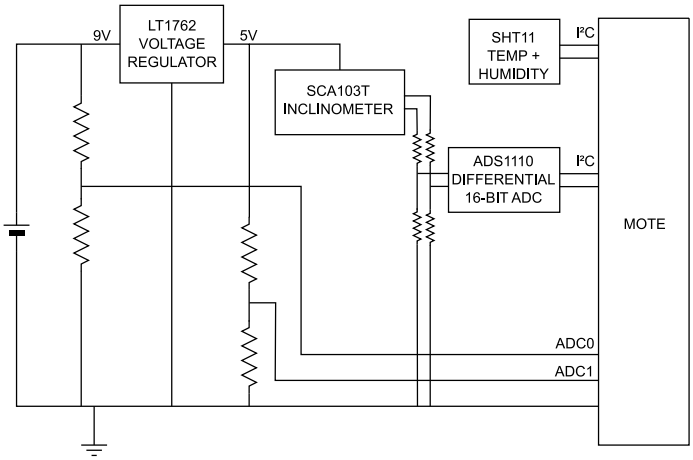

Figure 11: Circuit diagram of MEMS inclinometer sensor board (cfr. figures 8 and 9). A voltage regulator feeds the MEMS inclinometer with a clean 5V supply derived from the battery. The inclinometer's differential analogue output is digitized at 16 bits. Having learnt from experience (see principle 16) we now monitor the supply voltage, both before and after the regulator, using the mote's built-in lower precision ADC. We also include a single-chip temperature and humidity monitor as in the LPDT sensor board. (Cfr. principle 14.)

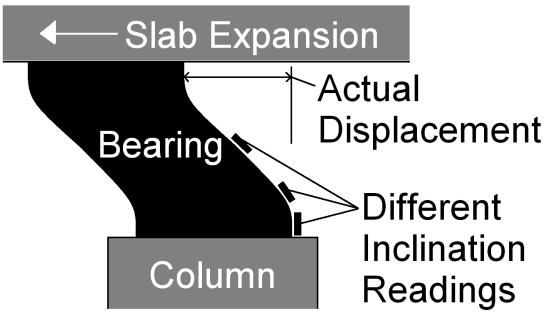

Figure 12: Measuring displacement with an inclinometer yielded unexpected results. We assumed the side of the rubber bearing would incline in a straight line as the slab expanded, whereas it instead deformed into a curve. Therefore the measured angle depended on where the inclinometer had been attached. In retrospect we should have measured the actual displacement, not the angle. (Cfr. principle 17.)

Principle 16. *You must be able to find out exactly* what *happened.*

Another interesting problem we noticed in many of our deployments was that data received from the inclinometers suggested that some had fallen off! But when we went to check where they had fallen to we discovered that they were actually all still attached (which came as a bit of a relief).

The actual problem was that the 9 V batteries that power the inclinometers had drained and were no longer supplying the 5 V required by the inclinometers. Rather than not working, the inclinometers simply started outputting a lower voltage, which was then interpreted as a change in angle. We overcame this problem by adding additional measurement lines on both the 9 V battery and the power being supplied to the inclinometer (see figure 11).

This anecdote illustrates the need to be able to independently verify or reconcile changes in measurement, whether that is through additional measurements, a surveillance camera or even direct visual inspection.

4.5. You've Got Data—So What?

Principle 17. *You must think about what you're measuring and why.*

Many times researchers install sensors because they are available rather than because they provide the data that is actually needed. We hereby confess that we are just as guilty of this as anyone.

On the Ferriby Road Bridge we wanted to know whether the rubber bearings that support the bridge had moved beyond their allowable range. When the bearings move they incline; we had inclinometers—a perfect match! Unfortunately, when the bearings incline, the shape they take on is complex (figure 12) and so the reading you get depends on where you put the inclinometer. The values we get tell us if the bearing is moving, but not the magnitude of this movement. A far better choice would have been to measure the displacement between the top and bottom of the bearing thus measuring the total movement.

You need to think before you measure. If the data you will get is not the data you need, there is no point in putting that sensor on in the first place.

Further reading: [9, 10].

Principle 18. *You must understand the end-users and their workflow and find a way to present the data that makes sense to them.*

With any monitoring system there are potentially a number of end users, each hoping to use the data in a different way.

The first distinction is that there are those who are concerned with the data from the network and those who are concerned with the management of the network itself. These two groups want to know entirely different things. For example, network administrators (or in our case our radio propagation and security people) want to know about the connectivity of the network, whereas civil infrastructure owners (or civil engineering researchers) are unconcerned with this as long as it works. (Similar concerns are discussed by Barrenetxea et al. [2], who also self-referentially note the necessity of planning ahead to collect the appropriate data for being subsequently able to write a paper about the experience.)

There are also different levels of abstraction required for the data, as managers may only want to know when action is required whereas engineers may require the complete data set for analysis. Unless the data is made available and presented in a way that is tailored to the user, it ends up being of no use to anyone.

In our case we have developed a web-based interface that plots the data, shows minimums and maximums, and allows access to the database containing both the network and sensor data.

Principle 19. *You must strive to preserve and present the spatial origin of the data.*

A further point to be kept in mind is the need to locate the source of the data in space. If this is not done, valuable time could be wasted trying to locate the origin of the critical data; or potentially critical data could be missed because its true importance was not realized.

In this project we have created 3-D models of our structures, which are then overlaid on Google Earth as shown in Figure 13. This allows the data to be located on the structure and ultimately may also allow for the data from nearby monitoring systems to be accessed and visualized.

Further reading: [9, 11].

5. Conclusions and further work

In our experience, the WSN systems currently available on the market are still insufficiently mature for practical deployments: the basics are there but many problems, from radio propagation to deployment to security, are still left unaddressed. Domain experts such as bridge and tunnel engineers are still far from being able to buy a turn-key WSN system and apply it to their own problem: instead, they are

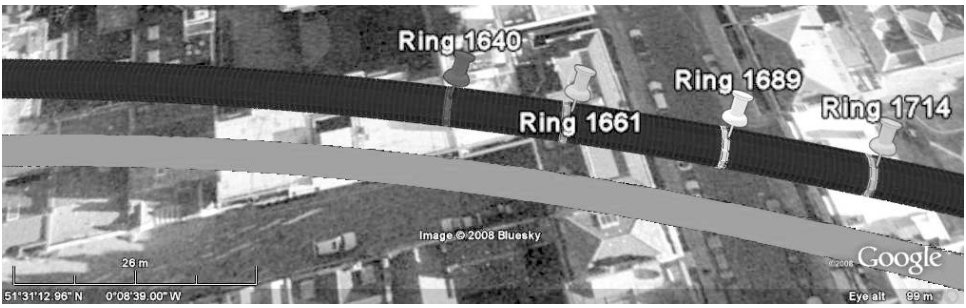

Figure 13: Underground tunnel overlaid on a map of London. The tunnel sections instrumented with sensors are highlighted and clicking on the pins gives access to the underlying readings. The tunnel is shown as floating above ground level because the software does not support underground structures. (Cfr. principle 19. Google Earth image courtesy of Google.)

forced to become co-developers of the WSN system and to build and sharpen their own tools. This situation needs to evolve and we sincerely hope that the principles we provided in this paper will help future adopters and developers.

Although we have done our best to address many of the issues that we encountered during our deployments, there are still a number of open research problems for the WSN community:

- Development of radio propagation models for complex environments.
- Exploitation of channel diversity in a power efficient and low complexity manner.
- Development of a "one pass" algorithm for the deployment of relay nodes that allows each candidate location to be visited at most once by the people in hard hats (for radio measurement and, if appropriate, node installation).
- Robustness against routing attacks.
- Safeguards to protect data integrity against slow corruption attacks.
- Satisfactory resolution of the "features vs security" trade-off.
- Management, integration and data sharing across WSNs belonging to multiple owners.

When talking specifically of civil infrastructure monitoring, rather than WSNs in general, the following problems are also still open:

- Development of other sensors. What transducers could we use to detect, for example, cracks in steel structures due to fatigue of the metal, or corrosion of the steel in reinforced concrete structures?
- The above-mentioned faults can occur at random locations throughout the structure: is a WSN really the best approach to detect such problems? What alternatives are there? Maybe the WSN is best for monitoring performance

after a repair, or for monitoring how an already-detected fault is progressing, than for detecting unknown faults; and we should perhaps look for other monitoring methods to complement the strengths of the WSN approach.

Despite all these open problems, we believe that there is real potential for WSNs one day to become an essential subsystem that will be easily and fruitfully integrated into much of the engineering infrastructure. Once the above research problems are addressed, the WSN industry will be in a position to develop products and systems that domain experts will be able to adopt and apply to their monitoring problems without having to become WSN developers themselves.

6. Acknowledgements

We are grateful to EPSRC for funding this work as part of project EP/D076870/1 "WINES Smart Infrastructure". We thank our civil infrastructure partners, both as individuals and as institutions, for their assistance and cooperation: Peter Hill (Humber Bridge Board), Peter Wright (Tubelines), Jim Moriarty (London Underground), Stephen Pottle (Transport for London). We gratefully acknowledge the essential contributions of other members of our team including, in alphabetical order, Dan Cvrcek (security; installation tools), Paul Fidler (node and gateway firmware), Chikara Hirai (deployment tools), Brian Jones (circuit board design), Yusuke Kobayashi (deployment tools), Matt Lewis (security), Min Lin (propagation measurement equipment development and empirical modelling), Ruoshui Liu (deployment), Yan Wu (electromagnetic propagation modelling). We also thank Cecilia Mascolo and Jon Crowcroft for commenting on an early draft. We are especially grateful to one of the anonymous reviewers of a previous version for pointing us towards the Barrenetxea et al. [2] paper, which inspired the structure of this one.

References

[1] A. Arora, R. Ramnath and E. Ertin. "Exscal: Elements of an Extreme Scale Wireless Sensor Network". In "Proc. 11th IEEE RTCSA", pages 102–108. 2005.

[2] G. Barrenetxea, F. Ingelrest, G. Schaefer and M. Vetterli. "The hitchhiker's guide to successful wireless sensor network deployments". In "SenSys '08: Proceedings of the 6th ACM conference on Embedded network sensor systems", pages 43–56. ACM, New York, NY, USA, 2008. ISBN 978-1-59593-990-6. doi: http://doi.acm.org/10.1145/1460412.1460418.

[3] K. Chintalapudi, T. Fu, J. Paek, N. Kothari, S. Rangwala, J. Caffrey, R. Govindan, E. Johnson and S. Masri. "Monitoring Civil Structures with a Wireless Sensor Network". *IEEE Internet Computing*, **10**(2):26–34, 2006.

[4] D. Didascalou, J. Maurer and W. Wiesbeck. "Subway tunnel guided electromagnetic wave propagation at mobile communications frequencies". *IEEE Transactions on Antennas and Propagation*, **49**(11):1590–1596, Nov 2001. ISSN 0018-926X. doi:10.1109/8.964095.

[5] G. Feltrin, J. Meyer and R. Bischoff. "Wireless sensor networks for long term monitoring of civil structures". In "Proceedings of the Second International Conference on Experimental Vibration Analysis for Civil Engineering Structures", pages 95–111. Porto, Portugal, 2007.

[6] C. Grosse. "Monitoring of structures using wireless sensors and acoustic emission techniques". In "Proceedings of the international conference on Challenges for Civil Construction", pages 28–38. Porto, Portugal, 2008.

[7] C. Hirai and K. Soga. "An Experimental Model of Relay Deployment Planning Tool for a Wireless Sensor Network System to Monitor Civil Engineering Structures". In "Proceedings of The Ninth IASTED International Conference on Parallel and Distributed Computing and Networks (PDCN 2010)", Feb 2010.

[8] R. Holt and J. Hartmann. "Adequacy of the U10 & L11 Gusset Plate Designs for the Minnesota Bridge No. 9340 (I-35W over the Mississippi River)". Technical report, Federal Highway Administration, Turner-Fairbank Highway Research Center, January 2008.

[9] N. Hoult, P. Fidler, P. Hill and C. Middleton. "Long-term Wireless Structural Health Monitoring of the Ferriby Road Bridge". *ASCE Journal of Bridge Engineering*, 2009. In press.

[10] N. Hoult, P. Fidler, P. Hill and C. Middleton. "Wireless Structural Health Monitoring of Bridges: Present and Future". *International Journal of Smart Structures and Systems*, 2009. In press.

[11] N. Hoult, P. Fidler, I. Wassell, P. Hill and C. Middleton. "Wireless Structural Health Monitoring at the Humber Bridge". *Proceedings of ICE—Bridge Engineering*, **161**(BE4):189–195, 2008.

[12] P. Johnson, A. Couture and R. Nicolet. "Report of the Commission of inquiry into the collapse of a portion of the de la Concorde overpass". Technical report, Gouvernement du Québec, 2007.

[13] J. M. Kahn, R. H. Katz and K. S. J. Pister. "Next Century Challenges: Mobile Networking for 'Smart Dust'". In "Proceedings of International Conference on Mobile Computing and Networking (MobiCom 99)", Seattle, WA, August 1999.

[14] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser and M. Turon. "Health Monitoring of Civil Infrastructures Using Wireless Sensor Networks". In "Proc. 6th IPSN", pages 254–263. ACM Press, 2007.

[15] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser and M. Turon. "Health Monitoring of Civil Infrastructures Using Wireless Sensor Networks". In "Proceedings of 6th International Symposium on Information Processing in Sensor Networks (IPSN 2007)", pages 254–263. Cambridge, MA, USA, 2007.

[16] L. Krishnamurthy, R. Adler, P. Buonadonna, J. Chhabra, M. Flanigan, N. Kushalnagar, L. Nachman and M. Yarvis. "Design and Deployment of Industrial Sensor Networks: Experiences from a Semiconductor Plant and the North Sea". In "Proc. 3rd SenSys", pages 64–75. ACM Press, 2005.

[17] R. Liu, Y. Wu, I. J. Wassell and K. Soga. "Frequency Diversity Measurements at 2.4 GHz for Wireless Sensor Networks Deployed in Tunnels". In "Proc. 20th IEEE International Symposium on Personal Indoor Mobile Radio Communications (PIMRC'09)", Tokyo, Japan, September 2009.

[18] J. Lynch, Y. Wang, K. Loh, J.-H. Yi and C.-B. Yun. "Performance monitoring of the Geumdang Bridge using a dense network of high-resolution wireless sensors". *Smart Materials and Structures*, **15**(6):1561–1575, 2006.

[19] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk and J. Anderson. "Wireless Sensor Networks for Habitat Monitoring". In "Proc. 1st ACM WSNA", pages 88–97. ACM Press, 2002.

[20] J. Molina-Garcia-Pardo, M. Lienard, A. Nasr and P. Degauque. "On the Possibility of Interpreting Field Variations and Polarization in Arched Tunnels Using a Model for Propagation in Rectangular or Circular Tunnels". *IEEE Transactions on Antennas and Propagation*, **56**(4):1206–1211, April 2008. ISSN 0018-926X. doi:10.1109/TAP.2008.919220.

[21] C. O'Connor. *Roman Bridges*. Cambridge University Press, 1993. ISBN 0521393264.

[22] F. Stajano, D. Cvrcek and M. Lewis. "Steel, Cast Iron and Concrete: Security Engineering for Real World Wireless Sensor Networks". In S. M. Bellovin, R. Gennaro, A. D. Keromytis and M. Yung (editors), "Proceedings of 6th Applied Cryptography and Network Security Conference (ACNS 2008)", volume 5037 of *Lecture Notes in Computer Science*, pages 460–478. June 2008. ISBN 978-3-540-68913-3. doi:10.1007/978-3-540-68914-0_28.

[23] G. Werner-Allen, K. Lorincz, M. Welsh, O. Marcillo, J. Johnson, M. Ruiz and J. Lees. "Deploying a Wireless Sensor Network on an Active Volcano". *IEEE Internet Computing*, **10**(2):18–25, March-April 2006.

[24] Y. Wu, M. Lin and I. Wassell. "Path Loss Estimation in 3D Environments using a modified 2D Finite-Difference Time-Domain Technique". In "IET 7th International Conference on Computation in Electromagnetics (CEM 2008)", Brighton, UK, April 2008.

[25] Y. Wu, M. Lin and I. J. Wassell. "Modified 2D Finite-Difference Time-Domain Based Tunnel Path Loss Prediction for Wireless Sensor Network Applications". *Journal of Communications (JCM),*, **4**(4):214–223, May 2009.

[26] Y. Wu and I. Wassell. "Investigation of Close-to-Wall Wireless Sensor Deployment Using 2D Finite-Difference Time-Domain Modelling". In "2nd International Conference on Wireless Communications in Underground and Confined Areas (ICWCUCA)", Val-d'Or, Québec, Canada, August 2008.

[27] K. Yee. "Numerical Solution of Initial Boundary Value Problems involving Maxwell's Equations in Isotropic Media". *IEEE Transactions on Antennas and Propagation*, **14**(3):302–207, May 1966.