Testbed Implementation of a Pollution Monitoring System Using Wireless Sensor Network for the Protection of Public Spaces

Siuli Roy, Anurag D and Somprakash Bandyopadhyay
Management Information Systems
Indian Institute of Management Calcutta
anurag@email.iimcal.ac.in, {siuli, somprakash}@iimcal.ac.in

Abstract

Air pollution is an important environmental issue that has a direct effect on human health and ecological balance. Factories, power plants, vehicles, windblown dust and wildfires are some of the contributors of pollution. Reasonable simulation tools exist for evaluating large scale sensor networks, however they fail to capture significant details of node operation or practical aspects of wireless communication. Real life testbeds, capture the realism and bring out important aspects for further research. In this paper, we present an implementation of a wireless sensor network testbed for automatic and real-time monitoring of environmental pollution for the protection of public spaces. The paper describes the physical setup, the sensor node hardware and software architecture for "anytime anywhere" monitoring and management of pollution data through a single web-based graphical user interface. The paper presents practical issues in the integration of sensors, actual power consumption rates and develops a practical hierarchical routing methodology.

Keywords: Real-time monitoring, hierarchical routing, testbed implementation, ZigBee, pollution sensors.

I. INTRODUCTION

The availability of consistent, accurate and timely information on environmental conditions greatly improves the speed of planning and decision making (Hammons & Chisholm, 2006). Such information, although required for most businesses, is of particular importance in chemical factories, mines and other high investment industries. The availability of precise information enables businesses to respond in quick time and take preventive measures against emission of poisonous gases and other hazards, thus improving the safety of personnel and equipment.

Air pollution (Jung, Lee, Lee, Ryu, & Nittel, 2008; Ma, Richards, Ghanem, Guo, & Hassard, 2008) is an important environmental issue that has a direct effect on human health and ecological balance. The primary airborne pollutants covered by European legislations are: SO_2, NO_x (NO/NO_2), Benzene, Ozone, CO/CO_2 and particulate matter (PM10/PM2.5). Air pollution has diverse causes and sources. "Stationary sources" such as factories, power plants and smelters; "mobile sources" such as automobiles; and "natural sources" such as windblown dust and wildfires are primary contributors to air pollution. Due to the trans-boundary nature of airborne pollutants, it is difficult for any single organization to take responsibility for overall emission levels. Thus, the control of air pollution is entirely legislation driven. As such the passing of new legislation may only be effective if the pollution level in the specified compounds can be monitored accurately using sensors.

Environmentalists can use sensors to measure atmospheric pollution and monitor industrial emissions; safety monitors can use sensors to detect harmful chemical vapours and explosives in public spaces, government or military facilities and chemical processing plants.

Gas sensor technologies are still developing and have yet to reach their full potential in capabilities and usage (Jung, Lee, Lee, Ryu, & Nittel, 2008). Some technologies are particularly accurate but prohibitively expensive for large-scale deployment. By using a sensor network, the problem of false positives could potentially be reduced. Multiple outputs can be compared for a more accurate analysis.

However, realizing sensor-based networks using wires are not feasible solutions for large scale deployment. Wireless sensor networks (Akyildiz & Wang, 2005) offer powerful new ways to monitor air quality and without the costs of major new installations or wire-runs that are typically associated with these types of projects. Wireless sensor networks comprises of:

- A multitude of bi-directional radio transceivers with sensors known as sensor nodes or tags. The sensors include a wide variety of pollution detectors and are connected together using the communicational capability of the wireless nodes to form the core of our proposed pollution monitoring system. Sensors capture pollution data and communicate them wirelessly over the network to a remote control station for further analysis and generation of alerts during critical events.

- The wireless sensor nodes are arranged in a networking topology called a “mesh”. Mesh networking is a type of network where each node can communicate with multiple other nodes thus enabling better overall connectivity than in the traditional hub-and-spoke or star topologies. State of the art mesh networks often have some or all of the following characteristics:

a) They are self-forming. As nodes are powered on, they automatically enter the network, compute efficient routes for data forwarding through their neighbors and are ready for operation.

b) They are self-healing. As a node leaves the network, the remaining nodes automatically re-route their packets around the out-of-network node to ensure a more reliable communication path.

c) They support multi-hop routing. This means that data from a node can jump through multiple nodes before delivering its information to a host gateway or controller that may be monitoring the network thus extending the reach and range.

The self-forming and self healing coupled with battery operable attributes of a mesh sensor network make it ideal for environmental monitoring applications in a wide range of facilities.

This paper presents a generic framework for “anytime anywhere” visibility of sensor data from a remote wireless sensor network using the Internet (Pujolle, 2006). Active RFID tags, combined with sensors and actuators, form the proposed wireless sensor network based on IEEE 802.15.4 (IEEE, 2003). The architecture also enables the control of the environment by triggering actuators. Lastly, we elaborate a test-bed implementation for monitoring environmental pollution for possible detection of toxic gases based on the above framework.

The paper is structured as follows. Related work on the test-bed implementation of wireless sensor networks is provided in section 2. The proposed conceptual framework for wireless sensor network for pollution monitoring is presented in section 3. Detailed system architecture is provided in section 4. Different technology options for wireless sensor network, components of the proposed wireless sensor node, software architecture and setting up of testbed is illustrated in section 5. Section 6 concludes the paper highlighting the scope for further work.

II. Related Work

In (Jung, Lee, Lee, Ryu, & Nittel, 2008), the design and implementation of an air pollution monitoring system based on a geo-sensor network has been discussed. In order to detect the status of air pollution, a context model which includes sensor data abstraction and air pollution prevention model has been designed. The sensed data is combined by the abstraction model after collecting the various kinds of geo-sensor data. The level is derived by the user defined rules using the values of the geo-sensors such as dust, carbon dioxide, ultraviolet, etc. The abstracted data also includes the boundary of the area, the sensors’ properties, air pollution types and maximum levels. This local abstracted data is combined in the global area by the air pollution prevention model. The system provides an alarm and safety guideline for near future dangerous situations and prevents severe damage and mitigates recovery cost. It also supports the flexible sampling-interval change depending on the pollution conditions of the context model.

In order to monitor the pollutants and analyze their effects on the environment, a Mobile Discovery Net (MoDisNet) was developed in (Ma, Richards, Ghanem, Guo, & Hassard, 2008) to collect real time pollution data on key aspects of traffic, emissions, ambient pollutant concentration and human exposure. The purpose is to develop a capability to measure, model and predict a wide range of environmental pollutants and hazards using a grid of pervasive roadside and vehicle/person-mounted sensors. The major challenge, as pointed out, is integrating data from heterogeneous fixed and mobile environmental sensor grids in real time to provide dynamic estimates of pollutants and hazard concentrations. This should demonstrate how such data can be usefully correlated with a wide range of other complementary dynamic inputs, such as weather, health or traffic. The MoDisNet system is designed as a two-layer network architecture – the mobile sub-network formed by the Mobile Sensor Nodes installed in vehicles and the stationary sub-network organized by the Static Sensor Nodes. The mobile nodes may pre-process the raw data (such as the noise reduction, local data cleaning and fusion, etc.) and send the data to a nearest static node. The static nodes now in turn, account for the data reception, updation, storage and exchange. Cooperating with a grid architecture, the static nodes can realize the distributed data analysis and mining.

As pointed out in (Werner-Aller, Swieskowski, & Welsh, 2005), deploying a wireless sensor network into a realistic environment requires iterative programming of dozens of nodes, locating them throughout an area large enough to produce an interesting radio topology and extract debugging and performance data. The advent of networked interfaces for sensor nodes, such as the Crossbow MIB-600, makes remote reprogramming and monitoring of permanently-powered sensor network nodes possible. The MoteLab sensor network testbed platform addresses these challenges. MoteLab also allows users to interact with individual nodes directly during experiments. By providing a web interface to the testbed, MoteLab simplifies and accelerates deployment and evaluation of wireless sensor network applications. They have also instrumented a node in MoteLab with a network-connected digital multimeter, allowing the MoteLab to continuously monitor the energy usage of the node. Current consumption data is logged and returned with other data generated during the experiment.

The experience of implementation, deployment and operation of SensorScope, an indoor environmental monitoring network is reported in (Schmid, Dubois-Ferrière, & Vetterli, 2005). From the data gathered, it has been observed that the network performance is greatly improved by using MAC layer retransmissions.

The tesbed experimentation done so far on wireless sensor networks for pollution monitoring does not take care of the requirements of “anytime anywhere” access of pollution data as well as control of environment, alert generation and sleep management through a single user interface. We propose to address these issues through development of a generic framework.

III. Conceptual Framework

Small, autonomic wireless devices, cheaply priced and capable of forming communications networks are expected to be the next technological innovation to bring about a mass cultural change. This new concept is due to the emergence of Device Area Network where billions of devices are inter spread in the everyday activities of humans. The devices help organize, manage and control a host of equipment and services. This device area network can be further extended to offer “Anytime Anywhere” visibility of network data over the Internet. This initiates a new technology paradigm called the “Internet of Things” (ITU, 2005).

Such a network comprises of wireless devices typically, sensors and actuators which are capable of forming wireless networks among themselves and are designed to run on batteries for prolonged periods of time. These devices and such networks need to support a low data rate and the computational capability is restricted. Low data rate protocols have been designed specifically for these needs lately, a significant deviation from the trends of building high-bandwidth networks. The protocol ratified by IEEE is 804.15.4 in 2003 (IEEE, 2003) and updated in 2006 (Wheeler, 2007). The 802.15.4 standard provides specifications for the Physical and Medium Access Layer. The upper layers (Network to Application) have been developed by a consortium of wireless device manufacturers and enthusiasts called ZigBee (ZigBee Alliance, 2004).

In this paper, we have developed an end to end architecture for remote monitoring of environment and remote actuation necessary for controlling the environment. The architecture comprises of three basic components: (a) the wireless mesh network, consisting of RFID tags connected to sensors and/or actuators (which may also be called as wireless sensor actuator network), (b) the gateway or the central server and (c) the Internet. The mesh networks collect local information (pollution level) which is communicated through a gateway to the central server. The gateway handles the messages to and from the mesh network. The gateway is expected to be A/C mains powered while the individual nodes (sensors/actuators) are battery driven. The information is then transported to a web end-point by the Central Server. The central server essentially does the job of protocol conversion from the Internet to the sensor network and vice-versa. The information is displayed in the requisite format and provides an interface for object management. The web end point application can be a simple hosted page viewed on a personal computer or a message transmitted to a mobile device. We could think of supporting both push based and pull based systems. A push based system updates the status of the devices automatically, either periodically or when a status changes. In a pull based system, the latest information is provided when the user asks for it. The conceptual framework is pictorially depicted in figure 1.

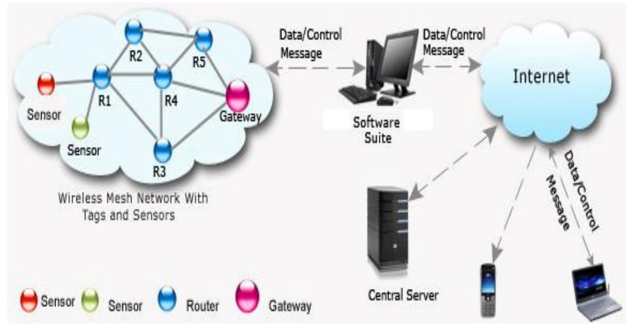


Figure 1. A conceptual framework for Monitoring and Actuation of environment through the Internet

IV. Detailed System Architecture

As depicted in figure 1, the proposed architecture has three components: Wireless Sensor-based Mesh Network, Central Server or Control Station and any web-enabled device accessing the central server through the Internet. Each of these components are explained in greater detail below.

a) Wireless Mesh Network

A Wireless Mesh Network, as its name implies, is a type of network where each node in the network can communicate with multiple wireless nodes, thus enabling better overall connectivity. A wireless mesh network has two kinds of nodes: the mesh routers and the mesh clients. The mesh clients can be made to communicate with multiple routers/other clients, or arranged in a hierarchical fashion where every client has a single router as its parent. A major advantage of this kind of network, other than self-forming and self-healing capability, is multi-hop routing. This means that data from a wireless node can jump through other nodes before delivering its information to a remote host gateway or controller that collects the data from sensors for further processing. Low power mesh networking, sensing and active RFID-based real-time location tracking has enabled us to design systems for tracking, locating and monitoring of people and other assets including environmental conditions.

Passive RFID tags (RFID without a battery) are already driving shifts in supply chain and retail capabilities for automatic identification of objects. However, Active RFID has much broader potential in the enterprise. Firstly, active RFID can form a wireless mesh network, providing automatic, dynamic visibility into what is going on in and around the enterprise. Secondly, active RFID technology, if combined with sensors and actuators in a networked environment, enables a spectrum of applications that can exponentially increase visibility and monitoring. Offering much more information than passive RFID, sensors can monitor and record conditions like temperature, humidity, pressure, wind direction and speed, illumination intensity, vibration intensity, sound intensity, chemical concentrations, pollutant levels and so on.

An important aspect of the wireless mesh network is to develop an efficient route to multi-hop the data packet from the sensor to the sink (or gateway). The de-facto standard for routing in mesh networks is AODV (Ad-Hoc On-demand Distance Vector) (Perkins & Royer, 1999; Al-Karaki & Ahmed, 2004) which develops route tables by broadcasting control packets. If we study the topology of the wireless mesh, the sensors and the gateway are static. The intermediate routers, which relay the data, are static as well. The AODV algorithm was designed for mobile nodes and its application in a static mesh network would be wasteful in terms of resources. A competing routing methodology is the hierarchical tree which labels every node such that no route tables are needed. Such labeling does not need broadcast control packets. A comparison (Cuomo, Luna, Monaco, & Melodia, 2007) with the hierarchical algorithm has shown AODV to incur a large number of control packets for developing and maintaining an active route thus expending a larger amount of energy and results in a slightly lower throughput. The hierarchical algorithm on the other hand suffers from its inability to support networks of large depths and fault tolerance (Wheeler, 2007). The ZigBee Stack (ZigBee Alliance, 2004) supports the hierarchical (denoted as C_{skip}) and AODV based routing and they have been compared in (Peng, Mao-heng, & You-min, 2006). Devices which join the network are given an address based on rules set by the network parameters. These parameters are predetermined and cannot be changed throughout the network lifetime. The Parameters are C_m, R_m and L_m; where C_m= Maximum Number of children that a full functional device (FFD) can have; R_m = Maximum number or children (out of C_m) which are FFDs; and L_m = Maximum depth of the network. The values of these parameters are stored in the NIB (Network Information Base) in each device (Ho-In & Yeonsoo, 2007).

An address of a requesting child is generated by the equation:

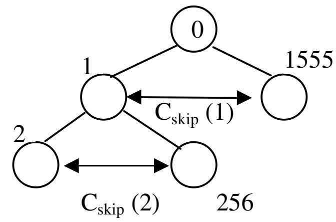
$$A_n = A_{parent} + C_{skip}(d)\, R_n + n \qquad (1)$$

Where A_n is the address which the new device will take. A_{parent} is address of the parent of the device that will assign the address. $C_{skip}(d)$ is determined as follows:

$$C_{skip}(d) = (1 + C_m - R_m - C_m * R_m\, C_m - d - 1) / (1 - R_m) \qquad (2)$$

$C_{skip}(d)$ determines the block of addresses which the parent device must skip before assigning the next address. The algorithm assumes the worst case scenario and makes provision for accommodation for all devices in the pre-determined network architecture. This assumption of a worst case scenario severely restricts the network depth. For example, values of C_m = 6, R_m = 6 makes L_m=7 for a 16 bit short address.

The addressing scheme based on C_{skip} develops a tree topology which makes possible the optimum routing. Routing in such networks is made by comparing the destination address with the C_{skip} allocation block. If the destination address is within the C_{skip} block of any of its children, the packet is forwarded to that child, else the packet is forwarded to its parent. It has been shown that such tree based routing provides the minimum latency (Peng, Mao-heng, & You-min, 2006). The address wastage problem in C_{skip} is pictorially shown in figure 2.



A sample topology with R_m = C_m = 6. If no more devices join the leg with node "1", a total of 1551 (1553 - 2) addresses are wasted.

Figure 2. The address wastage problem in C_{skip}

Our test bed implementation used a variation of the hierarchical network where node addresses are made once the network is in place. The addressing follows a depth first approach where the first address is given to the leaf of the network and works upwards towards the root. The addressing and routing in such a mechanism is show as algorithm 1 & 2.

The addressing algorithm is akin to the C_{skip} in the sense of addressing in the depth first approach. However, C_{skip} assumes the worst case and would earmark addresses for non-existent nodes. This leads to a huge amount of address wastage and this precisely is avoided in the static algorithm. The monitoring applications have a well defined path and this is known in advance. Thus, the need to maintain addresses for future nodes does not arise. Further, the address allocation can be suitably tweaked to include an address provision for future nodes as the need may be. This can be achieved, by including a larger value for E_n.

Algorithm 1: Modified Hierarchical Address allocation

```
Determine location of FFDs and their parent/children
input : En, set R of Routers and their children
Initialise node ← PAN coordinator, addr ← 0
For all children of node from left to right
  If node has no children
      assign node address ← addr
      Increment addr by En + 1
  End if
Else
   assign node ← node's children
End if
End For
```

Every router maintains an address list, which has the addresses of its child routers and child end devices. The routing is done based on this and is shown as Algorithm 2.

Algorithm 2: Routing in Modified Hierarchical Topology

```
input : set C, children address of Router R, Destination address, D, of received packet at R
For all children of node R from left to right
  If children address <= D
      assign next hop ← children address
      break from loop
  End if
End For
Return next hop
```

The address of a node denotes the maximum address of all devices under it. A router assigns address to end devices as node_address – n . Where 1 <= n <= E_n. At any given point, the property of the node addresses is maintained and thus routing is achieved. For example, consider an end device joins the network at node 35. It is assigned an address of 34. The routing decisions would be at each node and checks under which child, can the destination address exist and forward accordingly as shown in figure 3.

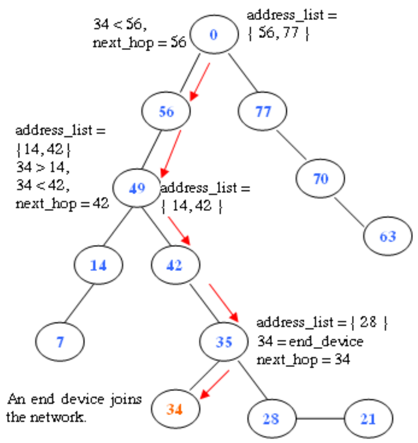


Figure 3. Addressing & Routing in the modified Hierarchical Tree.

b) The Central Server

The data obtained from the wireless mesh network is primarily stored in different staging tables in a server database and subsequently processed and analyzed by the software to get a real-time view and status of the monitoring zone. The data may be classified into two categories:

- Sensor data from sensors
- Periodic health beacons from Routers indicating their presence and battery status

The software system consists of the following modules:

i) **The Middleware**: It accepts the data through the serial port of the server and filters them for duplicates and is stored in an acceptable format in different tables depending on the type of data. It also accepts the control messages from the software system (central server or Internet), translates it into a proper format and is dispatched to the Gateway for the intended recipient.

ii) **System Configurator**: is used to map the mesh network components into the software system. The Mesh network is essentially a sensing zone with routers and actuators placed at different strategic locations.

iii) **Network Manager**: Allows the user to view status of the devices in the network (alive, dead, battery status etc.)

iv) **Sensor Monitor**: Allows the user to monitor the status of different sensor nodes in the sensing zone.

v) **Response Handler**: It enables the user to send context-sensitive control message/commands to any device (actuators) in the network in response to some critical event.

vi) **Report generator**: Allows the user to generate customized reports based on the available sensor data.

vii) **Query Manager**: This is a query-driven interface that allows the user to send a query to the mesh network asking for information about the status of sensor node(s).

viii) **Message Handler**: Allows the user to send control message to the network for actuation or sleep control.

c) Accessing through Web-Enabled Devices

A Web-based Monitoring system offers the flexibility to monitor the environment through the Internet, thus providing "anytime anywhere" visibility of the sensor data. Remote users can monitor through the control station and automated control or manual decisions can be taken. Other features like, Sensor monitoring, Response handler, Report generator, Query manager and Message handler are also available in the web-based version of the software. The components of the central software are shown in figure 4.

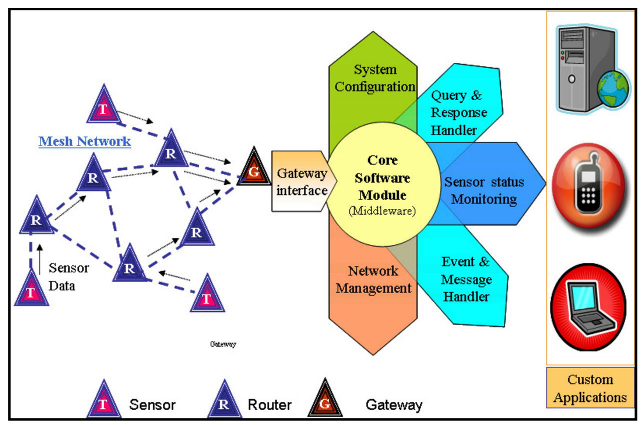


Figure 4. The system architecture

V. WIRELESS SENSOR NETWORK TESTBED

In this section we weigh the different technology options for a pollution monitoring testbed and provide a detailed account of the different aspects on the off the shelf sensors and the results of the implemented testbed.

i. Exploring different technology options for wireless sensor network

We have explored the different technology options for wireless sensor network like 802.11 and 802.15.4. It is identified that for real-time monitoring of pollution level in the environment, each sensor node needs to send its data (very low volume) at regular intervals to the central control station. Though it is possible to achieve the goal using 802.11 based technology, it is generally power-hungry and not suitable to operate in outdoor environments for a long period of time. Moreover, our application does not require the high data rate provided by 802.11. Thus, we have decided to explore low-power, low data rate 802.15.4 based ZigBee and non-ZigBee platform.

Each wireless sensor node in our system would be an IEEE 802.15.4 based RF module having interfaces to connect different types of pollution sensors. Since sensors and RF modules will be mostly battery-powered, in order to extend the battery life of those sensors as well as RF devices, we are exploring different sleep mechanisms. The devices can be made active (from the sleep mode) either periodically or on user-defined conditions (such as when the sensed data is beyond acceptable range). We have primarily selected products from Maxstream for this development. XBee and XBee-Pro OEM RF Modules from Maxstream are Zigbee/IEEE 802.15.4 compliant solutions that satisfy the unique needs of low-cost, low-power wireless sensor networks. The modules are easy-to-use, require minimal power and provide reliable delivery of critical data between devices. XBee-Pro yields 1.5 km LoS (line of sight) data communication range whereas XBee offers 100 m LoS data communication range. Our target is to implement a mesh network with XBee and XBee-Pro (figure 5) Modules and interfacing them with Sensors and Actuators.

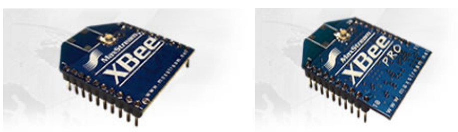


Figure 5. XBee/ XBee-Pro Zigbee OEM Wireless Modules

i. Pollution Sensors

The Wireless Sensor network based air pollution monitoring is all about monitoring the amount of pollutants in air wirelessly. We identify the pollutants normally present in air to be CO (carbon monoxide), CO_2 (carbon di-oxide), CH_4 (methane) and Volatile organic compounds (VOC). According to our proposed system, wireless devices with sensors attached will be put in different locations covering the entire area to be monitored. Sensor data will be captured by the respective sensors and will be sent wirelessly to the control station. If the control station is not within the radio range of the wireless sensor device, sufficient number of routers needs to be placed in between the sensor devices and the control station to transmit the data in multi-hop. On-line monitoring software at the control station will show the status of sensors in different areas so that the criticality in environment can be detected and necessary actions may be taken. The semiconductor sensors used are shown in figure 6 and their properties in table 1.

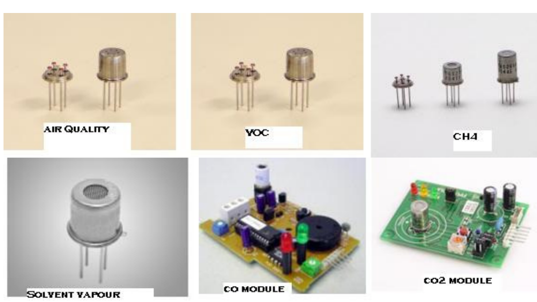


Figure 6. Pollution sensors

Table 1. Properties of Sensors used.

Sensor Type	Detectable gas	Range
TGS 2600	Air contaminants (smoke, fumes)	1-30 ppm
TGS 4161	CO_2	350-10000 ppm
TGS 2442	CO	30-1000 ppm
TGS 2611	Methane	500-10000 ppm

ii. Wireless Sensor Network Testbed

We have used 20 wireless sensor nodes, spread across four different buildings within our institute campus. A snapshot of testbed scenario is shown in figure 7. Two buildings are approximately 500 meters apart. Therefore, we have used multiple routers to collect sensor data from the buildings and send them to the control station which is approximately one km away from the deployment field. Each individual wireless sensor node can accommodate maximum of four different types of pollution detection sensors. For example, sensor "5678" can sense three parameters,: temperature, humidity and air contaminants, sensor "5679" consists VOC and solvent vapor sensors, "5681" measures CO and "5685" measures CH_4. We have also connected additional actuators like buzzers with some nodes in each location for generating alerts at the location in case of critical events. Our testbed is not only a typical wireless sensor network, but also serves as a wireless actuator network supporting two-way communication between the deployment and control station.

Figure 7. A snapshot of Testbed scenario

Each sensor node sends sensor data at a 10 second interval with its sleep cycle at 10 seconds. Once active (i.e. after completing a sleep cycle), the device waits for 3 seconds to warm up the heater coil (the sensors we used relies on heater coil to sense parameters) to enable the sensor reading to stabilize and then sends the observed sensor data over the network to the remote control station. Since the sensors draw significant power for each reading, therefore, even if proper sleep management technique is employed, node power drains out quickly. Therefore, for this experimentation, we have used mains to power the sensors, while the communication node is battery powered. To test the robustness of the IEEE 802.15.4 framework, the system was checked for the packet arrival variance for different sleep times of the sensors and is depicted in figure 8.

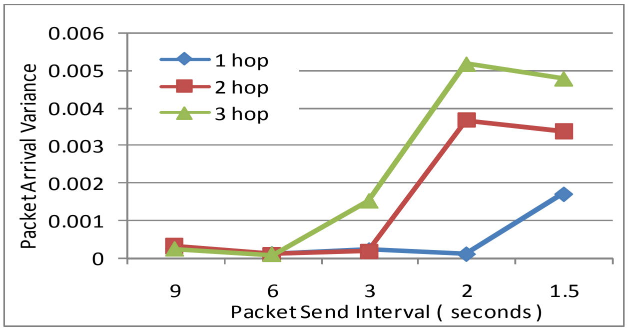


Figure 8. Variance in packet arrival rates.

The results show stable data communication performance for multihop routing when the packet send interval and thus the sleep time is 10 seconds. However, the requirement to wait for stable sensor data means that the life time of the battery operated nodes is affected. The actual measured power consumption is shown in table 2.

Table 2. Current Consumption Comparison

Parameter	Data Sheet Value	Measured Value
Tx current	27mA	53mA
Rx current	27mA	55mA
Sleep current	0.5 µA	2µA

Accounting for the sleep duration of 10 seconds, an active duration of 3 seconds and Tx Current equals the Rx current, the average current consumption works to (55 *3/13 + 2*10^-3*10/13) = 12.66mA. With standard AA batteries of 1400 mAHr, the life of a node is 110 hours.

Our testbed enables the user to change the sleep time of the sensors from the central server depending on the criticality of the situation. It can be achieved by sending a command from the central control station to the specific sensor in the format <<sensor node-id>> <<sensor type>> <<new sleep time>>. Since the target sensor may be in the sleep mode while the command is issued, therefore, we have devised a strategy to hold the command at the nearest router in the network. Since the network is static, the nearest router for a sensor node is also known to the system. Each sensor node after the sleep cycle will broadcast an "awake beacon" and a router, on receiving it, will send the pending "change sleep time" information accordingly. If a sensor shows a reading beyond a preset threshold the software will wait for a predefined time to evaluate whether the criticality is real or due to a spurious reading. If real criticality is detected then a command will be automatically generated by the system to sound the buzzer at that location.

Figure 9 shows the web based graphical user interface to monitor pollution data "anytime anywhere" using any web-enabled machine. In the top panel of the left-most pane of the GUI list of all wireless sensors nodes deployed in the field are shown. Middle panel shows the current sensor reading for each sensor associated with a single wireless sensor node. Bottom panel shows the continuous graphical representation of the trend and variation of readings for a selected sensor type. Map-based view panel shows the location and current readings of all the deployed sensors on the map of the deployment site. If the reading of a sensor crosses a preset threshold, the reading will be shown in red for quick identification of critical conditions and the location of the event so that proper control action can be taken. If for any reason (i.e., due to any fault or if it runs out of battery), a sensor node stops sending data, that can be detected at once from the map-based view, where that particular sensor will be shown in grey implying that it is deactivated.

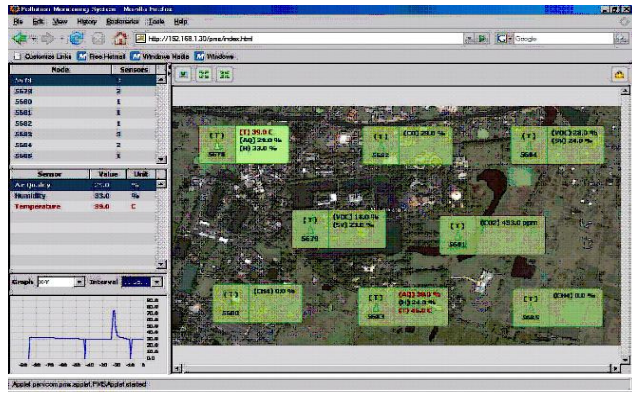


Figure 9. The GUI for the pollution monitoring system

VI. CONCLUSION AND FUTURE WORK

In this paper, we have shown a generic architecture to achieve the 'Internet of Things' paradigm and made a practical implementation using IEEE 802.15.4 based devices. We have provided detailed explanation of the various stages of the network setup and the aspects of selecting and configuring sensors for pollution monitoring. Contributions of our work include the generic architecture, the modified routing algorithm suited for the sensing system and the identification of the practical issues in the configuration and set-up of sensors for the mesh network. Specifically, research has to be directed at achieving low cost, low latency and low power sensors to achieve a truly usable pollution monitoring system. Till date, a large portion of the research has been directed at communicational aspects of the sensor network with little enhancements made to the actual sensor technology. The results confirm this proposition, where stable communication from the IEEE 802.15.4 based devices was achieved while the average lifetime deteriorated due to the latency requirement of the sensors.

There are a few aspects we would like to address in the future. The framework developed and implemented in this paper, gives the power of abstracting the underlying architecture at the gateway. For example, the sensor network can be an Ethernet based IP system or a host of WiFi Hotspots. The underlying structure and nature of the network does not affect the architectural framework. However, this abstraction requires a formalization of the services that must be offered at the sensor network and gateway interface. This set of services can only be determined when the entire end to end service requirements are ascertained. We, thus, would like to develop a structured approach and framework for identifying and designing the services at the interfaces.

References

Akyildiz, I. F., & Wang, X. (2005). A Survey on Wireless Mesh Networks. *IEEE Communications Magazine* .

Al-Karaki, J. N., & Ahmed, K. E. (2004). Routing Techniques in Wireless Sensor Networks: A Survey. *IEEE Wireless Communications* .

Cuomo, F., Luna, S. D., Monaco, U., & Melodia, T. (2007). Routing in ZigBee: benefits from exploiting the IEEE 802.15.4 association tree. *IEEE International Conference on Communications.*

Hammons, M. J., & Chisholm, G. (2006). Enabling Total Asset Visibility. *Defense Transportation Journal* .

Ho-In, J., & Yeonsoo, K. (2007). Efficient, Real-Time Short Address Allocations for USN Devices Using LAA (Last Address Assigned) Algorithm. *Ninth International Conference on Advanced Communication Theory.*

IEEE. (2003). *Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks.*

ITU. (2005). *International Telecommunications Union.* Retrieved February 17, 2009, from www.itu.int/internetofthings/

Jung, Y. J., Lee, Y. K., Lee, D. G., Ryu, K. H., & Nittel, S. (2008). Air pollution monitoring system based on geosensor network. *IEEE International Geoscience & Remote Sensing Symposium.* Boston, Massachusetts, U.S.A.

Ma, Y., Richards, M., Ghanem, M., Guo, Y., & Hassard, J. (2008). Air Pollution Monitoring and Mining Based on Sensor Grid in London. *Sensors Journal: Special Issue on Urban Environmental Monitoring* , Volume 8 3601–3623.

Peng, R., Mao-heng, S., & You-min, Z. (2006). ZigBee Routing Selection Strategy Based on Data Services and Energy-balanced ZigBee Routing. *IEEE Asia-Pacific conference on Services Computing.*

Perkins, C., & Royer, E. (1999). Ad-Hoc On Demand Distance Vector Routing. *IEEE Workshop on Mobile Computing Systems and Applications.*

Pujolle, G. (2006). An Autonomic-oriented Architecture for the Internet of Things. *IEEE John Vincent Atanasoff 2006 International Symposium on Modern Computing.*

Schmid, T., Dubois-Ferrière, H., & Vetterli, M. (2005). SensorScope: Experiences with a Wireless Building Monitoring Sensor Network. *Workshop on Real-World Wireless Sensor Networks (REALWSN).*

Werner-Aller, G., Swieskowski, P., & Welsh, M. (2005). MoteLab: A Wireless Sensor Network Testbed. *Fourth International Conference on Information Processing in Sensor Networks (IPSN).*

Wheeler, A. (2007). Commercial Applications of Wireless Sensor Networks using ZigBee. *IEEE Communications Magazine* .

ZigBee Alliance. (2004). Retrieved February 17, 2009, from ZigBee Alliance: www.zigbee.org