Framework for a Self-managed Wireless Sensor Cloud for Critical Event Management

Nithya G. Nair, Philip J. Morrow, and Gerard P. Parr

School of Computing and Information Engineering, University of Ulster, Coleraine, UK
gopalakrishnannair-n@email.ulster.ac.uk,
{pj.morrow,gp.parr}@ulster.ac.uk

Abstract. Wireless sensor networks (WSNs) can be widely used for managing various scenarios existing in social, industrial and numerous environmental management systems. They have been widely used in environmental monitoring and management applications and have also found application in disaster management scenarios. One of the greatest problems faced by the scientific community in organizing data collection through sensor networks in areas of disaster is the disorder and destruction brought about in the communication systems prevailing in such situations. In this paper, a scientific study of the various scenarios that could occur post-disaster and the various housekeeping functions each sensor node would adopt as part of the self management requirement is provided. We also present a sensor task management framework that could be implemented to provide a low energy consuming, reliable network for WSNs deployed for critical infrastructure management.

Keywords: self-managed wireless sensor cloud, critical event management.

1 Introduction

The demand for wireless sensor networks has increased in recent years due to advances in technology, which has led to extensive research into the field of critical infrastructure monitoring/management. One of the greatest problems faced by the scientific community in organizing data collection through sensor networks in areas of disaster is the disorder and destruction brought about in the communication systems prevailing in such situations like tsunamis, typhoons, earthquakes etc. With the increasing popularity of the emergent concepts of cloud technology, it would be quite beneficial to have the data recorded by wireless sensor networks deployed across a large area, made accessible from anywhere in the world.

Some of the most desired characteristics of such a network could be stated as follows: it performs energy efficient operations to increase the lifetime of the network and it possesses auto configuration capability as there may be a number of nodes that could join or leave the sensor cloud as the calamity sweeps across the region. When a WSN is deployed over a large geographical area, it assumes the format of a multi-hop communication network. It should also become data centric to avoid data loss due to node failures. The other characteristics that would be desirable are higher quality of

F. Martins, L. Lopes, and H. Paulino (Eds.): S-Cube 2012, LNICST 102, pp. 15–29, 2012.

service, higher fault tolerance, scalability of the network, lower power consumption/ better power management, better security, programmability, ease of maintenance and lower costs.

The focus of this paper is on Wireless Sensor Networks that are deployed to monitor and sense critical data. These could be event monitoring applications, for example: sensing seismic vibrations to warn of the possibility of a tsunami or landslide or flash floods and manage scenarios of post quake effects. Such WSNs could be deployed in an area with limited access and may be prone to drastic topological changes such as landslides and flood displacements. Since a sensor network is being utilized for critical applications, data being sensed would be of high importance. This requires the data being transmitted by the WSN to be highly reliable and error-free. The area to be sensed may be geographically inaccessible which would make it a top priority to make the network self-manageable and have an extended life time of operation.

A WSN deployed for monitoring purposes is susceptible to the situation of chaos as a result of the occurrence of a disaster. It would have to reconfigure using autonomic self-intelligent methods from a chaotic disarranged sensor distribution to form a robust network as was previously in place. This robust network is expected to carry out data collection and also provide reliable data transmission and delivery to a sink node. These functions have to be performed using the lowest energy consumption possible to extend the lifetime of the network. Hence, the important features a WSN monitoring a critical infrastructure would require are a) robust management protocols b) efficient power management and c) reliable data management and delivery.

The contribution of this project would be an efficient task management system with appropriate task suppression mechanisms that could contribute to the reliability of the deployed sensor network.

In this paper we are focusing on the specific issues dealt with when designing a framework for a self-managed wireless sensor network. The remainder of the paper is organized as follows: Section 2 provides a brief overview of related work done on the elements that contribute to the design of WSNs; Section 3 gives details on the various management aspects required in the design of a self-managed WSN; Section 4 gives an outline of the various scenarios that could arise in the case of a disaster and Section 5 concludes the paper and outlines further work that has to be carried out.

2 Related Work

Wireless sensor networks used for monitoring applications can be prone to various events like nodes dying, topology reconfiguration, loss of connection to the base station etc. Depending on various events that could occur during a WSN deployment, elements that could be said to contribute to the design of a WSN are protocols, data fusion / aggregation techniques, topology management and methods of deployment.

Protocols that are used in WSNs can be categorized as network / routing, transport and MAC protocols. There have been several network protocols designed to address the issue of connectivity establishment. According to the authors of [1], LEACH

(Low Energy Adaptive Clustering Hierarchy) is a popular choice among the various network protocols as it is flexible and self-adaptive by nature and uses a TDMA (Time Division Multiple Access) technique for transmission at the Management Node (Gateway) level. This makes it more energy efficient. Communication between nodes within a cluster is done through a CDMA (Code Division Multiple Access) to prevent interference between neighboring node transmissions. This protocol is said to be robust as node failures would not affect the overall network connectivity. The reason that this protocol would not be fit for implementation in the application mentioned here is because the initial setup phase of this protocol is quite energy consuming and has a large overhead when forming clusters. The deployment scenario discussed in this paper would require the network to be a multi hop communication infrastructure. LEACH is single hop communication protocol which would not be suitable for our purposes. It is understood that hierarchical/cluster based routing can be used in one-to-many and many-to-one communication scenarios. Hierarchical routing also helps reduce the overall energy consumption thereby increasing the lifetime of the network and providing scalability to the network [2]. These features make hierarchical routing protocols a favorable option to be used for the application mentioned here.

Due to limited bandwidth and the convergence nature of upstream data, two major issues to be dealt with would be packet loss and congestion within the WSN. Transport protocols are supposed to provide orderly transmission, flow and congestion control and QoS guarantee [3]. When dealing with WSN they need to provide a simple connection establishment process which has reduced transmission delays. Cross-layer optimisation would be an added advantage as the transport protocol would recognise if packet loss was caused due to congestion or route failure.

However, in [4] the authors indicate that TCP based protocols are not suitable for sensor networks as they do not consider energy conservation as a priority. The authors suggest that the Event-to-Sink Reliable Transport (ESRT) protocol supports congestion detection and also provides an acceptable level of reliability to the WSN and could be used for critical infrastructure monitoring.

When considering MAC protocols, random access protocols are considered suitable for WSN because of their self-organising nature. Since the project undertaken here is an event based application, duty cycle based and hybrid protocols could also be considered for use. ER-MAC [5] is an obvious choice as this protocol was designed for emergency response applications and unlike fixed protocols, provides scalability and flexibility to the network. The PACT algorithm [6] is another possible choice as it provides low latency and better QoS. It uses an adaptive duty cycle that depends on the traffic load thus increasing the network life time.

Data management in the case of WSNs would involve the task of acquiring data from sensors, storage of the collected data and efficient transmission of data to the end users [7]. It should be energy efficient, scalable and robust to failures as well. The sensors can be considered as distributed storage points which could be queried on demand. Efficient data transmission would require appropriate levels of data aggregation in order to preserve energy within the sensor node. Incorporating data fusion/aggregation techniques would provide enhanced system reliability, robustness due to redundant data being available, improved bandwidth utilization, lower energy consumption and extended network lifetime [8][9].

In [10] the authors explain topology management as the capability of the network to maintain connectivity across the network. Topology control is essential to increase the lifetime of a network by helping reduce the energy expenditure. It is also useful in reducing radio interference, improving the efficiency of MAC and routing protocols and also improves the robustness of the network and thereby the reliability of the network, quality of connection and also the coverage and have an influence on the scalability of the network. The authors in [11] suggest that these algorithms trade energy conservation for increased routing latency. They propose an algorithm for WSNs in failure-prone environments called adaptive Naps which conserves on energy and also is able work in a distributed manner thereby making it adaptive to node failures which was the disadvantage in the Naps algorithm.

Regarding deployment methods for a WSN, there are two types of deployments. They are deterministic and random [12]. A deterministic deployment is required when the cost of the sensor is high and their operation is significantly influenced by the location, e.g.: underwater applications (imaging sensors) or indoor applications. Random deployment is used where the cost of nodes is not of high priority and also where the environment is harsh, e.g.: battle field or disaster areas. The factors on which node deployment depends are: area coverage, network connectivity and longevity and data reliability. Other deployment strategies available depend on the functions of the node as well. In order to achieve maximum area coverage, random deployment may be a feasible option which is widely adopted in applications like target tracking etc.

Thus, we can say that the protocols that could be adopted for use within this project can be summarised using the protocol stack described by the authors in [3] as in Figure 1:

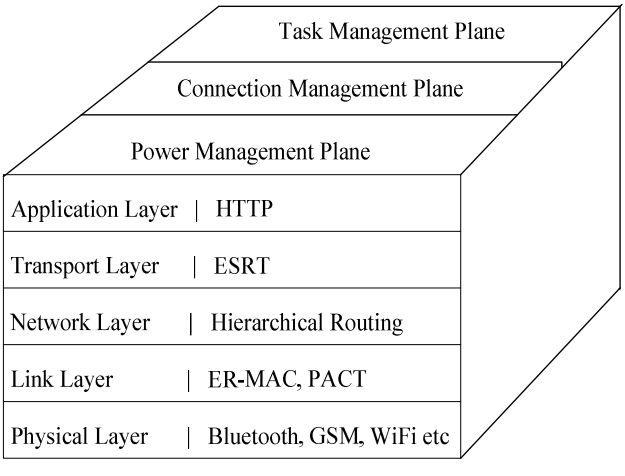

Fig. 1. Protocol stack adapted from [3]

There are several case studies where the authors provide WSN architectures for specific applications. In [13] the authors discuss an architecture that is self configurable in terms of topology reconfiguration in an energy efficient manner. This

particular architecture does not talk about data management which is also vital in applications like monitoring of critical infrastructures. The authors of [14] describe an architecture that deals with a low power network management but have not given enough importance to data reliability and management. Another case study for a WSN used for disaster management is given in [15]. This network is used to manage rescue operations after occurrence of a large scale disaster. Here the nodes are randomly deployed by individuals at their home, office etc. In this network the end user is mobile and so can collect data from the nodes without needing to have an extensive setup for connection management and data management etc.

3 Task Monitoring Framework for WSNs

A deployed WSN may experience events like topology changes or nodes dying or malfunctioning etc. The scenarios that arise as a result of these events are explained in Section 4. An 'event' could be a change in any parameter used by the node for initial setup. For example, movement of the node, loss of connection and depletion of energy or memory reserve. The 'event' could also be described as an external event that would be picked up by a sensor on the node. It could be any event like change in temperature, vibration, detection of movement etc. The events mentioned above would require the WSN to reorganize and adopt functionalities that would attempt to provide network connectivity to a majority or all nodes present in the WSN. In order to achieve this, various management functions have to be executed. The management of a WSN could be influenced by sensor tasks that are needed to be carried out within the network.

These tasks could be classified according to the modes of operation of the sensor, the interaction among sensor nodes, mobility of the sensor which in turn would deal with the resource management within the node, and also the priority of the tasks being executed.

3.1 Sensor Tasks

Figure 2 provides a classification of the various tasks that could be performed by the node. Modes of operation of a sensor node could be either autonomous (stand alone) or connected (networked to base station or adhoc). Events may occur that could cause the connectivity within the network to change and hence the node may find itself in various situations such as:

- The sensor has a connection to the base station (networked)
- The sensor is part of a group/network of sensors that would have lost contact with the base station (ad-hoc mode)
- The sensor is alone, disconnected from the network (i.e. in standalone mode)

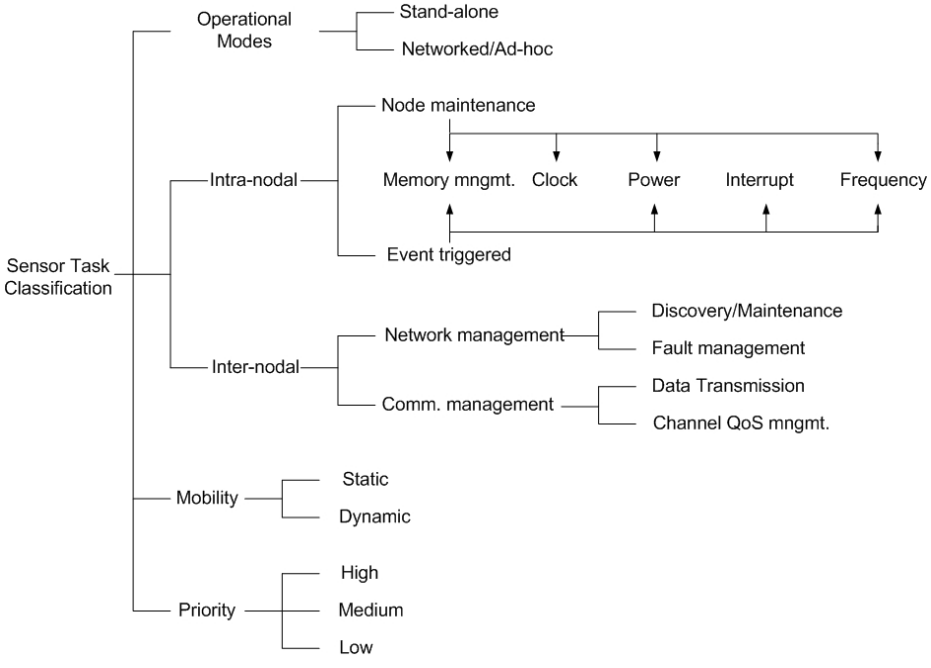

Fig. 2. Sensor task Classification

Depending on the interaction of nodes when the execution of a task takes places, they could be divided into intra-nodal and inter-nodal tasks. Intra-nodal tasks could further be divided into node self management / maintenance tasks and event triggered tasks. The self-management/maintenance tasks carried out by the node are memory management, clock synchronization tasks, power management and task-frequency management. The event triggered tasks would overlap with most of the self-management tasks performed by the node but would also include interrupt management tasks.

Inter-nodal tasks require the node to have interaction with other nodes in the form of polling and information transfer etc. These are further divided into network management tasks and communication management tasks. The network management tasks involve tasks required to be performed for discovery and formation of the sensor network and also maintenance of the network. The network management tasks also deal with the fault management of the sensor network setup. The communication management tasks deal with the data transmission and also the channel QoS management.

The sensor tasks could further be classified based on the mobility of the sensor i.e. the resource management tasks that would be required if a node is static or in a dynamic environment. The resource management tasks would overlap with the intra-nodal tasks as it would be related to tasks that maintain the node thereby attempting to extend the node's lifetime.

The sensor tasks could also be classified according to priority i.e. high, medium and low. This is provided to put in place a task suppression system which would be

essential to avoid overloading the node CPU and also wasting the limited resources available to the node. It would also provide a level of legitimacy to the alerts that would be created by the sensor network.

3.2 Task Management Groups

The various sensor tasks have been discussed above and can be further categorized into management groups. Within our framework we propose the following systems / groups: software management, power management, sensor assessment, buffer management, data management and task frequency management groups. A $P_{i, j}$ matrix is used to represent the various parameters within each management system where *i* is the management group and *j* is the state.

Software management refers to the various states that a sensor node could adopt depending on the situation it finds itself in. Three states can be defined for node operation as follows:

Active state ($P_{1, 1}$):	All systems are functioning at full power
Idle/Standby ($P_{1, 3}$):	The node is in low power state where the transceiver is turned off to save power.
Sleep/ Off ($P_{1, 2}$):	The node is completely inactive for a set time period. It would poll for anomalies in between the idle cycles to decide if it needs to wake up and change state or not.

The Active state is executed for data processing and transmission. The Sleep/Off state is executed when the node is not in use for a particular cycle. The Idle/ Standby state is adopted when the node is sensing but no data processing or transmission is needed. The energy levels that are consumed during these states vary and the switching between these states could be used in the future to formulate an optimal energy conservation technique.

Power management is required by the sensor node to conserve battery power by means of regulating node functions thereby extending the lifetime of the sensor network as a whole. Sensor node battery level thresholds could be defined before deployment. The thresholds could be defined as T1 for normal level and T2 for critical level (Fig. 3).

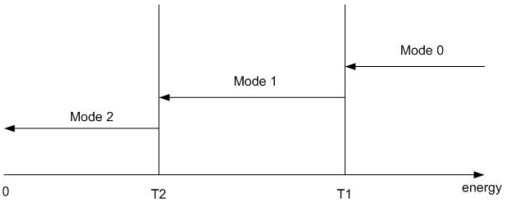

Fig. 3. Power thresholds and modes for power management

Depending on the thresholds we could define three modes of operations. They are:

Mode0 ($P_{2,1}$): sensor node functioning.
Mode1 ($P_{2,2}$): send a signal to gateway about "time to extinct" period to cease message bridging functions undertaken.
Mode2 ($P_{2,3}$): sensor node in critical level and so emergency measures adopted.

'Time to extinct' period is the time that would be taken for the node's battery level to reach critical level after which most of the node's regular functions would be put on hold.

Sensor Assessment is a house keeping function used by the node to decide whether the data obtained from the sensors is reliable i.e. if the error tolerance is within the acceptable band. The band is to be pre-determined before deployment. Three levels that can be used to determine if the data received is reliable or not, are:

Sensor intact ***($P_{3,1}$):*** data collected during each cycle is within the acceptable margins.
Sensor lost ***($P_{3,2}$)***: data received is garbled continuously or no data received by the node from the sensor.
Sensor doubtful ***($P_{3,3}$)***: if there are occurrences of garbled data within a set of observations the sensor state would be at a level where the sensing function would be considered doubtful

Buffer Management deals with memory management i.e. the management of memory levels within a sensor node. Data management strategies on the received and stored data could be adapted when a particular level is reached in the memory storage. This level could be calculated according to a *relinquishing ratio* (Fig. 4) which depends on the rate at which data is sent to a safe location from the current node. This could also be set at the pre-deployment period, depending on the characteristic of the predicted use and event occurrences.

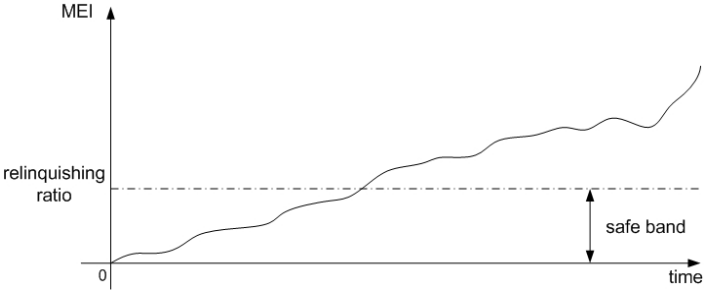

Fig. 4. Depiction of relinquishing ratio relative to Memory engagement index (MEI)

Memory engagement index (MEI) is defined as the ratio of the buffer occupied by data and the total size of buffer.

The *relinquishing ratio* is the MEI value which distinguishes levels below it as the safe band. This ratio could be adjusted according to the different states in which the sensor would operate.

Data management is closely related to buffer management in terms of *data sieving*. Data sieving refers to data quality management of the sensed data. Data quality could be divided into three bands. i.e., Positive faults, likely to be a fault and acceptable data.

$$\text{Data quality bands} \quad \begin{cases} \text{DQ1} \Rightarrow \text{positive fault } (P_{4,1}) \\ \text{DQ2} \Rightarrow \text{likely to be a fault } (P_{4,2}) \\ \text{DQ3} \Rightarrow \text{acceptable data } (P_{4,3}) \end{cases}$$

When the data received by the sensors falls within DQ1 or DQ2 it is discarded and if it is within DQ3 it would be saved within the node memory, if MEI is favorable. This affects the buffer management and energy consumption. It contributes to energy conservation by avoiding unnecessary processing of data that does not fall in the acceptable range.

Data management also includes housekeeping functions for *data communication* from the sensor node to the base station. A healthy communication would normally work on a fixed range of transmission frequencies. When an event occurs and loss of communication takes place, the node can switch over to recovery mode i.e. to achieve maximum energy conservation and data preservation. In the case of a post disaster communication channel loss, we could adopt an exponentially behaving *rescue-cling algorithm* (Fig. 5). The purpose of this algorithm would be to enable a ping sequence for other nodes with gradually decaying polling frequency on repeating failures to avoid excess use of power. In case an acknowledgement is received, the node would maintain the normal repetition rate at which it sends messages from then on. If no acknowledgement is received then the frequency with which polling is done decreases exponentially.

Fig. 5. Frequency change when using *rescue-cling* algorithm

Data aggregation is also an operation which forms part of data management which occurs during failure mode. When the sensor is isolated, a failure mode flag is set and a dynamic interval sampling frequency (i.e. sensing frequency) may be adopted. In the case of a gateway, compression techniques may be applied to the data that is being transmitted to the base station.

Task Frequency Management: All repetitive processes within the sensor nodes will be governed by certain frequency properties. Major domains for this house keeping function are the sensor data sampling functions and the data transmission cycles. Variations in these rates of operations depend on the various situations in which the sensor nodes find themselves.

3.3 Sensor Task / Management Relationships

As discussed above, various management groups have been defined along with several parameters. Most of the sensor tasks discussed earlier in the section overlap between the various management groups. Table 1 provides the various sensor tasks and management groups and the overlap of tasks with the management groups.

Table 1. The placement of the various sensor tasks in the different groups

Sensor Tasks \ Management Groups		Software	Power	Buffer	Data	Sensor Assessment	Frequency
Modes of Operation	Stand alone	✓	✓	✓	✓		✓
	Networked/ Ad-hoc	✓	✓			✓	
Intra-node	Memory			✓	✓		
	Clock						✓
	Power		✓				
	Interrupt	✓		✓		✓	✓
	Frequency		✓	✓		✓	✓
Inter-node	Topology Discovery/ Maintenance	✓	✓		✓		✓
	Fault Management				✓		✓
	Data transmission		✓	✓	✓	✓	✓
	Channel QoS management		✓		✓	✓	✓
Mobility	Static	✓	✓	✓	✓		✓
	Dynamic	✓	✓	✓	✓	✓	✓
Priority		✓	✓	✓	✓	✓	✓

A ‘tick’ indicates that a sensor task as listed on the left side of the table has a relationship with a management group listed on the top of the table. Consider the case of an interrupt task. Within a node, several interrupts maybe present that could influence the software of a node, buffer management and frequency management. The sensor assessment would have an influence on the interrupt tasks i.e. by determining if the sensor is functioning or not, an interrupt could be executed which would in turn influence one of the other management groups it is related to.

4 WSN Scenarios Following an Event Occurrence

As discussed in Section 3, there are various tasks that could be executed when an event occurs. This is in relation to the various situations in which the node would have to operate i.e. the node could find itself disconnected from the network or it could find itself connected to a cluster of nodes that has lost connection to the network.

We investigate a number of different scenarios that may be dealt with by a wireless sensor network when it adopts operations similar to adhoc wireless networks [16]. When an event occurs, the network infrastructure that was previously built could get partially or fully disorganized. This could result in a loss in communication/coverage of an area leading to loss of vital data that is required for analysis by the user.

An ideal world situation is considered initially to provide a baseline for comparison with the subsequent scenarios.

4.1 Ideal World (Scenario 1)

In an ideal scenario, as illustrated in Fig 6, there are no factors that could lead to loss in performance or reduce the reliability of the deployed wireless sensor network.

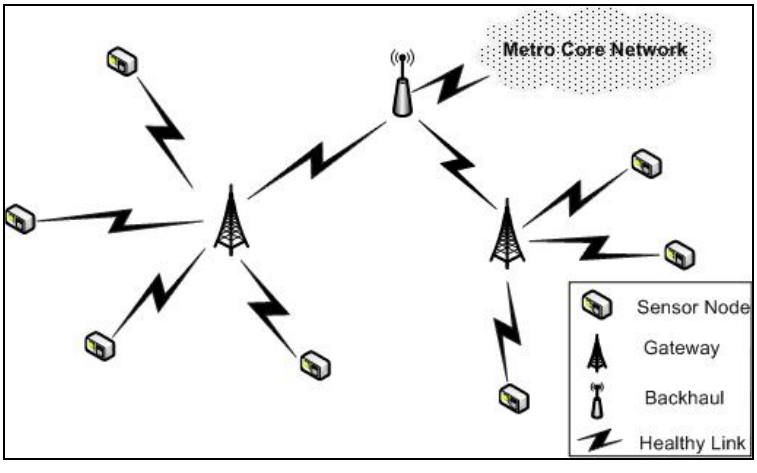

Fig. 6. Illustration of an Ideal World Scenario

There would be no possible loss of connectivity, no shortage of memory for vital data storage within the nodes, and energy within the nodes would not be exhausted. Thus, node failure due to energy depletion would not occur. The sensors that form part of the WSN could be deployed randomly or in a pre-deterministic manner and are connected to the sink or base station for operations. In the ideal world or near ideal world scenario, disruption to communication links would not occur and also minimal power management and data management techniques need to be adopted. Polling frequency can be varied as and when required.

When an event is triggered, the polling frequency is increased initially to determine if the event was a random occurrence or not. If an event trigger is received consistently over a period of time then an alarm is sent upstream to the core network and data is gathered for further analysis by the users.

4.2 Real World Scenarios

When dealing with real world scenarios, anything can go wrong. We look at two main types of scenarios which can be divided further into two subsections according to the

housekeeping function they have to adopt to form a robust reconfigured network. Each of these scenarios give rise to a number of issues that must be considered for the continuous operation of the wireless sensor network and also extending the lifetime of the nodes and in turn the network as a whole. The issues that arise in the real world scenarios, mainly topology reconfiguration, power management and data management are also discussed.

Scenario 2: Dynamic Topology Where There Is a Break in the Connection between the Node and Gateway but the Connection to Backhaul Still Exists (Fig 7)

Case (i): A sensor node loses its connectivity to the gateway/base station due to change in network structure. There are several housekeeping functions that would need to be undertaken. It would have to reduce its polling frequency to conserve power and storage space for long periods of isolation, until it can reestablish the communication link with the gateway and transmit the data it has collected. Depending on the energy reserves it could scan its surroundings, to check for neighboring gateways or sensor nodes within its transmission range, to reconnect with the network.

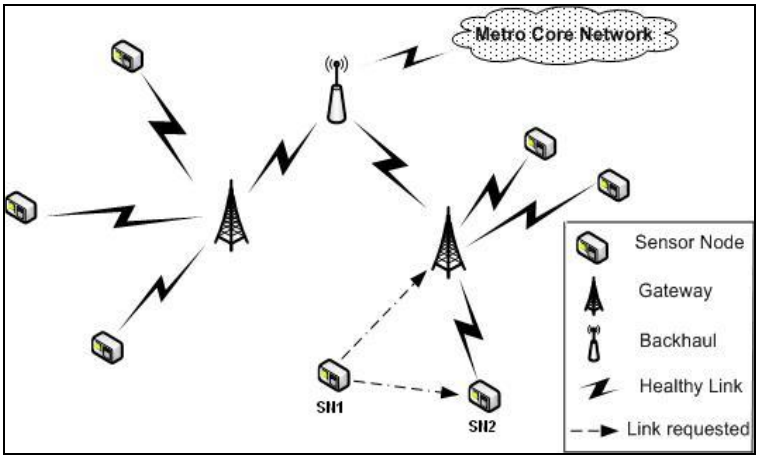

Fig. 7. Illustration of a scenario where the sensor would lose communication to the network

When an event occurs, the isolated sensors would increase their polling frequency and store the sensed data in the expectation that the link to the network would be restored at a later time. The situation that the node could experience in the isolated state would be depletion of memory. If such is the case, the oldest data could be deleted i.e. overwritten to store the newer data. Another option is to prioritize the data stored and the lower priority data could be overwritten if a high priority data is sensed and recorded. The priority levels of the data would have to become user defined using certain means of threshold setting etc.

Data aggregation would be necessary for the transmission of the data collected as it could be sometime before the communication link is re-established with the network and there may be lots of data to be transmitted. Attempting to transmit all the data without performing aggregation would result in depletion of the nodes energy

reserves. It would consume the energy and storage reserves of the gateway or node with which it would reestablish connection as they would have their own data to transmit and also other sensors connected to them as well.

When considering the energy aspect of the node, when dealing with an emergency situation, if the energy reserve in the node is severely diminished, it would forgo the connection to the gateway and opt for a shorter range communication (Bluetooth etc. or IR) as and when it finds a suitable neighbor node.

Case (ii): The housekeeping functions that have to be adopted by a sensor node and the corresponding gateway when an isolated node gets into transmission range and requests a link to send the data gathered upstream towards the sink.

In the case when an isolated sensor node (SN1) would request a connection to the network through either a gateway or a sensor node connected to the network (SN2), there are certain housekeeping functions that need to be performed by the gateway or SN2 to accept SN1 into the network. Power management would be of importance if the connection is being established between SN1 and SN2 as SN2 will have to maintain connection with the gateway as well as cater to SN1 henceforth. Data aggregation would be required as SN2 would have its own gathered data to send upstream to the sink. If SN1 is able to establish connection with the gateway then the gateway would have to do further aggregation of the received data to accommodate the data received from SN1 and also let the other nodes in its vicinity know about the new addition i.e. SN1 to the network.

Scenario 3: The WSN Is Subjected to a Dynamic Topology Where the Connection to the Sink Is Lost to a Gateway (Fig 8)

Case (i): When a gateway loses connectivity with the sink/ backhaul or the bridging gateway. This case is slightly similar to the one where a sensor node loses its connection to the gateway. The main difference here is the level of aggregation required when transmitting the sensed data upstream towards the sink.

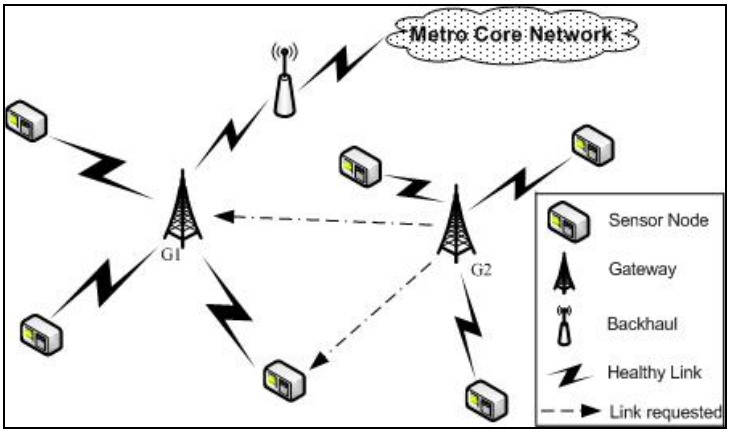

Fig. 8. Illustration of the scenario of a gateway attempting to reestablish connection to the backhaul through a bridging gateway or a sensor node

In this scenario when an event occurs and a gateway loses connectivity to the network, data management and in turn energy management becomes an issue. Data aggregation would have to be performed at a higher rate at the sensor node level and gateway level. Based on the residual energy of sensor nodes, some of them would be assigned the task of scanning their vicinity for other sensor nodes or gateways while others would go on with their task of sensing.

Case (ii) The housekeeping functions that have to be adopted by a sensor / gateway (G1) when a gateway with no communication link (G2) to the sink requests to reestablish a connection to the sink.

In this case, G2 request a link to either G1 or a child sensor node of G1. As with scenarios discussed earlier, G1 or a sensor node within the cluster of G1 would have to perform power management to compensate for the addition to the network. Data aggregation would have to be at a higher level as data of an entire subnet would have to be transmitted upstream through G1.

From the scenarios discussed here, the main issues identified are power management in terms of connectivity management and data management. In a post-event situation, when certain links to backhaul network is broken, the scattered network could reconfigure and form adhoc clusters which at some stage could reestablish connection to backhaul network. When connection is reestablished with sink, data management would have to be taken into consideration at that stage. Therefore, to build a WSN framework that is robust and energy efficient, protocols would have to be adopted that would need to provide connectivity management and data management with energy efficiency as an important feature.

5 Conclusion and Further Work

We are proposing a framework for a task monitoring system that provides details on the various sensor tasks that are executed when an event occurs. We discuss the different management groups into which the sensor tasks can be organized. Also discussed are the various scenarios that a node could find it-self in after an event occurs. An experimental test bed with Libelium Wapmotes is being used to create a multi-hop network topology which could be used to emulate the scenarios mentioned in Section 4. The experiments would be set up such that each of the scenarios could be created and some of the tasks could be monitored and thus create benchmarks for a metric of the costs for the various tasks for any sensor network applications.
The costs that are associated with the various tasks could be: 1) energy consumption when executing a task, 2) bandwidth consumption during task execution, 3) load on the CPU for a task, 4) memory required by a task, and 5) time taken to execute a task

Along with designing a sensor task monitoring system, an optimised network management protocol, a data management system and an energy management protocol need to be provided as there are very few available for a sensor network that would be used in critical infrastructure monitoring.

Acknowledgement. We would like to acknowledge the University of Ulster for providing a VCRS studentship as part of this IU-ATC (India-UK Advanced Technology Center of Excellence in Next Generation Networks) project.

References

[1] Frye, L.: Network management of a wireless sensor network, pp. 1–13 (2007)

[2] Lotf, J.J., Hosseinzadeh, M., Alguliev, R.M.: Hierarchical routing in wireless sensor networks: a survey. In: 2010 2nd International Conference on Computer Engineering and Technology, pp. V3-650–V3-654 (2010)

[3] Zheng, J., Jamalipour, A.: Wireless sensor networks: a networking perspective, vol. 2008, p. 489. Wiley-IEEE Press (2009)

[4] Wang, C., Sohraby, K., Hu, Y., Li, B., Tang, W.: Issues of transport control protocols for wireless sensor networks. In: Proceedings of 2005 International Conference on Communications, Circuits and Systems, pp. 422–426 (2005)

[5] Sitanayah, L., Sreenan, C.J., Brown, K.N.: ER-MAC: A Hybrid MAC Protocol for Emergency Response Wireless Sensor Networks. In: 2010 Fourth International Conference on Sensor Technologies and Applications (SENSORCOMM), pp. 244–249 (2010)

[6] Ali, M., Böhm, A., Jonsson, M.: Wireless Sensor Networks for Surveillance Applications – A Comparative Survey of MAC Protocols. In: 2008 The Fourth International Conference on Wireless and Mobile Communications, pp. 399–403 (2008)

[7] Cantoni, V., Lombardi, L., Lombardi, P.: Challenges for Data Mining in Distributed Sensor Networks. In: 18th International Conference on Pattern Recognition (ICPR 2006), pp. 1000–1007 (2006)

[8] Aguilar-Ponce, R., McNeely, J., Baker, A., Kumar, A., Bayoumi, M.: Multisensor Data Fusion Schemes for Wireless Sensor Networks. In: 2006 International Workshop on Computer Architecture for Machine Perception and Sensing, pp. 136–141 (September 2007)

[9] Chen, Y., Shu, J., Zhang, S., Liu, L., Sun, L.: Data Fusion in Wireless Sensor Networks. In: 2009 Second International Symposium on Electronic Commerce and Security, pp. 504–509 (2009)

[10] Gengzhong, Z., Qiumei, L.: A Survey on Topology Control in Wireless Sensor Networks. In: 2010 Second International Conference on Future Networks, pp. 376–380 (2010)

[11] Frye, L., Bigrigg, M.W.: Topology Maintenance of Wireless Sensor Networks in Node Failure-prone Environments. In: 2006 IEEE International Conference on Networking, Sensing and Control, vol. 15213, pp. 886–891 (2006)

[12] Younis, M., Akkaya, K.: Strategies and techniques for node placement in wireless sensor networks: A survey. Ad Hoc Networks 6(4), 621–655 (2008)

[13] Asim, M., Yu, M., Mokhtar, H., Merabti, M.: A Self-Configurable Architecture for Wireless Sensor Networks. In: 2010 Developments in E-systems Engineering, pp. 76–81 (September 2010)

[14] Pogkas, N., Karastergios, G.E., Antonopoulos, C.P., Koubias, S., Papadopoulos, G.: Architecture design and implementation of an ad-hoc network for disaster relief operations. IEEE Transactions on Industrial Informatics 3(1), 63–72 (2007)

[15] Cayirci, E., Coplu, T.: SENDROM: Sensor networks for disaster relief operations management. Wireless Networks 13(3), 409–423 (2006)

[16] Gopalakrishnan Nair, N., Morrow, P.J., Parr, G.P.: Design Considerations for a Self-Managed Wireless Sensor Cloud for Emergency Response Scenario. In: The 12th Annual Post Graduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting, PGNet 2011 (2011)