Ontology-driven Adaptive Sensor Networks

Sasikanth Avancha Chintan Patel
Anupam Joshi
Department of Computer Science and Electrical Engineering,
1000 Hilltop Circle, University of Maryland Baltimore County, Baltimore, MD 21250
{savanc1, cpatel2, joshi}@cs.umbc.edu

Abstract

A wireless sensor network deployed in an area of interest is affected by variations in environmental conditions associated with that area. It must adapt to these variations in order to continue functioning as desired by the user. We present a novel, two-phase solution to the wireless sensor network adaptivity problem. In the first phase, nodes in the network, organized as clusters, execute an efficient algorithm to dynamically calibrate sensed data. Each node provides its current energy level and the state of each on-board sensor to a cluster-head. In the second phase, each cluster-head executes an efficient, ontology-driven algorithm to determine the future state of the network under existing conditions, based on information received from each sensor node. We describe an example application scenario to show how our two-phase solution can be employed to enable a real-world wireless sensor network to adapt itself to variations in environmental conditions.

1. Introduction

The potential applicability of ad-hoc, wireless sensor networks (WSNs) in a variety of domains, ranging from simple distributed monitoring (e.g., habitat and environmental monitoring) to complex surveillance (e.g., battlefield surveillance, homeland security) [4, 6] has fueled research interest in this area. In the near future WSNs are expected to consist of sensor nodes containing on-board micro-electromechanical systems (MEMS) as sensors, which we expect to also be remotely programmable.

In security-related applications, such as battlefield surveillance and homeland security, users of WSNs require a high level of accuracy (close to 100%) and may specify tight accuracy bounds for all data reported by the network. For example, military personnel obtaining information from a WSN deployed in a battlefield must be assured that the network accurately reports the presence or absence of the enemy assets in the area. If the WSN cannot report data at user-specified accuracy levels or within the bounds, then the data would be considered useless.

WSNs consist of sensor nodes of various capabilities, including base stations (BS) and powerful, "rich uncle" (RU) nodes, which in turn consist of one or more types of sensors which perform the sensing function. The functioning of each type of on-board sensor is affected by external environmental conditions, such as temperature, pressure and humidity. Additionally, each sensor node is affected by its own internal environment consisting of variables such as energy level and available memory. In both these cases, accuracy of data reported by the sensor node is adversely affected. Thus, there is a need for sensor nodes to adapt to variations in their environment, if possible, and restore data accuracy levels to their original values.

An additional issue in security-related applications and hostile situations is that the entire WSN is affected by the prevailing security conditions in the area of deployment. Thus, the WSN must adapt itself as a whole to variations in security conditions in addition to environmental conditions so that it can continue to perform its job as desired by the user. This issue will be addressed in future work.

We address the **wireless sensor network adaptivity problem** associated with an established WSN and propose a two-phase approach as a solution.

In the first phase, sensor nodes possessing moderate computational and storage capabilities adapt themselves to variations in environmental variables by dynamically calibrating data from on-board sensors (Section 4). (We refer to these nodes as capable sensor nodes in the rest of this paper.) Based on the magnitude of the error between expected and observed output values of on-board sensors, these nodes determine the most appropriate state of operation of each sensor (Section 2) which they communicate to BS/RU along with the data.

In the second phase, BS/RU employ a pre-deployed sen-

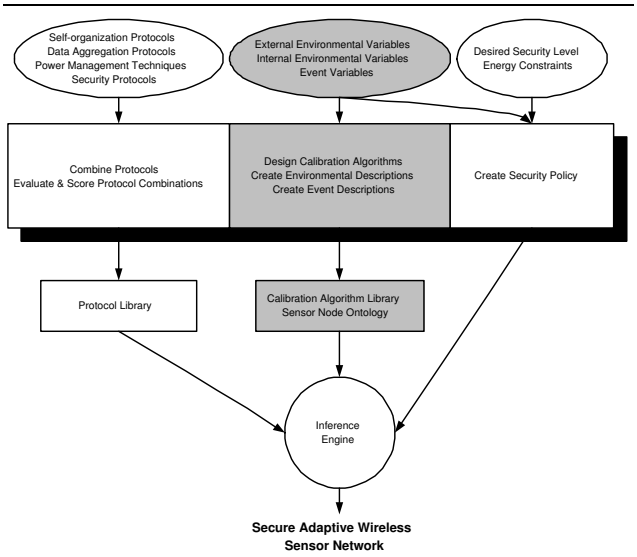

Figure 1. Framework for Secure Adaptive Wireless Sensor Networks

sor node ontology (Section 5) written in OWL-Lite [9] to reason over data received from sensor nodes and data that the BS/RU nodes have themselves sensed. Using data obtained from their sensors, BS/RU nodes calibrate data obtained from resource-poor nodes which did not perform data calibration before transmission. By reasoning over calibrated data BS/RU nodes collectively determine the most appropriate operating state of the WSN under existing environmental conditions and instruct sensor nodes to operate at that state.

Our work on adaptive wireless sensor networks, discussed in this paper, is part of a larger effort to build a framework to enable the design of secure wireless sensor networks that can adapt to changing environmental, topological and security conditions. Figure 1 shows the various components of this framework, and pinpoints the component discussed in this paper. Our goal is to build, simulate and evaluate the entire framework as a whole. This precludes us from simulating the component discussed in this paper in isolation and providing empirical results. Instead, we present a complete application scenario, and discuss how our two-phase approach is a solution to the wireless sensor network adaptivity problem.

The rest of the paper is organized as follows. In Section 2, we discuss our WSN model including capabilities of BS/RU and sensor nodes. In Section 3, we lay out the application scenario to which we can apply our solution. In Section 4, we describe the data calibration algorithm in detail. We discuss the sensor node ontology and the WSN state determination algorithm in Section 5. In Section 6, we discuss examples of classification schemes that can be implemented in the application scenario. In Section 7 we present related work in the area of adaptive sensor networks. We conclude and present directions for future work in Section 8.

2. Wireless Sensor Network Model

In our model, the WSN consists of a limited number of static or mobile BSs and RUs, but is otherwise unattended [1]. Each BS and RU can support hundreds of sensor nodes, which can range from the simplest, least expensive ones (e.g., Berkeley Motes, SmartDust) to medium sized sensor nodes. Each BS and RU node possesses significantly higher computational, storage and communication capacity compared to sensor nodes. The sensor node ontology, which completely describes a sensor node in terms of its individual components and their characteristics, is pre-deployed on each BS and RU. Each BS and RU possesses sensor interfaces to sense a set of core environmental variables, such as temperature, pressure, humidity and wind velocity.

Each sensor node possesses limited computational, storage and communication capabilities. Each node contains a protocol stack [1], which enables it to communicate with neighbors, establish connections with them and reach the nearest BS or RU. Thus, in our model, the WSN consists of clusters of sensor nodes, in which either a BS or RU plays the role of a cluster-head [2, 3].

As discussed in Section 1, each sensor node consists of one or more sensor types. The following operating modes are associated with each sensor type, based on information provided in specifications (e.g., data sheets) for that type: NORMAL, HIGH-SAMPLING, VERY-LOW-SAMPLING and OFF. Further, specifications provide information about energy consumption of each sensor type in each of these modes. Capable sensor nodes can compute the accuracy of each sensor in its present mode of operation. Therefore, the state of each on-board sensor is described by the following quadruple: $< t, m, e, a >$, where t, m, e and a stand for sensor type, operating mode, energy consumed and accuracy, respectively. Each sensor node tracks its overall energy consumption and knows the latest value of the remaining energy level. Thus, dynamic information about a sensor node can be summarized by the following tuple: <E, {S}>, where E stands for the node's present energy level and {S} is a set of one or more quadruples, describing the state of each on-board sensor.

Capable sensor nodes store a limited amount of information about input-output characteristics of each on-board sensor in the form of lookup tables. Input-output characteristics are a strictly one-to-one mapping between input and output values of the sensor. Similarly, they store the range of expected output values from each sensor type for different values of each of the core environmental variables, ei-

ther in a lookup table or as an executable function. Nominal values of core environmental variables are pre-deployed on each sensor node. All sensor nodes periodically receive current values of core environmental variables broadcast by BS/RU nodes functioning as cluster-heads. Capable nodes employ these values during data calibration (Section 4).

Each sensor node is in one the following three states at all times: ACTIVE, LOW-POWER and INACTIVE. Upon deployment, all sensor nodes discover each other and establish secure communication channels with each other. Subsequently, all nodes transition to the LOW-POWER state. All nodes in our model contain the required security mechanisms and protocols required to establish secure communications [2].

We emphasize that the WSN adaptivity problem is meaningful and therefore considered, only for an established network. Thus, issues of routing, transport, power management and data aggregation are dealt with in other work [1, 2, 3] and are beyond the scope of this paper.

3. Application Scenario

In this section we describe a scenario that requires a deployed WSN to adapt to changing environmental conditions. The application we consider deploys the WSN in a hostile environment, i.e., a battlefield. The task of the network is surveillance of enemy forces. There exists a set of user-defined energy consumption and accuracy bounds that can be combined in different ways. These energy-accuracy combinations are generated, prioritized and made available to WSN by the user.

In this example, the geographical environment is a desert region consisting of plains intersected by a river bed that can be crossed using one or more bridges. The major task is to detect troop/vehicle movement across the river bed and determine, according to prescribed energy-accuracy bounds, the types of troops, vehicles, weapons and their numbers.

Given that the network is deployed in a desert, the principal environmental variable is temperature. The temperature in this region is very high in the afternoon and very low at night. Chances of high wind velocity and frequent sand storms are very high. There is considerable variation in humidity in the region over different seasons. All BS/RU in the WSN possesses high-quality sensors to detect these environmental conditions and periodically transmit their current values to sensor nodes as appropriate.

Five types of sensors are deployed in the network. Each type has unique input-output characteristics, dynamic characteristics and responses to changes in environmental variables.

1. Vibration Sensors: These are fixed on the bridge to detect sudden high vibrations caused by movement of heavy vehicles (e.g.,. tanks, armored carriers) over the bridge. Their input-output characteristics are directly affected by inherent vibrations of the bridge and frequency of input vibrations. The characteristics are indirectly affected by temperature.

2. Pressure Pad Sensors: Pressure pads are embedded in the ground on all paths leading to the bridge to detect movement and estimate the weight the vehicles. Their input-output characteristics are directly affected by frequency of input pressure and indirectly affected by temperature.

3. Acoustic Sensors: These sensors are deployed both on the bridge and on all paths leading to it, to detect sound generated by moving troops and vehicles. Their input-output characteristics are directly affected by noise and indirectly affected by temperature.

4. Radioactive Sensors: These sensors are used to detect radioactive emission from weapons systems on-board the vehicles. They are highly accurate over short spans of the bridge. Their input-output characteristics are directly affected by available energy. Thus, these sensors must be activated selectively to conserve energy.

5. Line of sight optical sensors: These sensors employ a transmitter and receiver on either side of the bridge. Detection is accomplished by breakage in the line of sight beam from the transmitter to the receiver. These sensors are employed on all paths leading to the bridge for early detection of enemy movement. The number of times the beam is broken and the amount of time it stays broken can be used to give information about counts and lengths of the vehicles. These sensors are directly affected by reduction in visibility (e.g., fog, sandstorms) and humidity.

4. Data Calibration Algorithm

In this section, we discuss the algorithm executed by capable sensor nodes and possibly BS/RU nodes to perform data calibration on each type of sensor whenever significant changes in environmental conditions occur.

There are three inputs to the algorithm as shown in the flowchart in Figure 4: current and nominal values of each of the core environmental variables and the current data output of the sensor type. Based on these inputs, the data calibration algorithm computes the expected output of each sensor type at existent values of core environmental variables. For example, the core environmental variable affecting acoustic sensors described in Section 3 are temperature and external noise. Further, the acoustic sensor's output is a voltage corresponding to the input sound intensity. The final output of the algorithm for the acoustic sensor is a calibrated value of the sensor's output.

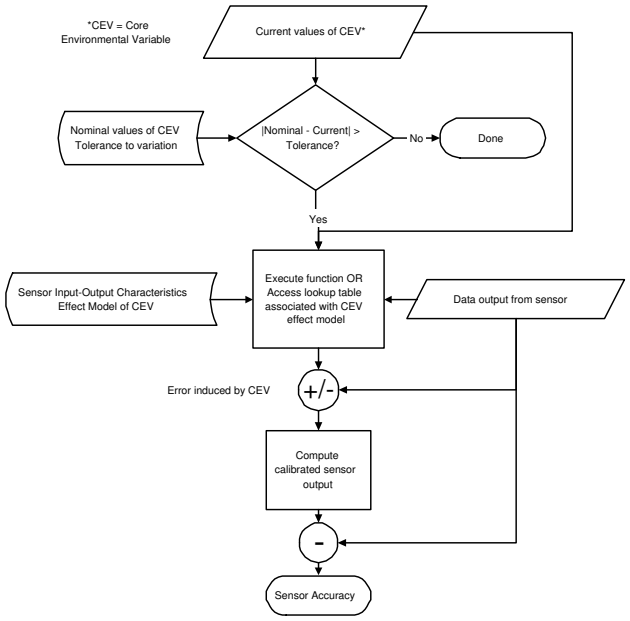

Figure 2. Data Calibration Algorithm

We now explain the flowchart in detail. Initially, the algorithm determines whether data calibration must be performed. This is accomplished by determining if the current value of each core environmental variable exceeds its nominal value by a pre-determined threshold value. If so, data calibration is required.

The algorithm uses either the lookup table or function (Section 2) describing the effect of each core environmental variable on the input-output characteristics of each sensor to determine its expected output value. The lookup table maps different values of a core environmental variable to corresponding output values of the sensor. On the other hand, a function relates sensor input, sensor output and the core environmental variable and describes how the output value changes according to the environmental variable.

For example, assume that the average temperature coefficient of sensitivity (the function in this case) of above acoustic sensor is 1%/°C with respect to the nominal value. Therefore, if temperature increases by 10°C, then sensitivity (which is the ratio of output to input of a sensor) changes by 10%, which implies that for the same input value, the output of the acoustic sensor either reduces or increases by 10%. Thus, having obtained the error induced by the increase in temperature, the algorithm corrects the measured output accordingly.

Based on the measured and calibrated output values, the algorithm computes the accuracy of the sensor and stores this value in the a field of the $< t, m, e, a >$ quadruple. The current accuracy value of a sensor type determines its most appropriate operating state (m) and corresponding energy consumption (e) in that state. Individual sensor nodes cannot make this determination because they do not possess either the user-defined energy accuracy bounds or a global view of the existing state of the WSN. Only BS/RU nodes, which possess both the capability and the information to obtain the global state, can determine the final operating state of each sensor type in the WSN, under existing environmental conditions (Section 5.1). Thus, each capable node places the current operating state and corresponding energy consumption value in the m and e fields of the quadruple, respectively. The capable sensor node transmits the quadruple in a message to the BS/RU node functioning as the sensor node's cluster-head.

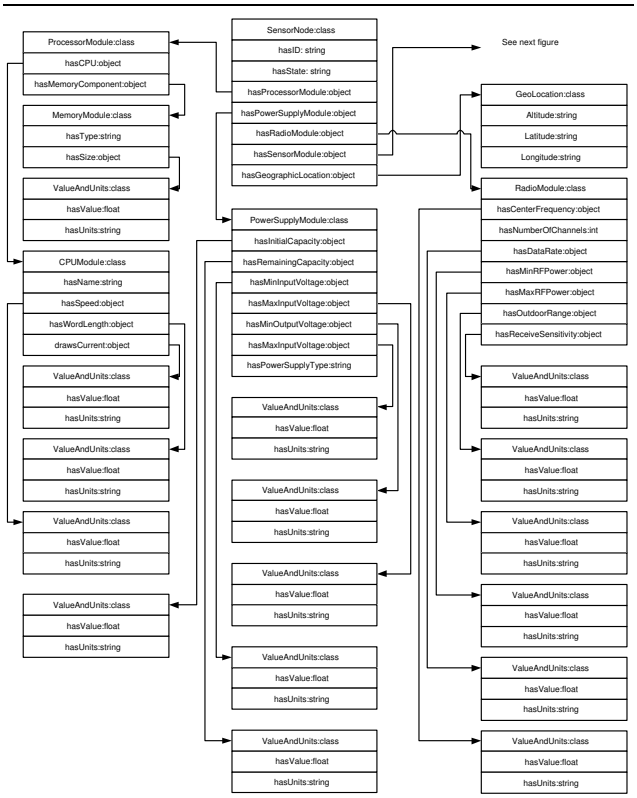

Figure 3. Sensor Node Ontology – Processor, Power Supply and Radio Modules

Sensor nodes that are incapable of executing this algorithm do not transmit the quadruple to their respective cluster-heads. The nodes merely transmit uncalibrated data. Cluster-heads calibrate data received from these nodes and use prior state information associated with sensor types onboard the nodes to determine the final operating state of each sensor type.

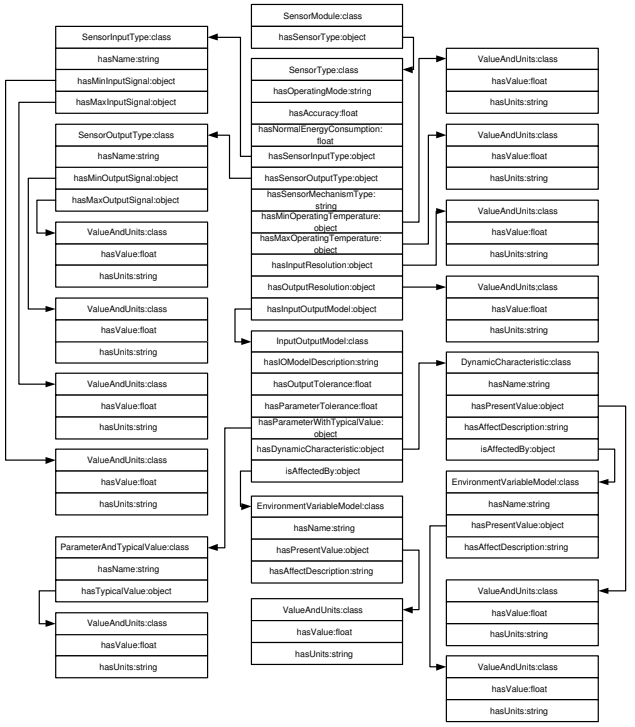

Figure 4. Sensor Node Ontology – Sensor Module

5. Sensor Node Ontology

The sensor node ontology we have designed attempts to capture the most important features of a sensor node that describe its functionality and its current state. The information captured by our ontology is used by BS/RU to determine the current and future state of the WSN. The ontology describes the main components of a sensor node: processor, power supply, radio and sensor modules. A common thread that runs through these modules is energy, both in terms of capacity and consumption. Figures 3 and 4 show our sensor node ontology in detail.

The processor module consists of the CPU and memory components. The most interesting and important facets of a CPU that influence the functioning of a sensor node are the CPU's available operating modes (i.e., active, low-power, idle) and the amount of power it consumes in each mode. Based on this information, the WSN can decide the most appropriate CPU operating mode for each sensor node or a class of sensor nodes, under existing environmental conditions. The memory component reflects both static and dynamic memory on the sensor node.

The power supply module is an essential component of a sensor node. This module determines how long the sensor node can function usefully. Given that a typical sensor node is self-contained and is typically unattended after deployment, the power supply module is the final limiting factor on the sensor node's capabilities. This is reflected in two of the most important characteristics of the power supply module – initial energy capacity and remaining energy capacity. Remaining energy capacity dictates the ability of the sensor node to continue functioning in a certain mode. Information regarding energy capacity of a node plays a very important role in WSN adaptivity.

The radio transceiver module helps the sensor node communicate with other nodes in the network. It consists of a transmitter and receiver. Identically to the CPU component, the most important properties of the radio transceiver that affect the sensor node's functioning are its available modes of operation (i.e., transmit, receive, low-power, sleep) and power consumption in each mode. Each operating mode of the transceiver reflects environmental conditions affecting the node.

The sensor module consists of descriptions of one or more sensor types. For each sensor type, primary information captured in the ontology includes the $< t, m, e, a >$ quadruple, i.e, sensor type, operating mode, accuracy and energy consumption along with a description of the effect of each core environmental variable on the sensor type. Additionally, it also captures static information about the sensor type, such as its input and output signal types, input and output resolutions, and minimum and maximum operating temperatures.

Additional useful information present in the description of a sensor node includes its ID and geographical location. The former is pre-deployed on the node while the latter can be obtained post-deployment if the node possesses a Global Positioning System (GPS) unit on-board.

5.1. WSN State Determination Algorithm

This algorithm enables a WSN to dynamically determine its most appropriate state of operation under existing environmental conditions. The operating state of a WSN reflects the corresponding states of sensor nodes and each type of sensor in the WSN. Figure 5 shows a flowchart of this algorithm, which is executed by each BS/RU node currently functioning as a cluster-head (Section 2).

The input to this algorithm is the <E, $\{$S$\}$> tuple transmitted by each sensor node to its cluster-head. We recall that E is the remaining energy on the sensor node and $\{$S$\}$ is the set of $< t, m, e, a >$ quadruples associated with its sensor types.

Initially, the algorithm extracts and classifies each $< t, m, e, a >$ quadruple according to a pre-determined classification scheme (e.g., sensor type, sensor energy consumption, sensor operating mode, node location). In Section 6, we discuss two different classification schemes applied to the WSN described in the application scenario (Section 3)

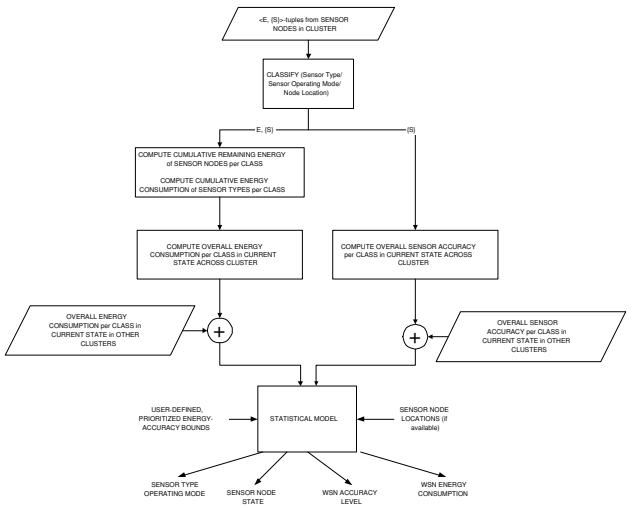

Figure 5. WSN State Determination Algorithm

and show how the WSN adapts to environmental changes, based on each classification scheme. Subsequently, the algorithm computes the overall energy consumption per class of sensor nodes by combining the cumulative remaining energy E, of sensor nodes in this class and the cumulative energy consumption of sensor types (in their current operating modes) that belong to the class. In a similar manner, the algorithm computes the overall accuracy of sensor types in the class.

At this point, each cluster-head has obtained a class-based view of the current energy consumption and accuracy of nodes in the cluster. In order to obtain a global view that encompasses all nodes in the cluster, each cluster-head obtains similar class-based views of all other clusters in the WSN computed by their corresponding cluster-heads. This exchange of class-based views is restricted to cluster-heads, which employ out-of-band secure channels for this purpose.

As shown in the flowchart, the algorithm combines all class-based views of all clusters in the WSN to obtain a global view of the energy consumption and accuracy per class of sensor nodes in the WSN. We emphasize that these class-based energy consumption and accuracy values reflect the current state of the WSN. In order to enable the WSN to continue functioning according to user-specified energy-accuracy bounds on the WSN, each cluster-head computes the most appropriate operating state (i.e., ACTIVE, LOW-POWER, IDLE) of sensor nodes and operating mode (i.e., NORMAL, HIGH-SAMPLING, LOW-SAMPLING, OFF) of sensor types, respectively, per class of sensor nodes.

The algorithm accomplishes this task by providing the class-based energy consumption and accuracy, along with user-specified accuracy and energy bounds, and sensor node locations (if available) as inputs to a statistical model pre-deployed on each BS/RU node. The class-based operating state of sensor nodes and operating modes of sensor types are disseminated to each sensor node (only the values relevant to the node) in the corresponding class.

Upon receipt of this information, each sensor node changes its state and the operating modes of its sensors accordingly, thereby ensuring that all sensor nodes have adapted to environmental conditions.

Additional information output by the statistical model includes WSN accuracy and energy consumption, after nodes have adapted to environmental changes. This information can be provided to the user who may decide, for example, to deploy more sensor nodes to increase WSN accuracy.

6. Examples of Classification Schemes

We now discuss two different schemes using which each BS/RU node obtains global views of energy consumption and accuracy of the WSN deployed in the application scenario described in Section 3.

6.1. Location-based classification

This scheme requires either prior knowledge of sensor node locations (e.g., nodes deployed on the bridge) or mechanisms to determine node locations after deployment (e.g., GPS). Thus, each BS/RU obtains the geographical location of every node in its cluster. Prior to deployment, each BS/RU is programmed to classify nodes using geographical location as the classifier.

Using this classification scheme, BS/RU functioning as cluster-heads of all clusters in regions farthest from the bridge instruct their cluster to transition to the ACTIVE state. From this point onward, sensors on-board nodes in these clusters actively sense and attempt to detect enemy presence and movement in the region. These clusters form the largest ring in the "perimeter" around the bridge. Immediately after activation and periodically thereafter, each sensor node in these clusters receives current values of external environmental variables that affect it. Given that conditions in a desert region can vary significantly during the day (due to sudden sandstorms, rising temperatures etc.) and night (sudden cooling), each capable node in these clusters executes the data calibration algorithm discussed in Section 4 to calibrate data from each sensor before disseminating it. Additionally, it produces the <E, {S}>-tuple as discussed in Section 4 based on accuracies of each of its sensor types and their operating modes.

After receiving the <E, {S}>-tuple from each sensor node in its cluster, the cluster-head determines the cumulative accuracy and energy consumption of its cluster. It uses the statistical model to compute the most appropriate

operating mode for each sensor on each node in the cluster. It also determines whether or not to activate nodes in the next inner ring of clusters in the perimeter around the bridge. This becomes necessary if the accuracy of the previous ring of clusters falls below user-defined bounds; otherwise the next ring is not activated. Each BS/RU repeats this process until either the ring of clusters closest to the bridge is reached or energy bounds specified by the user are reached.

As outer rings of clusters detect enemy movement within some accuracy bounds, inner rings are consecutively activated (if previously inactive). Assume that the enemy has reached the bridge and different rings of clusters have confirmed its presence. Now, nodes on the bridge are activated to actually determine the types of vehicles and possibly the types of weapons carried in those vehicles. This calls for a new global view of the WSN – one based on sensor types.

6.2. Classification Based on Sensor Type

As discussed earlier, the main sensor types on the bridge are pressure pad sensors, vibration sensors and radioactive sensors supported by optical sensors on either side of the bridge. Radioactive sensors consume high energy when active, therefore they should only be activated when absolutely necessary. Thus, initially only pressure pad and vibration sensors on the bridge are activated along with the optical sensors. The pressure pad and vibration sensors' task is to identify heavy vehicles (e.g., a 30-ton truck) which may suggest that they are carrying nuclear weapons. This causes the radioactive sensors to be activated to detect the weapons. The optical sensors' task is to count the number of vehicles, troops etc., attempting to cross the bridge.

In this situation, BS/RU are concerned about the accuracy of pressure pad and vibration sensors separately. Thus, when they receive <E, {S}>-tuples from all nodes on the bridge, they use sensor type as the classifier to separate pressure pad sensors from vibration sensors. Each BS/RU computes cumulative accuracies of these sensors separately and determines their most appropriate mode. If they detect heavy vehicles with high accuracy then radioactive sensors are activated immediately and their energy consumption is monitored to ensure that the energy-accuracy bounds are continuously met. They are made inactive as soon as a determination on the weapon type is made. This process repeats until all vehicles capable of carrying nuclear weapons are identified. Finally, each BS/RU reports the cumulative data, accuracy, confidence on accuracy and remaining energy back to the command and control, which is the primary user of this WSN, for appropriate action.

7. Related Work

In this section, we discuss work published by the research community on sensor node calibration [5, 11], WSN adaptivity [7, 10] and mechanisms to describe sensor nodes [8].

Whitehouse and Culler [11] discuss macro-calibration in sensor/actuator networks and describe an ad-hoc localization system. The authors focus on methods to estimate the distance between nodes and thereby calibrate their location. They describe an ad-hoc localization system called *Calamari* that uses a fusion of RF received signal strength information (RSSI) and acoustic time of flight (TOF). The authors contend that ad-hoc localization cannot be accomplished on a global scale, i.e., across the sensor/actuator network, using calibration functions. This is because the transmitter/receiver pair of each device in the network can only be calibrated with that of a neighboring device, thereby achieving only pairwise localization. Therefore, they frame calibration as a parameter estimation problem; they parametrize each device and choose values of those parameters that optimize the performance of the overall system. In our solution, sensor nodes do not employ data calibration for the purpose of localization. Our data calibration algorithm is used to determine the error induced in sensors due to variations in environmental conditions.

Bychkovskiy et al. [5] describe a collaborative approach to sensor calibration. They describe a two-phase technique designed to calibrate devices in large-scale, dense networks. In the first phase, the technique derives relative calibration relationships between pairs of co-located sensor nodes. Subsequently, in the second phase, the technique maximizes the consistency of pairwise calibration functions among groups of sensor nodes. The main idea in the first phase is to employ temporal correlation of signals received at neighboring sensor nodes when the nodes are observing the same phenomenon. The second phase is formulated as an optimization problem. In our solution, input-output characteristics of sensors at nominal values of core environmental variables are pre-deployed on BS/RU and capable sensor nodes. Thus, there is no need to correlate input signals with neighboring sensors in order to calibrate each sensor. In fact, in our solution neighboring sensor nodes need not have any sensor type in common. This is because calibration is a per sensor node operation and does not involve neighboring sensor nodes. Classification, which is akin to correlation described above, occurs only at BS/RU nodes which obtain a global view across types of sensors.

Heinzelman et al. [7] discuss the design and evaluation of a family of adaptive protocols, called SPIN, for information dissemination in WSN. The main idea in this work is to eliminate redundant data transmissions by sensor nodes to help conserve energy. Nodes achieve this by naming

data, i.e., create meta-data, prior to transmission. Neighbors which receive this data use the meta-data in conjunction with application-specific knowledge and their own energy levels to decide whether or not to accept, aggregate and re-broadcast this data. In our work, nodes are assumed to have established security associations with each other during the bootstrap process, thereby precluding them from re-broadcasting data arbitrarily. This indirectly ensures energy conservation on nodes because, unless the destination of the data is specified as a broadcast address, the security association only allows unicasting. Additionally, in our work, while each node can decide whether or not to participate in the WSN, BS/RU in the WSN make a globally appropriate decision, based on existing environmental and topological conditions. SPIN does not address the issue of data calibration and does not consider environmental variables other than the local energy level to make the WSN adaptive.

Tilak et al. [10] discuss infrastructure trade-offs in sensor networks. For a given application and infrastructure of the sensor network, they study the effect of changing the underlying network protocol. The study is conducted for different applications (habitat monitoring, temperature tracking) and WSN infrastructures (random, uniform, planned). The goal of the study is to determine application-level performance based on metrics such as accuracy (with respect to distance between sensor and target), latency, energy efficiency, goodput and scalability.

Sensor Model Language (SensorML) [8] is an XML schema for meta-data that describes sensors from a functional perspective rather than hardware. The schema also defines sensor platforms (i.e., physical mounts for sensors), interfaces that allow tasks to be submitted to sensors and the data derived from sensors. SensorML descriptions of sensors are primarily intended for human users. In contrast, our ontology provides high-level descriptions of the various components of sensor nodes (i.e., the platforms in OpenGIS literature), which can be given as input to a reasoning engine for purposes of classification. In addition, our ontology provides functional descriptions of sensors and the affects of internal and external environmental variables on sensor functionality. This information enables BS/RU nodes in the WSN to determine the most appropriate operating state of each class of sensor nodes, based on existing environmental conditions. Finally, our ontology allows human users to interact with a deployed WSN via BS/RU nodes to obtain class-based, cluster-based or network-based views.

8. Conclusions and Future Work

We have proposed a novel two-phase solution to the wireless sensor network adaptivity problem. Our solution employs a data calibration algorithm, a comprehensive sensor node ontology and a WSN state determination algorithm to enable a deployed WSN to dynamically adapt itself to changing environmental conditions. Our solution provides a two-fold benefit – it allows human users to define initial conditions and expected behavior of the network upon deployment and it enables the network to adapt to changing conditions to meet the user-defined requirements. It has applications in the domain of the Semantic Web as it enables human users to communicate with the network using semantically rich descriptions of queries and requests, which the latter can reason over and understand. Additionally, our approach is applicable to networking and control domains as it enables the network to determine existing conditions, detect significant changes to the environment, infer the meaning of these changes and adapt to them.

In future work, we will describe results of simulations of the proposed solution, in conjunction with the other components of our framework for secure, adaptive sensor networks, on a large scale. Work on incorporating mechanisms to enable a deployed WSN to adapt to changing security conditions in addition to environmental conditions is ongoing and results will be reported in future work. For example, if the underlying security protocols determine that a sensor node in a cluster has been compromised, the WSN should determine if the entire cluster is likely to be compromised and take appropriate action. The WSN must be able to determine the consequences of such an action, in terms of connectivity and other metrics.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: A survey. *Computer Networks*, 38(4):393–422, March 2002.

[2] S. Avancha, J. Undercoffer, A. Joshi, and J. Pinkston. A Clustering Approach to Secure Sensor Networks. Technical Report TR-CS-03-19, University of Maryland, Baltimore County, October 2003.

[3] S. Avancha, J. Undercoffer, A. Joshi, and J. Pinkston. An Approach to Secure Self-organization in Distributed Wireless Sensor Networks. Technical Report TR-CS-04-03, University of Maryland, Baltimore County, April 2004.

[4] S. Avancha, J. Undercoffer, A. Joshi, and J. Pinkston. *Security for Wireless Sensor Networks*, chapter 12. Wireless Sensor Networks. Kluwer Academic Publishers, 2004.

[5] V. Bychkovskiy, S. Megerian, D. Estrin, and M. Potkonjak. A Collaborative Approach to In-Place Sensor Calibration. In F. Zhao and L. Guibas, editors, *Proc. of the International Workshop on Information Processing in Sensor Networks, IPSN'03*, volume 2634, pages 301–316, 2003.

[6] J. Elson and D. Estrin. *Sensor Networks: A Bridge to the Physical World*, chapter 1. Wireless Sensor Networks. Kluwer Academic Publishers, 2004.

[7] W. R. Heinzelman, J. Kulik, and H. Balakrishnan. Adaptive Protocols for Information Dissemination in Wireless Sensor Networks. In *Mobicom '99*, August 1999.

[8] M. Smith. *Sensor Model Language (SensorML) for In-situ and Remote Sensors*. OpenGIS Consortium, Inc., December 2002.

[9] M. K. Smith, C. Welty, and D. M. eds. *OWL Web Ontology Language Guide*. W3C, December 2003.

[10] S. Tilak, N. B. Abu-Ghazaleh, and W. R. Heinzelman. Infrastructure Tradeoffs for Sensor Networks. In *WSNA '02*, September 2002.

[11] K. Whitehouse and D. Culler. Macro-Calibration in Sensor/Actuator Networks. *Mobile Networks and Applications*, 8(4):463–472, 2003.