Behavior Monitoring Framework in Large-Scale Wireless Sensor Networks

Feng Wang
Mathematical and Natural Sciences
Arizona State University
fwang25@asu.edu

Jianhua Gao
Computer School
Wuhan University
Jianhua.Gao@asu.edu

Abstract

Wireless sensor networks (WSNs) have been increasingly deployed for both civil and military applications under harsh, unpredictable or open environments. Such environments create opportunities for the intruders to launch a variety of attacks on multiple protocol layers in WSNs. This paper proposes a behavior monitoring and analysis framework for large scale WSNs. Within this framework, we address the monitor node selection problem and introduce a dedicated monitor nodes approach and propose a greedy algorithm for selecting monitor nodes. The simulation results show that the greedy algorithm is efficient in terms of monitor node set size.

1 Introduction

In recent years, wireless sensor networks (WSNs) have been increasingly deployed for both civil and military applications, such as military surveillance, structural health monitoring, environmental monitoring, search and rescue, and target tracking. These sensor networks contain a large number of resource-constrained sensor nodes deployed in harsh, unpredictable or open environments. Such environments create opportunities for the intruders to launch a variety of attacks on RF layer, MAC layer, routing protocols and applications in WSNs, such as radio jamming, node capture, selective forwarding, and message manipulation [1], etc. These attacks, if successful, could easily disrupt the normal functionality of the WSNs. Given the importance of many critical applications of WSNs and the frequent emergence of attacks in sensor networks, it is very important and necessary to gain a deep understanding of communication patterns of sensor nodes and develop an efficient security monitoring system for WSNs.

Many researchers have recently focused on the reliability, robustness, health monitoring and diagnosis, and intrusion detection of WSNs. Most efforts on sensor network health monitoring and diagnosis are centered on the availability and health status of sensor nodes, communication links, routing paths, and identifying the root causes of different failures. The research studies of intrusion detection mechanisms mostly address specific attacks and their countermeasure. However, there is little attempt to build a security monitoring framework which extends availability monitoring framework to monitor the behavior of the sensor nodes in order to capture anomalous behaviors and attacks.

In this paper, we propose to develop a behavior monitoring framework towards this end. In the behavior monitoring framework, each monitor node creates a behavior profile of its neighbors and its neighborhood. It helps discover interesting events by exploring 1)whether a node behavior profile matches a pre-defined attack signature, ii) whether the behavior of a specific node deviates from its neighbors, and iii) whether a node exhibits a significant behavior change. Our ultimate goal is to point out the suspicious behavior and aid the sensor network administrators for in-depth investigations.

Behavior monitoring of WSNs is not a trivial task considering the inherent limitations of wireless sensor networks, such as scarce sensor energy, limited sensor computational power and storage space, limited RF bandwidth, frequent node/link failures, and uncontrolled environment. A behavior monitoring framework is based on the answers of the following three key questions: *what behavior information to collect*, *how to collect these information*, and *how to analyze, or make sense of, the collected information*. To answer the "what to collect" question, the challenges lie in the design of behavior metrics that introduces a minimum amount of monitoring overhead while maximizing the intelligence of the monitoring messages. For the "how to collect" question, the challenges lie in how to determine the topology of monitor nodes and the relationship between a sensor node and its monitor node. To answer the "how to analyze" question, the challenges include how to adaptively categorize the behavior of an individual sensor node or a group of sensors, and how to efficiently and accurately pinpoint the root cause of attacks or suspicious activities.

In light of the above challenges, we choose to adopt a combination of passive local monitoring and active global monitoring for reducing the communication overhead and the response time to attacks. A critical problem in this monitoring model is how to strategically locate local monitors. We address the monitor node selection problem and introduce a novel dedicated monitor nodes approach and propose a greedy algorithm for selecting monitor nodes. We evaluate the impact of node number and average node degree on the monitor node set size. The simulation results show that for a deployed network with average node degree around 15, only 6% of the deployed sensor nodes are chosen as the monitor nodes. To answer the "what to collect" question, we study the communication patterns of normal traffic and existing attacks, and propose a set of behavior metrics that the monitor nodes collect. The anomaly detection employs the principles of message symmetry, node similarity, data verification, and data plane visualization. We also present the design of a prototype of the behavior monitoring system.

The remainder of this paper is organized as follows. Section 2 briefly summarizes related work. Section 3 provides the architecture and design of the proposed behavior monitoring framework. Section 4 presents a greedy algorithm for the monitoring node selection problem and analyzes its performance through simulations. Section 5 concludes the paper.

2 Related Work

Most of the relevant work can be categorized into diagnosis and debugging, health monitoring, and intrusion detection in WSNs.

Diagnosis and debugging: Sympathy [2], a centralized prototype sensor network debugging tool, aims to detect and debug failures through collecting a comprehensive set of metrics including node connectivity and data flows. Although Sympathy is able to efficiently localizes the root causes of a number of sensor network failures, its centralized metric collection triggers a significant amount of additional data communications traffic and energy consumptions in wireless sensor networks. [3] proposes a passive approach based on inference-based network diagnosis. This work employs a packet marking algorithm that constructs and maintains the inference model to infer the root causes of abnormal phenomena. This passive approach introduces little traffic to the network, however it heavily depends on the data flows of the applications, which, in the cases of event-detection applications, could delay the diagnosis and notifications of failure events.

Health monitoring: Several work [4, 5] are devoted to the health monitoring of individual sensor nodes or entire sensor networks. In [4], a two-phase timeout system for health monitoring is proposed to utilize local monitoring as much as possible to reduce network traffic. Instead of having each sensor to periodically report its liveness to the sink, only the neighbors, which detect and confirm node failures, report such events. [5] summarizes three classes of valuable information that a sensor network management system can provide for network operators: i) failure detection, e.g., informing node failure; ii) symptom alerts, e.g., informing symptoms of impending failure or degraded performance; iii) ex post facto inspection, e.g., informing the operators of the event timeline to infer the reasons of the failures. In addition, [5] highlights the tradeoff between three important factors of health monitoring: accuracy, timeliness and efficiency.

Intrusion Detection: Many intrusion detection techniques in wireless sensor networks have been recently proposed, however most of these work focus on specific attacks and their solutions. In [6], Yang et al. are interested in improving the survivability of sensor networks under worm attacks through a software diversity mechanisms, while [7] models and analyzes the spreading process of code compromise by viruses or worms, and identifies the key factors that determine the potential outbreaks. [8] focuses on threat models and attacks against routing protocols for wireless sensor networks. [9] designs and implements a secure access system for wireless sensor networks for restricting the network access only to eligible sensor nodes and filtering messages from outsiders, while [10] explores the spatial correlation among the networking behavior of sensor nodes to detect inside attackers. [11] introduces a suite of security protocols for sensor networks to provide authenticated and confidential communication, and authenticated broadcast. Similarly, [12, 13] integrate confidentiality and DoS-attack-resistance in code dissemination protocols for wireless sensor networks. [14] detects node compromise during the stage of compromised sensor redeployment based on the change of node neighborhood and the change of measured distances between sensor nodes. [15] provides a real-time detection of the clone attacks in sensor networks through social fingerprints of sensor nodes. [16] mitigates the attacks against control traffic by detecting, diagnosing, and isolating the malicious nodes, while [17, 18] implement a general mechanism of packet leashes and explore connectivity information to uncover hidden substructure in the connectivity graph respectively for detecting and defending against wormhole attacks.

Our work is different from the diagnosis and debugging and health monitoring work since it is not limited to availability monitoring and failure detection. It is different from the intrusion detection work because the existing intrusion detection techniques have different assumptions on hardware, topology and applications which makes it very challenging to integrate them together into a general platform. In other words, there lacks a systematic framework to detect a comprehensive set of suspicious behavior based on the behavior signatures or fingerprints of the attacks. In this paper, we propose to develop a behavior monitoring framework in wireless sensor networks to detect anomalous behaviors or attacks based on behavioral characteristics of these attacks. More importantly, this framework can also be extended to include emerging attacks by simply incorporating their behaviors.

3 Behavior Monitoring Framework

In this section, we present the principles and design of our behavior monitoring framework. Our design is based on the following three main observations: 1) communication patterns of wireless sensor network is mostly well-defined, which means normal behavior of each sensor can be defined; 2) passive monitoring introduces less monitoring overhead to the wireless sensor networks due to the broadcast nature of the wireless channel; 3) wireless sensor network is a closed environment. In the remaining of this section, we further elaborate these observations and present a general *behavior-oriented* methodology for modeling sensor nodes and detecting suspicious behaviors on routing and data planes in WSNs.

3.1 Behavior-oriented methodology for modeling sensor nodes

Fig. 1 illustrates the overall architecture of the proposed behavior monitoring framework in WSNs where monitor nodes collect the behavior metrics through sniffing the broadcast channel, make local decisions, aggregate behavior metrics of sensor nodes and then report to the base station for further global analysis and visualization.

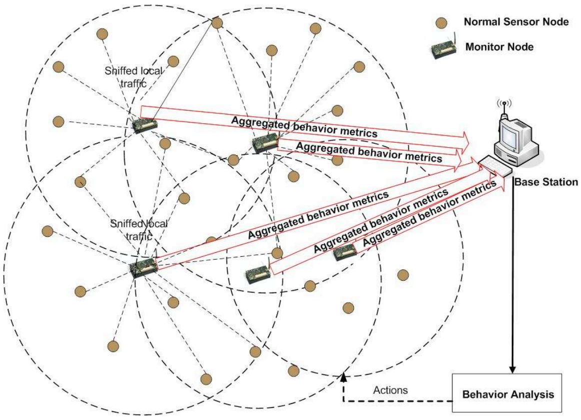

Normal Sensor Node
Monitor Node
Shiffed local traffic
Aggregated behavior metrics
Aggregated behavior metrics
Base Station
Sniffed local traffic
Aggregated behavior metrics
Aggregated behavior metrics
Aggregated behavior metrics
Actions
Behavior Analysis

Figure 1. The architecture of behavior monitoring framework for WSNs.

The major intuition of this framework lies in that attacks typically leave certain behavioral fingerprints reflected in the transferred messages. By focusing on *when*, *where*, *who* and *to whom* of the messages delivered in the sensor networks, we could collect a large number of important information for efficiently uncovering malicious behavior. Specially, we would like to explore 1) what behavior information should the monitoring framework collect for detecting anomalous behavior; 2) how should the monitoring framework collect these metrics and forward to the base station in an energy-efficient fashion; and 3) how will the monitoring framework make sense of the behavior metrics to find the events of interest.

To answer these questions, this new framework is designed with three major components that work together to collect, analyze, and detect anomalous behavior in sensor networks. Specifically, these components are 1) modeling communication behavior of sensor nodes; 2) monitoring model; and 3) detecting anomalous behavior in Wireless Sensor Networks. In the following, we briefly summarize the main ideas and intuitions of each component.

3.1.1 Modeling communication behavior of sensor nodes

This component builds simple and intuitive behavior models for understanding the normal communication patterns of sensor nodes, which provides the baseline to detect the deviant behaviors that are likely triggered by interesting events such as active attacks or node/link failures. WSNs are typically deployed for specific applications such as health monitoring, and military tracking. The normal communications in such applications are mostly guided by certain routing and applications protocols. Therefore, the communications in the control plane and data domain are well defined, and sensor nodes are expected to exhibit certain behaviors in routing discovery phases and application data collection phases.

Applications are the main driving force of rapid development of wireless sensor networks. Unlike end systems in wired networks such as the Internet that could communicate with any other end systems, sensor nodes in wireless sensor networks follow a much simpler communication pattern with a predictable behavior under normal scenarios. In general, the communication pattern in a typical wireless sensor network can be categorized into *local communication* and *global communication*, as summarized in Table 1. Local communication serves the purpose of maintenance of neighborhood list, time synchronization, localization, and maintenance of routing paths, and such communications usually appear periodically. In addition, message relaying or forwarding is also categorized into local communication, since a relaying sensor node only forwards the data to one of its neighbors that is the next hop on the routing path.

Global communication happens between the sink (or the base station) and distributed sensor nodes for two major purposes: i) *many-to-one communication:* sensor nodes report events or application data to the sink, and ii) *one-to-many communication:* the sink disperses control messages or data queries to the sensor network or selected regions in the network. In event-driven applications, global communication from a sensor node to the sink is not predictable, since the communication largely depends on the occurrence of an event. However, in monitoring applications, sensor nodes regularly report sensor data such as temperature or acoustic information to the sink, thus creating a predictable communication pattern. The global communication from the sink to sensor nodes is also predictable, since its behavior is typically well defined during the deployment.

The simplicity of the communication patterns in wireless sensor networks creates an opportunity to build simple communication models to analyze the normal baseline behavior patterns for sensor nodes, local neighborhood, as well as the entire wireless sensor network.

3.1.2 Monitoring model

Data communications in wireless sensor networks typically consume most of the energy, thus we adopt the passive monitoring [19] model in the behavior monitoring framework, in which each monitor node passively listens to all the traffic on the broadcast channel, instead of requesting each individual sensor node reporting its status and traffic summary. The advantage of passive monitoring is the significant reduction of messages transferred in network, compared with active monitoring or self-monitoring where each sensor node periodically reports its own status and behavior information.

In order to reduce the number of monitoring messages transferred in the sensor networks, we combine the local and global monitoring. Based on the roles in the monitoring framework, we classify sensors nodes as *monitor nodes* and *normal sense nodes*. Local monitoring, carried out by distributed monitor nodes, could detect attacks in the neighborhood in a timely and energy-efficient manner. For example, by gathering the *local* information on the number of the incoming messages to a sensor node and the number of forwarded messages from the node, a monitor node can reveal the attacks of selective forwarding. On the other hand, global monitoring at the base station has a number of advantages in the behavior monitoring framework, given the knowledge, information, and computing resource available at the base station. More importantly, some attacks could only be detected or confirmed by the base station due to its comprehensive collection of the behaviors in the network. For example, a wormhole attack coordinated by two distant and compromised sensor nodes is very difficult for distributed monitor nodes to detect and confirm, but could be relatively easy for the base station to uncover with its collection of the behavioral fingerprint collected from all the distributed monitor nodes.

3.1.3 Detecting anomalous behavior in wireless sensor networks

The major goal of collecting behavior fingerprint is to detect anomalous behaviors in wireless sensor networks. We develop four major principles: *message symmetry*, *node similarity*, *data verification* and *data visualization*, for guiding the root cause analysis.

Message symmetry: wireless sensor networks are typically deployed for very specific purposes under isolated environments, where communications are expected to be closed. In other words, all the control and application traffic in a sensor network should stay within the network. Even if the sensor work connects to the Internet, and the data packets to or from the Internet traverse through the base station. A simple intuition behind such type of communications is that if a node in the sensor network sends a message and there is no node failure or link failure at the moment, another node or a set of other nodes must be receiving the message. On the other hand, if a node receives a message, the source of the message must be within the network. Therefore, the messages sent and the message received should be symmetric in sensor networks. By accounting the messages in the network, one could detect several attacks of message manipulations such as injecting false messages and dropping messages.

Node similarity: the intuition of node similarity lies in the nature of sensor nodes. Unlike the end systems on the Internet which could have different roles such as servers, clients or peer-to-peer nodes, sensor nodes perform similar tasks or functions to collect data or detect events for the applications and then report to the base station. As a result, the sensor nodes within a proximity area should exhibit a strong similarity in the messages they send or receive. By studying node similarity or dissimilarity, the distributed monitor nodes could discover compromised sensor nodes or attacks that behavior differently from other sensor nodes in the neighborhood.

Data verification: Local monitor nodes have the capability to validate the results from the intermediate routing nodes that aggregate the data before propagating further to the base station. These intermediate routing nodes, if compromised, could launch a number of attacks on the sensor network such as manipulating the aggregation and reporting false data, thus validating the aggregated results by the monitor nodes is very necessary and important. Through reconstructing the aggregation process in WSNs, *data verification* is able to uncover the attacks on the data aggregations.

Table 1. Communication patterns in a typical wireless sensor network.

Range	Pattern	Delivery	Objective
Local communication	one-to-many	multicast or broadcast	maintenance of neighbor list and routing path, time synchronization, localization
	one-to-one	unicast	forwarding of application sensor data, or its aggregations
Global communication	one-to-many	multicast or broadcast	a base station spreads control information or issues a query
	many-to-one	unicast	sensor nodes report occurrence of event, or report application sensor data

Data plane visualization: Visualization of data flows in the sensor networks at the base station is a very powerful mechanism to detect the attacks towards the routing protocols. For example, one could find indications of possible wormhole attacks, if one or a few links carry a very high percentage of data traffic in the sensor networks.

3.2 Design of a prototype system

Guided by these above principles and intuitions, we design a prototype system to evaluate the operational feasibility of the proposed monitoring framework, and to study the performance and cost of the framework. Fig. 2 illustrates the schematic design for the functions of the monitor nodes and the base station in the behavior monitoring framework.

In each *monitor node*, we implement a sniffing function, behavior fingerprint generator, behavior analyzers for one-to-many communications, many-to-one communications and local communications. The local communications happen in wireless sensor networks when sensor nodes exchange route or neighbor discovery information. The monitor node distinguishes the local communication by simply examining the source and destination node IDs in the messages captured by sniffing in the open channels. A packet is considered as local communication if neither the source or destination is the base station and the source and destination nodes are within each other's transmission range.

Inspired by previous studies on Internet end system behavior [20], we propose to generate *behavior fingerprint* of each sensor node at distributed monitors with a number of important properties of the sensor nodes: hops to the base station, neighbor list, number of initiated messages, number of forwarded messages, number of control messages of different categories such as route discovery, route reply and "hello" messages, and temporal dynamics of these properties.

Behavior fingerprint of sensor nodes could detect a variety of attacks in wireless sensor networks. For example, using the proposed approach one could detect the wormhole attack if the two endpoints of the wormhole show significant change of hop count to the base station. Another important dimension that is studied in behavior analysis is *temporal information*. In particular, we explore time series analysis techniques to analyze the behavior metrics over time for detecting behavior deviations of sensor nodes in the network. Therefore, with the behavior fingerprint, we could analyzer one-to-many communications, many-to-one communications and local communications and discover interesting events by exploring i) whether a node behavior matches a pre-defined attack signature, ii) wether the behavior of a specific node deviates from its neighbors by applying the *node similarity* principle), and iii) whether a node exhibits a significant behavior change.

The *base station* implements a behavior visualization component and an aggregated behavior analyzer that uses the behavior fingerprint metrics from distributed monitor nodes to form a global view of behavior profiles of sensor nodes in the network. Due to the visibility limitations of distributed monitor nodes, several attacks including *wormhole attacks* and *node replications* need to be analyzed by the base station which has a global view of the behavior activities of all the sensor nodes as well as more computing power and energy resources. The monitor nodes need to send the base station the following information of a forwarder: i) its next hops of the forwarding operations, and ii) the number of aggregated and forwarded messages for each next hop. Through data visualization analysis tools or graphic models, the base station could detect significant data forwarding between certain intermediate routing nodes that launch *wormhole attacks*.

In addition, the monitor nodes could report the number of unique messages sent from each sensor node to the base station, which could use *entropy measures* to find nodes with multiple messages in a given time epoch, which is a typical behavior characteristics of *node replications*. Therefore, we could leverage the global knowledge and a comprehensive collection of the behavior metrics at the base station to perform correlation analysis to detect anomalous behavior of compromised sensor nodes or outside attackers.

In this section, we give a high-level description of the behavior metrics, the monitoring models, and the principles for anomaly detection we identified for our behavior monitoring framework. To implement such a framework, there are many specific problems, for example, how to deploy the monitor nodes to satisfy the monitoring requirement, how to detect node similarity and change of behavior over time at the monitors nodes and the base station, and how to design condensed reporting message of small size but being reconstructable at the base station. In the remaining of this paper, we study one important problem, that is, how to strategically choose monitor nodes for local monitoring.

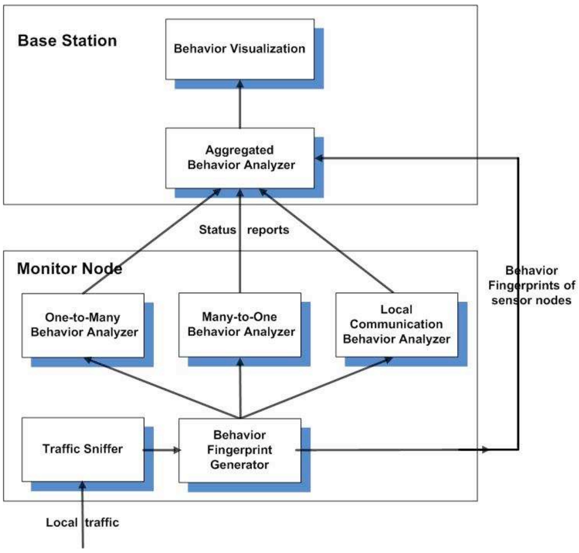

Base Station
Behavior Visualization
Aggregated Behavior Analyzer
Status reports
Monitor Node
One-to-Many Behavior Analyzer
Many-to-One Behavior Analyzer
Local Communication Behavior Analyzer
Behavior Fingerprints of sensor nodes
Traffic Sniffer
Behavior Fingerprint Generator
Local traffic

Figure 2. A prototype system of the behavior monitoring framework.

4 Monitor Node Selection Problem

A critical problem in the proposed behavior monitoring framework is how to select monitor nodes. Current work can be categorized into two types: One approach, which most recent work adopts, is to let any active sensing node act as a monitor node, that is, every active node can monitor its neighbors. The other approach is to create a monitoring infrastructure in parallel with the sensor network, i.e., the monitoring infrastructure consists of a separate set of monitor nodes (usually more powerful than sensor nodes) which communicate through separate channel from sensor communication channel. The first approach is cost-effective since it does not need to introduce extra monitor nodes. However, it puts extra work on the sensors. Due to the resource limitation of sensor node and the consistent monitoring requirement to monitor node, it might overload the sensor nodes since they take multiple tasks as sensing, routing, and monitoring at the same time. It is worthwhile to have separate monitor nodes in sensor networks. The second approach is very expensive and needs extra time to deploy the monitoring infrastructure.

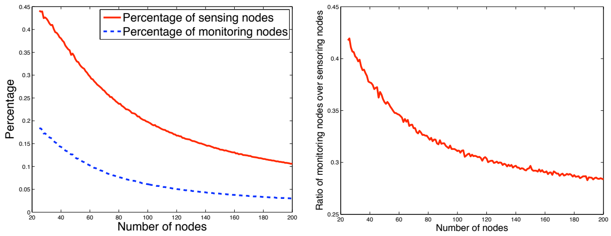
We propose a novel monitoring architecture called Dedicated Monitoring Set (DMS) which distinguishes from the current monitor node selection work in two aspects: 1) chooses a subset of sensors in the sensor network which are disjoint to the active sensing nodes as monitor nodes and 2) the monitor nodes utilize current routing trees maintained by the sensing/routing nodes in the sensor network. Note that in a densely deployed sensor network, to prolong the lifetime of the network, a duty cycle is introduced to allow each sensor switch between active and sleep mode to save energy consumption. Thus we can choose the monitor nodes from the nodes in sleep mode. This architecture balances between the current two approaches. It is cost effective since it utilizes the deployed sensors and avoid the routing overhead among monitor nodes. On the other hand, it separates the functionality of sensing/routing and monitoring, thus avoids overloading resource-constraint sensors in the sensor network.

In our behavior monitoring framework, we propose to let monitoring node creates condensed behavior profile to capture the behavior of a sensing node and make local decisions. Only aggregated behavior is reported to base station in order to minimize the monitoring traffic. Since for most of the time, the monitor nodes observe the network, it does not interference with the regular function of the sensor network.

4.1 Problem Formulation

The problem of finding a Dedicated Monitoring Set from the deployed sensor network can be defined as follows: Given a densely deployed wireless sensor network, if the set of active sensing nodes have been chosen, how to find a monitor node set from the deployed nodes which is disjoint from the sensing nodes but dominates the sensing node. That is, every sensing node has a monitor node within its transmission range to observe its behavior. In the following, we formulate the DMS problem with graph representation. In a sensor network, the actually topology graph consisting of all the active nodes is different from the deployed communication graph induced from all the deployed sensors.

In this paper, we call the topology graph G and the deployed communication graph G_c. The minimum disjoint monitoring set problem is formulated as below: Given a communication graph $G_c = (V_c, E_c)$ and a topology graph $G = (V, E)$ where $V \subset V_c$ and $E \subset E_c$, find a monitoring set $V_m \subset V_c - V$ such that every $v \in V$ is dominated//monitored by at least one node in V_m and $|V_m|$ is minimized.



(a) ratio of active sensing node and monitoring node to deployed nodes

(b) ratio of monitoring node over sensing node

Figure 3. Impact of node number on monitor node size

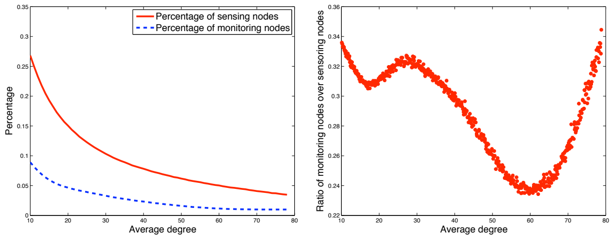
4.2 Greedy Algorithm

Given a deployed network, we propose a greedy algorithm for the DMS problem as below: first, calculate a connected Dominating Set (CDS) of all the deployed sensor nodes. The CDS represents all the active sensing nodes which can cover the whole area. Note that CDS is a solution for point coverage problem which an area coverage problem can covert to [21]; second, initialize the monitor set to be empty and associate a monitor impact metric with each of the in-active sensor. The monitor impact records the number of active sensing nodes which are neighbors of an in-active sensor and has not had a monitor in the monitor set; third, select an in-active sensor which has the largest monitor impact and add it to the monitor set; fourth, update the monitor impact of the remaining in-active sensors. Repeat step three and four until every active sensing node has a monitor in the monitor set. All the nodes in the monitor set act as the monitor node.

4.3 Simulation Results

Simulation is carried out to evaluate the size of monitor set for communication graphs of different scales and densities. Given a deployed network, first we calculate the active sensing node set, then calculate the monitor set which monitors the active sensing nodes. In the simulation, we randomly generate different connected network topologies. For each setting, we perform the simulation 1000 times and compute the average value.Given a fixed area, two sets of simulations are implemented. First, fix the transmission range of each node and vary the number of nodes in the area to evaluate the impact of node number on the active sensing node size and monitor node size. Second, fix the node number and vary the transmission range. For each transmission range, we calculate the corresponding average node degree and evaluate the impact of node density on the active sensing node size and monitor node size.

Impact of node number on monitor set: In this simulation, we randomly place 25 to 200 nodes in $1000 \times 1000m^2$ area and set node transmission range fixed at $250m$. Fig. 3 illustrates the impact of node number on the size of monitor set. In Fig. 3.(a), x-axis is the number of nodes and y-axis is the percentage of active sensing nodes to deployed nodes and the percentage of monitor nodes to deployed nodes. For example, when there are 100 nodes in the network which induces the average node degree 15.11, out of 100 nodes, 20 nodes are chosen as active sensing nodes, and 6 nodes are selected as dedicated monitor nodes to monitor these active sensing nodes. The proposed DMS algorithm shows a nice trend that as the number of nodes increases in the area, the percentages of active sensing nodes and the monitoring nodes over total nodes decrease correspondingly. An interesting observation in Fig. 3.(b) is that the ratio of monitoring nodes to sensing nodes decreases as the number of deployed nodes in the network increases. This is a nice feature since it implies that adding more nodes to the area can reduce the monitor set size.



(a) ratio of sensing node and monitoring node over deployed node

(b) ratio of monitoring node over sensing node

Figure 4. Impact of transmission range and node degree on monitor node size

Impact of node density on monitor set: In this simulation, we randomly place 100 nodes in an $1000 \times 1000m^2$ region. The node transmission range varies from $200m$ to $750m$, which causes the average degree ranges from 10 to 79. Fig. 4 shows the impact of average node degree on the size of monitor set. In Fig. 4.(a), x-axis is the average degree of the deployed 100 sensor nodes and y-axis is the percentage of sensing nodes over deployed nodes and the percentage of monitoring nodes over deployed nodes. As we can see, as average degree increases, less deployed nodes are chosen as sensing nodes and monitor nodes since one sensing node can cover more deployed area and one monitor node can cover more sensing nodes. An interesting discovery in Fig. 4.(b) is that the ratio of monitor node to active sensing node does not show consistent decreasing trend as node degree increases. This shows that, unlike increasing node number, increasing transmission range thus to increase the node density does not necessarily decrease the ratio of monitoring node to active sensing node.

In summary, the greedy algorithm can effectively construct a small-sized dedicated monitor node set and the ratio of the monitor nodes to deployed nodes decreases as the node number and node degree increase.

5 Conclusion

In this paper, we propose a behavior monitoring framework for sensor network. In the framework, we study and categorize the communication pattern in wireless sensor networks, combine local monitoring and global monitoring for energy efficiency and quick response, and utilize message symmetry and node similarity for the analysis of anomaly behavior. As a first step to build the framework, we study the monitor node selection problem and propose to select in-active sensor nodes as dedicated monitor node and introduce an algorithm to find a small-sized monitor set. In the future, we will implement the system and perform field experiments and behavior simulations to evaluate the performance and cost of the monitoring framework. Another important research direction is to ensure the security of the monitoring infrastructure and handle monitor node failures.

References

[1] X. Chen, K. Makki, K. Yen, and N. Pissinou, "Sensor network security: A survey," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 2, pp. 52–73, 2009.

[2] N. Ramanathan, K. Chang, R. Kapur, L. Girod, E. Kohler, and D. Estrin, "Sympathy for the sensor network debugger," in *Proceedings of International conference on Embedded networked sensor systems (SenSys)*, 2005.

[3] K. Liu, M. Li, Y. Liu, M. Li, Z. Guo, and F. Hong, "Passive diagnosis for wireless sensor networks," in *Proceedings of ACM conference on Embedded network sensor systems (SenSys)*, 2008.

[4] C. Hsin, and M. Liu, "A distributed monitoring mechanism for wireless sensor networks," in *Proceedings of ACM workshop on Wireless security*, 2002.

[5] S. Rost and H. Balakrishnan, "Memento: A health monitoring system for wireless sensor networks," in *Proceedings of IEEE SECON*, September 2006.

[6] Y. Yang, S. Zhu, and G. Cao, "Improving sensor network immunity under worm attacks: a software diversity approach," in *Proceedings of ACM international symposium on Mobile ad hoc networking and computing*, Hong Kong, China, May 2008.

[7] P. De, Y. Liu, and S. Das, "Modeling node compromise spread in wireless sensor networks using epidemic theory," in *Proceedings of International Symposium on on World of Wireless, Mobile and Multimedia Networks*, June 2006.

[8] C. Karlof, and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," in *Proceedings of IEEE International Workshop on Sensor Network Protocols and Applications*, May 2003.

[9] K. Sun, A. Liu, R. Xu, P. Ning, and D. Maughan, "Securing network access in wireless sensor networks," in *Proceedings of ACM conference on Wireless network security*, March 2009.

[10] F. Liu, X. Cheng, and D. Chen, "Insider Attacker Detection in Wireless Sensor Networks," in *Proceedings of IEEE INFOCOM*, May 2007.

[11] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. Culler, "Spins: security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, 2002.

[12] H. Tan, D. Ostry, J. Zic, and S. Jha, "A confidential and dos-resistant multi-hop code dissemination protocol for wireless sensor networks," in *Proceedings of ACM conference on Wireless network security*, March 2009.

[13] S. Hyun, P. Ning, A. Liu, and W. Du, "Seluge: Secure and DoS-Resistant Code Dissemination in Wireless Sensor Networks," in *Proceedings of IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, April 2008.

[14] H. Song, L. Xie, S. Zhu, and G. Cao, "Sensor node compromise detection: the location perspective," in *Proceedings of International conference on Wireless communications and mobile computing*, August 2007.

[15] K. Xing, F. Liu, X. Cheng, and D.H.C. Du, "Real-time detection of clone attacks in wireless sensor networks," in *Proceedings of International Conference on Distributed Computing Systems*, June 2008.

[16] I. Khalil, S. Bagchi, and C. Nina-Rotaru, "Dicas: Detection, diagnosis and isolation of control attacks in sensor networks," in *Proceedings of International Conference on Security and Privacy for Emerging Areas in Communications Networks*, December 2005.

[17] Y.C. Hu, A. Perrig, and D. Johnson, "Packet leashes: a defense against wormhole attacks in wireless networks," in *Proceedings of IEEE INFOCOM*, April 2003.

[18] R. Maheshwari, J. Gao, and S. Das, "Detecting Wormhole Attacks in Wireless Networks Using Connectivity Information," in *Proceedings of IEEE INFOCOM*, May 2007.

[19] B. Chen, G. Peterson, G. Mainland, and M. Welsh, "Livenet: Using passive monitoring to reconstruct sensor network dynamics," in *Proceedings of IEEE/ACM International Conference on Distributed Computing in Sensor Systems (DCOSS)*, June 2008.

[20] K. Xu, F. Wang, S. Bhattacharyya and Z.-L. Zhang, "A Real-time Network Traffic Profiling System," in *Proceedings of International Conference on Dependable Systems and Networks*, June 2007.

[21] Shuhui Yang, Fei Dai, Cardei, Mihaela, Jie Wu, "On multiple point coverage in wireless sensor networks," in *Proceedings of Mobile Adhoc and Sensor Systems Conference*.