ENERGY EFFICIENT HIGH PERFORMANCE WIRELESS SENSOR NETWORKS USING LOCAL MONITORING TECHNIQUES

[1]Bipin R,[2]S. Sudhakar Ilango,
[1]PG Scholar, [2]Assistant Professor,
Department of Computer Science and Engineering
Sri Krishna College of Engineering and Technology, Coimbatore
[1]bipin1680@gmail.com, [2]sudhakarilango@skcet.ac.in

Abstract

Energy consumption and security has becomes a primary concern in a Wireless Sensor Network. In multi-hop communications, nodes that are near a sink are responsible for forwarding data from nodes that are farther away. This makes the battery drain faster in sensor node closer to a sink while those nodes farther away may maintain more energy. This leads to non uniform depletion of energy which causes network partition and the formation of energy holes. As a result, the sink becomes disconnected from other nodes, there by impairing the WSN. Hence, preventing formation of energy holes and thus balancing the energy consumption of the sensor nodes is a critical issue in WSN. Several studies have demonstrated the benefits of using a mobile sink to prevent the formation of energy holes and to reduce the energy consumption of nodes in wireless sensor networks. Particularly in delay-sensitive applications, as all sensed data must be collected within a given time constraint. PANEL-ELMO is an energy aware algorithm for reducing the energy consumption and also increasing the security of the WSN. ELMO promises the operation of WSN in a manner that is both energy-efficient and secure. The PANEL protocol supports asynchronous sensor network applications where the sensor readings are fetched by the base stations after some delay. PANEL protocol supports reliable and persistent data storage applications, intra and inter-cluster routing while ensuring load balancing. The local monitoring is combined with more secure form of sleep-wake scheduling along with aggregator election. The new methodology enables sleep-wake management in a secure manner even in the face of adversarial nodes that choose not to awaken nodes responsible for monitoring their traffic.

Keywords: Compressive Sensing (CS), Wireless Sensor Network (WSN), sleep wake techniques, malicious nodes.

I.INTRODUCTION

Wireless Sensor Networks generally consist of a large number of sensor nodes which are spatially distributed to measure, collect and process information of interest with respect to a target area. They have been used in many applications such as climate, habitat, and infrastructure monitoring with appropriate temporal and spatial scales. There exist some bottlenecks in the successful deployment of wireless sensor networks. Primarily, when the number of sensor nodes increases, a large amount of data has to be processed, transferred, and stored at the fusion centre. Secondly, sensor nodes are generally deployed with limited energy capacity, computational capability and wireless bandwidth.

COMPRESSIVE SENSING

Compressive sensing is a sampling technique that takes advantage of the sparse characteristic of the natural physical signals and it allows recovering the signals with a reduced number of random samples [1], [2]. With respect to the temporal correlation and spatial correlation among the densely deployed sensor nodes, compressive sensing can be used as a data acquisition technique. This helps in reducing the operating cost of wireless sensor networks.

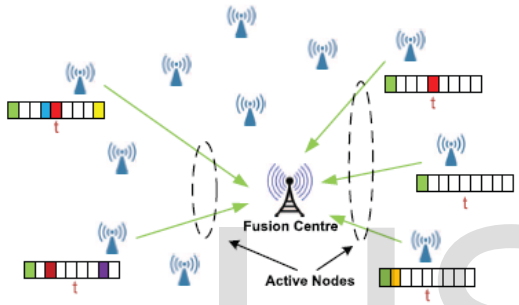

Fig.1. Representation of compressive sensing in wireless sensor network.

ELMO

The Energy Aware Local Monitoring (ELMO) methodology consists of a set of mechanisms that significantly reduce the node wake time required for monitoring. These mechanisms are derived from existing local monitoring techniques. Depending on the scenario, the ELMO mechanisms either makes modifications on the existing sleep protocol used in the network hence it is referred to as the baseline sleep-wake scheme (SWS), or it constitutes of a totally new protocol. In both cases the goal of the protocol is energy conservation while achieving the same level of security that was achieved with the baseline local monitoring (LM). When a network does not have a baseline SWS, then local monitoring is not modified since the goal is to reduce the impact of local monitoring on existing energy conservation schemes. ELMO does not impose any additional hardware requirements beyond what is used in the network. For networks that uses synchronized sleep algorithms (e.g., [3], [4], [5], [6]), nodes wake up and go to sleep in a synchronized manner, hence ELMO does not need to do anything since a node and its monitoring neighbours will be automatically awakened by the baseline SWS itself.

PANEL

PANEL, Position- based Aggregator Node Election protocol for wireless sensor networks. PANEL uses the geographical position information of the nodes to determine which of them should be marked as aggregators. Like other aggregator node election protocols, PANEL also ensures load balancing in the sense that every single node is elected aggregator nearly equally frequently. The PANEL protocol is unique from other aggregator node election protocols because it supports both synchronous and asynchronous applications.

In particular, the PANEL supports TinyPEDS (Tiny Persistent Encrypted Data Storage) [7], and other similar asynchronous sensor network applications. In TinyPEDS, aggregator node collects and aggregates sensor readings from the clusters they belongs to, and then persistently store the aggregated values. In order to increase reliability, the aggregators replicate their stored data at the aggregators for some selected backup clusters. These backup aggregators must be chosen in such a way that they are farther away from the primary aggregator and are kept at a certain distance called the disaster radius. The reason behind this is that if there is a disaster in which the primary aggregator is destroyed, its data will be still available and can be retrieved from the backup aggregators.

II EXISTING SYSTEM

COST AWARE COMPRESSIVE SENSING

CS Reconstruction

For a data vector $f \in {}_R{}^n$ can be represented by the sparse vector $x \in {}_R{}^n$ $(n \le \hat{n})$ by $f = \Psi x$, the CS measurement vector is given by

$$y = \Phi f + z = \mathrm{A}x + z \qquad (1)$$

Where $z \sim \mathrm{N}(0, \sigma^2 I_m)$ represents the sensing noise, $\Phi \in R^{m \times n}$ and $\mathrm{A} = \Phi\Psi \in R^{m \times \hat{n}}$ denotes the sensing matrix and the equivalent sensing matrix [8], respectively, and $m<n$.

Compressive Data Gathering

The activated sensor nodes measure the important signals and then encode their measurements into a packet which is then modulated and transmitted to the fusion centre in a conflict-free manner via time division multiplexing access (TDMA) or frequency division multiplexing access (FDMA). The received measurement vector $y \in R^m$ at the FC can be expressed as (1), where $\Phi \in R^{m \times n}$ denotes the activity matrix. The rows of the activity matrix Φ can be regarded as m rows of an $n \times n$ identity matrix, i.e., the entries are all zeros except for entries in different columns and rows, where the columns with 1s correspond to the active nodes.

Cost Aware Activity Scheduling

Conventional CS completely assumes that all measurements have the same cost, which might be suitable for many applications like magnetic resonance imaging (MRI) and compressive radar. Yet, the cost for conducting measurements and communication at different sensor node in a wireless sensor node can vary significantly. The sampling cost is directly computed for a given activity pattern, but it is difficult to quantify the reconstruction accuracy because the signal reconstruction accuracy of the CS systems are not tractable. The sampling cost of a compressive sleeping wireless sensor network consists of various factors including wireless spectrum resource and system considerations and not just limited to energy consumption and extra cost will be used for communicating active Ids with the FC. For example, the costs in terms of energy consumption and bandwidth are not uniform for the sensor nodes since different channel conditions are experienced at the sensor nodes. To sustain the function of the network with a lifetime as long as possible from the system point of view, a higher priority of activity is considered necessary for sensor nodes with sufficient energy reserves, or those with greater energy harvesting capability. The sampling costs of different sensor nodes are denoted by a vector $c \in R^n$ and various cost factors can be integrated into this vector. For instance, if energy consumption is the main concern, the sampling cost of each node can be identified by using an appropriate path loss model with respect to the distance from the fusion canter to each sensor node, else by using a feedback scheme in which the frequency of feedback depends up on the trade-off between overhead and the degree that the channel condition varies If crowd sourced signal strength data for the monitored area is available, the dynamic sampling cost can be obtained at almost zero cost.

The sampling costs that are occurring due to the limitations of the devices and the physical environment will generally have spatial and temporal correlations. By activating sensor nodes having the lowest costs will not necessarily improve the performance trade-off between sampling cost and reconstruction accuracy [9]. Hence, it is advisable to follow cost aware activity scheduling approaches in order to achieve a good performance trade-off for wireless sensor networks.

LOCAL MONITORING

Local monitoring is a combined detection strategy where a node monitors and

controls the traffic going in and out of its neighbours. This strategy was primarily developed for static sensor networks [10].

For a node α that can watch a node N_2, α should be a neighbour of both N_2 and the previous hop that is N_1. In such a case we say that α is a guard node for N_2 over the link $N_1 \rightarrow N_2$. The notation $R(N)$ is used to denote the set of all nodes which are within the range of node N and $G(N_1, N_2)$ to denote the set of all guard nodes of N_2 over the link $N_1 \rightarrow N_2$. Unless A itself is the destination the guards expect that A will forward the packet toward the ultimate destination. Each entry in the watch buffer is time stamped with a time threshold T_w, by which A must forward the packet. Each packet forwarded by A with X as a previous hop is checked in the watch buffer for the corresponding information. The check can be used to verify if the packet is fabricated or duplicated, corrupted, dropped or delayed and misrouted.

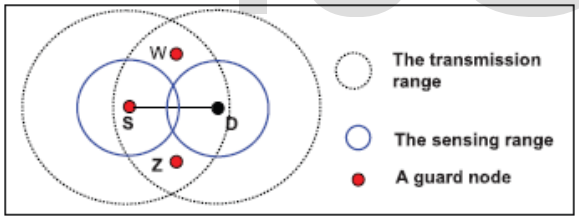

Fig.2. Relationship between communication and sensing ranges

A malicious counter ($MalC(i, j)$) is maintained at each guard node i, for a node j, at the receiving end of each link that i is monitoring over a sliding window of length T_{win}. ($MalC(i, j)$) is incremented for any malicious activity of j detected by i. When ($MalC(i, j)$) value crosses a threshold rate ($MalC_{th}$) over T_{win} node i revokes j from its neighbour list (called direct isolation, since it will henceforth not perform any communication with node j), and sends to each neighbour of j, an authenticated alert message indicating j is a suspected malicious node. When a neighbour N_i gets the alert, it verifies the authenticity of the alert message. When N_i gets enough alert messages about j, it marks the status of j as revoked (called indirect isolation). The notion of enough number of alerts is quantified by the detection confidence index. A node becomes isolated when all its first-hop neighbours revoke it either directly or indirectly.

AGGREGATOR NODE ELECTION

The aggregator node election procedure needs communications within the cluster. PANEL, Position based Aggregator Node Election takes advantage of these communications and uses them to establish routing tables for intra-cluster routing. In particular, at the end nodes also learn the next hop towards the aggregator elected for the current epoch.

PANEL also includes a position-based routing protocol that is used in inter-cluster communications. As the nodes are aware of their geographical position, this seems to be a natural choice that does not result in any further overhead. The position-based routing protocol is used for routing messages from a distant aggregator or distant base station towards the reference point of a given cluster. Once the message enters the cluster, the message further routed towards the aggregator using the intra-cluster routing protocol based on the routing tables which was generated during the aggregator node election procedure. Any position-based routing protocol can be integrated with PANEL.

III PROPOSED SYSTEM

The compressive sensing mechanism does not generally consider the energy conservation as a major aspect as it is mainly focused on cost reduction of message

transmission. This obviously reduces the life time of the overall wireless sensor network. The reduced life time of sensor nodes makes the wireless sensor network less reliable. There are several protocols which are designed to enhance the life time, security, processing and storage of the deployed node. In this paper, the compressive sensing wireless sensor network is enhanced by integrating some existing protocols which are indented to increase the life-time of the network by reducing the energy consumption and increase the security so that the wireless network are made more reliable and practical. As the protocols which are integrated with the compressive sensing mechanism are autonomous in nature no additional hardware changes are required.

ON-DEMAND ELMO

The basic idea of On-Demand ELMO is for a node to wake up the requisite guard nodes to perform local monitoring on the communication that is going out from that node. The challenge comes from the fact that any of the nodes may be malicious hence may not faithfully wake up the suitable guards.

On-Demand ELMO enables the guards to go to sleep when not required for monitoring. The approach used here is on-demand sleep-wake of the guards instead of scheduling the sleep-wake periods. The significant characteristic of on-demand sleep-wake protocols is that any node in the network can initiate communication with any other node in the network randomly. No fixed communication pattern in the network is needed for the sleep-wake protocol. OD-ELMO uses either passive antennas with circuitry that can harvest signal energy to trigger a node to wakeup [11] or low-power wake-up antennas (e.g., [12], [13], [14]).

GUARD SCHEDULING ALGORITHM

The guard scheduling algorithm is used to select the suitable set of guards (SSG) which are sufficient to monitor a certain communication link. Sufficient means enough number of guards to completely isolate a malicious node and it is dependent on the detection confidence index (γ) of local monitoring. The number of selected guards should be at least equal to γ. Suitable means the selection of the guards which are able to detect the malicious activity even under the transmission power level control attack [15].

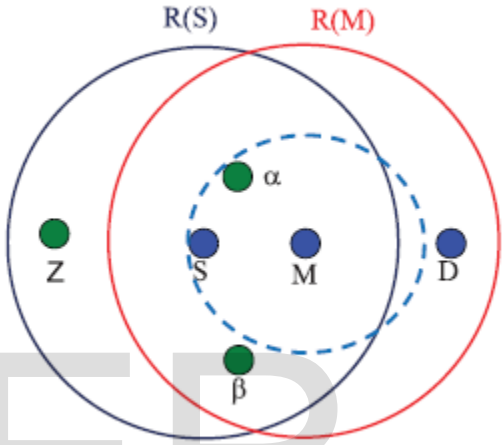

Fig.3. transmission power control attack.

To illustrate the idea, consider Fig. 3, where *S* and *D* are the source and destination nodes, α, β and *Z* are the guard nodes, *M* the malicious node and *R(S), R(M)* are the range of source and destination nodes respectively. By controlling the transmission power level (the dotted circle in Fig. 3) to exclude the next-hop node D the malicious node M tries to drop a packet without being detected. Node M succeeds if the selected guard is α and fails if the selected guard is β. Node α is included within the reduced transmission range and falsely thinks that M faithfully relayed the packet. However, α is out of the reduced range and correctly accuses M of dropping the packet. Thus, the guard β would be more suitable to monitor this link (*S*→*M*) than the guard α.

The GS algorithm shown in Fig. 4 is performed by a node, say *S*, to select the sufficient and suitable set of guards to monitor a current receiver over the link

CurrentSender(CS)→CurrentReceiver(CR).
We use *FS(i)* to refer to the list of first-hop and second-hop neighbors' locations of node *i (Li)*. Moreover, we use *NH* to refer to the next-hop node from the current receiver *(CR)*.

```
GS (γ, CS, CR, NH, FS(S))
{
SSG = {};
#Find the list of all possible guards (LG)
LG = G (CS, CR);
#To balance guard responsibility
Randomize the entries in LG;
If (|LG| <= γ)
  SSG = LG;
Else
 {
   TGS = LG;
   While (|SSG| < γ && |TGS| > 0)
   {
                g = first element in TGS;
                If (Distance (CR,g) >= Distance(CR,NH))
                    Add g to SSG;
                Remove g from TGS
   }
 }
Return SSG;
}
```

Fig.4. Guard Scheduling Algorithm

PANEL

Based on the reference points generated, the nodes starts start the aggregator node election procedure. Each node *i* set a timer, the expiration time of which is proportional to the distance $D(\vec{P}_i, P_j)$ between the node's position $\vec{P}_i$ and the reference point $\vec{R}_j$ of its cluster. When this timer expires, the node broadcasts a message with maximum power in which it announces itself as the aggregator unless the node heard such an announcement from another node before its timer expires. The announcement message has the following format:

[*type* | *epoch* | *id* /*pos*]

Where *type* is announcement, *epoch* is the current epoch number, and *id* and *pos* are the identifier and the position of the originator of the announcement, respectively.

When a node gets an announcement, it verifies if the originator of the announcement is closer to the reference point than the node known to be the closest so far. If so, then the node records the originator of the announcement as the candidate aggregator, and re-broadcasts the announcement. Moreover, if the node still has its timer active, then it cancels it. Otherwise, the node silently discards the announcement. Announcements that belong to other clusters are also discarded in order to limit the propagation of an announcement within the cluster that it is concerned with.

The closest node to the reference point first sends its announcement so there are chances for this single announcement to get flooded inside the cluster. So in most cases, each node re-broadcasts a single message during the aggregator election procedure. But in some cases depending on the topology of the network more than one node send their announcements. In those cases, only the announcement originated by the node that is the closest to the reference point will "survive", hence those announcement will be received and recorded by every node in the cluster.

After some time *T*, the aggregator node election phase stop and each node will consider the recorded candidate aggregator as the aggregator for the current *epoch*. The value of *T* depends on the time needed for a flooded message to cover the maximum distance within the cluster. This ensures that at the end of the aggregator election phase, each node have received the announcement of the future aggregator.

IV SIMULATION RESULTS

In this paper to simulate the data exchange protocol over a network with local monitoring enabled, ns-2 [17] is used.

Output Metrics

1. *Average end-to-end delay* – the time taken for a data packet to reach the end

destination averaged over all successfully received data packets;

2. *Packet delivery ratio* – the ratio of number of packets delivered to the destination to number of packets sent by a node averaged over all the nodes;
3. *Authentication security* – the rate of successful secured communications;
4. *Throughput* – the maximum number of energy efficient communications.

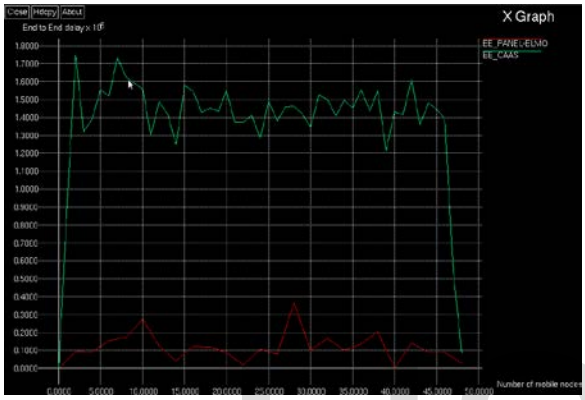

Fig.5. End to end delay

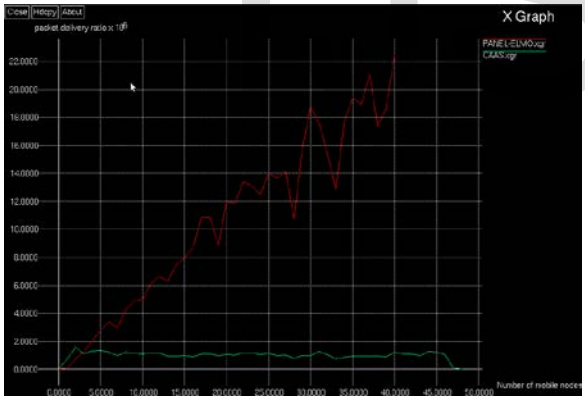

Fig.6. packet delivery ratio

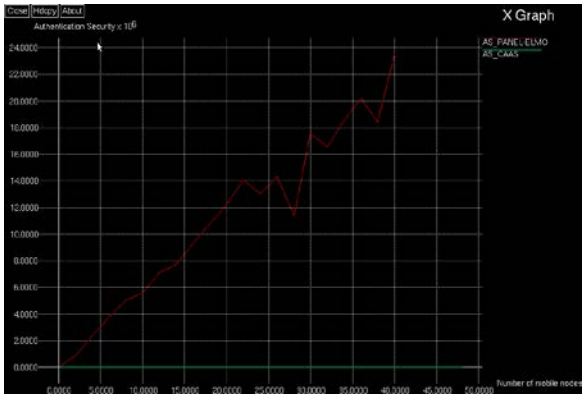

Fig.7. Authentication security

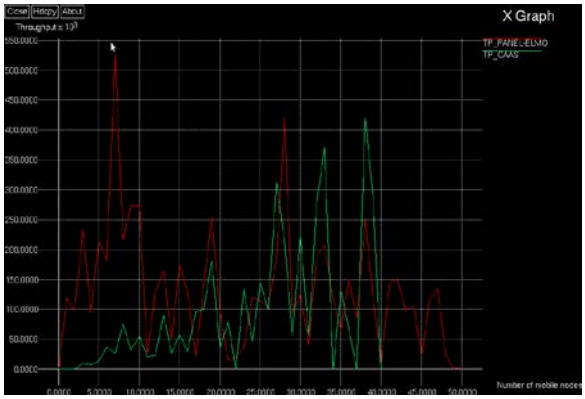

Fig.8. Throughput

V CONCLUSION

In this paper the ELMO protocol provides high-performance energy-efficient local monitoring for wireless sensor networks. ELMO has three manifestations which correspond to the three classes of sleep wake schemes. For the first class (synchronized sleep-wake WSNs), the local monitoring needs no modification. For the second class (continuously acting WSNs), the local monitoring can call the baseline sleep-wake scheme (SWS) with modified parameter values. The third class (triggered WSNs), which is the most demanding case in terms of the needed adaptation of local monitoring. The extensive simulation result shows the newly integrated system provides greater energy efficiency in all the above cases along with improved message delivery rate and security against malicious node attacks. The energy savings capability of the combined PANEL-ELMO is on the order of twenty to hundred times depending on the network parameters. The local monitoring is combined with more secure form of sleep-wake scheduling along with aggregator election. The new methodology enables sleep-wake management in a secure manner even in the face of adversarial nodes that choose not to awaken nodes responsible for monitoring their traffic.

REFERENCES

[1] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," IEEE Trans. Inf. Theory, vol. 52, no. 2, pp. 489–509, 2006.

[2] D. Donoho, "Compressed sensing," IEEE Trans. Inf. Theory, vol. 52, no. 4, pp. 1289–1306, 2006.

[3] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks," Proc. MOBICOM '01, 2001.

[4] W. Ye, J. Heidemann, and D. Estrin, "An Energy Efficient MAC Protocol for Wireless Sensor Networks," Proc. IEEE INFOCOM, pp. 1567-1576, 2002.

[5] R. Naik, S. Biswas, and S. Datta, "Distributed Sleep-Scheduling Protocols for Energy Conservation in Wireless Networks," Proc. 38th Ann. Hawaii Int'l Conf. System Sciences (HICSS), pp. 285b-285b, 2005.

[6] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless Sensor Networks for Habitat Monitoring," Proc. ACM Int'l Workshop Wireless Sensor Networks and Applications, pp. 88-97, 2002.

[7] J. Girao, D. Westho®, E. Mykletun and T. Araki. TinyPEDS: tiny persistent encrypted data storage in asynchronous wireless sensor networks. Elsevier Ad Hoc Networks, June 2006.

[8] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," Construct. Approx., vol. 28, no. 3, pp. 253–263, 2008.

[9] L. Xu, X. Hao, N. D. Lane, X. Liu, and T. Moscibroda, "Cost-aware compressive sensing for networked sensing systems," in Proc. ACM 14th Int. Conf. Inf. Process. Sensor Netw., 2015, pp. 130–141.

[10] I. Khalil, S. Bagchi, and N.B. Shroff, "LITEWORP: Design and Analysis of a Protocol for Detection and Isolation of the Wormhole Attack in Multihop Wireless Networks," Proc. Elsevier Computer Networks J., vol. 51, no. 13, pp. 3750-3772, Sept. 2007.

[11] C. Guo, L.C. Zhong, and J.M. Rabaey, "Low Power Distributed MAC for Ad Hoc Sensor Radio Networks," Proc. IEEE Global Telecomm. Conf. (GLOBECOM '01), pp. 2944-2948, vol. 5, 2001.

[12] C. Guo, L.C. Zhong, and J.M. Rabaey, "Low Power Distributed MAC for Ad Hoc Sensor Radio Networks," Proc. IEEE Global Telecomm. Conf. (GLOBECOM '01), pp. 2944-2948, vol. 5, 2001.

[13] J. Rabaey, J. Ammer, T. Karalar, S. Li, B. Otis, M. Sheets, and T. Tuan, "Picoradios for Wireless Sensor Networks: The Next Challenge in Ultra-Low-Power Design," Proc. Int'l Solid-State Circuits Conf., pp. 200-201, 2002.

[14] J. Silva, J. Shamberger, M.J. Ammer, C. Guo, S. Li, R. Shah, T. Tuan, M. Sheets, J.M. Rabaey, B. Nikolic, A. Sangiovanni-Vincentelli, and P. Wright, "Design Methodology for Picoradio Networks," Proc. Conf. Design Automation and Test in Europe, pp. 314-323, 2001.

[15] I. Khalil and S. Bagchi, "MISPAR: Mitigating Stealthy Packet Dropping in Locally-Monitored Multi-Hop Wireless Ad Hoc Networks," Proc. Fourth ACM Secure Comm., pp. 1-10, 2008.

[16] Issa M. Khalil, "ELMO: Energy Aware Local Monitoring in Sensor Networks", IEEE Trans. on

Dependable and Secure Comm., vol. 8, no. 4, 2011

[17] "The Network Simulator ns-2," www.isi.edu/nsnam/ns/, 2011.