A Trust-Based Geographical Routing Scheme in Sensor Networks

Ka-Shun Hung, King-Shan Lui, and Yu-Kwong Kwok
Department of Electrical and Electronics Engineering
The University of Hong Kong
Pokfulam Road, Hong Kong SAR, China

***Abstract*— Devices in a sensor network need to work in a hostile environment and they are usually powered by batteries. Yet the whole purpose of deploying a sensor network is to perform distributed collaborative computing, possibly in a massive scale. In a hostile computing environment, the sensor devices might be routinely tampered with. Together with the possibility of faulty devices due to extreme conditions or low power, the trustworthiness of a device varies. Specifically, a device should only communicate with another device which has a trust level above a certain threshold. However, setting up trusted communication channels among sensor devices remains a major challenge.**

In this paper, we propose a trust-based routing scheme in sensor networks for providing a high level of robustness in node selection based on packet trust requirement with lifetime consideration. Our protocol allows messages to be routed through malicious and faulty devices with the selection of trusted neighbors. On the other hand, the network lifetime can also be prolonged by selecting those with their sensing functions covered by some existing nodes. Simulation results show that our scheme is possible to prolong the lifetime of sensor networks and maintain certain satisfactory delivery ratio.

I. Introduction

More often than not, sensor networks are deployed in a hostile environment. The hostility manifests in many aspects: energy deficiency, processing power limitations, poor physical conditions (e.g., high temperature, high humidity, etc.), non-existence of physical security (i.e., devices are easily tampered with), etc. One of the many implications of the hostility is that some sensor devices may not be trustworthy after deployment, possibly because the devices may become malfunctioned due to poor physical conditions, or may be deliberately altered by some human intruders.

However, devices in a sensor network need to cooperate in order to achieve their goals in the deployment. For instance, the devices in a sensor network usually need to report events to facilitate higher level decision making. Thus, the sensor devices have to communicate among each other from time to time. Obviously, data items sent from a corrupted or malfunctioned device should not be used. Simply put, a sensor device should differentiate between "good" devices from "bad" devices so that it only communicates with the former but not the latter. Unfortunately, traditional sophisticated technologies (e.g., the use of public key cryptography and certificates in authentication) are generally considered as infeasible due to the physical and processing limitations in a sensor device. In this paper, we consider using a trust/reputation-based communication system in a sensor network with lifetime consideration.

With the help of a reputation system, a mutual trust relationship between neighboring devices can be developed. Thus, a trust-based communication path can be set up. Data packets can then be sent from a source node to a destination node, with a higher confidence of successful delivery and data integrity. The key design questions in building a reputation system for trusted communication in a sensor network are: How the trust level is calculated? Where the trust level is stored? How to use the trust levels in constructing a communication path?

There are some obvious constraints in a sensor network bounding the possible design choices. For one thing, sensor devices are deficient in processing and storage power so that it is infeasible to implement sophisticated trust level calculation. Furthermore, it is infeasible to designate a sensor device to serve as a centralized server for storing trust levels. More importantly, sensor nodes have energy-constraints and different deployment location will have different effects to the whole network. As a result, one more design question in building up the system is: effect of the death of this neighbor sensor node to the whole network?

Before we introduce our proposed ideas to tackle these design problems, let us review previous works in reputation systems and network lifetime.

II. Related Work

In ad hoc networks, Watchdog and Pathrater [9] can be regarded as one of the earliest works in trust-based routing schemes. It is mainly focused on detecting the not-forwarding (selfish) behavior in ad hoc networks. Under the watchdog mechanism, the current node sends a packet to a neighbor node and then sets the wireless network card into promiscuous mode, so that the current node can overhear the packet sent by the neighbor node. The pathrater mechanism is then used to select path based on the maximum value of the average rate of all the nodes in the whole path provided by the watchdog mechanism. CONFIDANT [3], [4] is then proposed with the addition of a trust manager and a reputation system which can be used to evaluate events reported by the watchdog mechanism [9] and recommendations from neighbor nodes.

On the other hand, Pirzade *et al.* worked on both trust model and trust-based reactive ad hoc routing. They classified trust as aggregate trust and situational trust which involved five

categories [17]. Then, they proposed the idea of integrating two situational trusts (*Acknowledgment* which is similar to the idea of the watchdog mechanism and *Precision* which makes sure that packets are unmodified) to select route to destination based on the modified Dijkstra's algorithm with trust as the routing metrics [18], [15]. They compare the performance of integrating trusts to different kinds of common reactive routing protocols [16] e.g. DSR [7], AODV [13] and TORA [11].

There are also other systems including CORE [10], SORI [5], Context-aware detection [12], etc. Unfortunately, the trust-based routing systems suggested in this context are usually extended from some traditional wireless ad hoc routing protocols. In wireless sensor networks, a position based routing protocol is adopted instead. New secure routing mechanisms, thus, have to be developed to cope with the constraints of sensor networks.

Tanachaiwiwat *et al.* [19] targeted on the sensor network with trust-based geographical routing protocol based on some existing geographical routing protocols, e.g., GPSR [8]. They suggested that the communications in sensor networks are query-reply based and should be initiated by a sink node. As a result, the sink node will send packets only to the nodes with trust value over a certain threshold in the request for certain location. If the sink node does not get any replies from that location several times, it will initiate a search for the location of malicious node location using probing. Once the malicious location is known, the sink node can take two approaches. The first is that the sink installs the information about malicious node location to its neighbor explicitly and the second is that the sink sends the information of malicious node location within the packet. However, they assume that sensor networks are using query-reply model. We adopt a more generic approach in this paper that the data can be sent from one sensor node to another sensor node directly instead of coupling coupling as a query-reply pair.

Abu-Ghazaleh *et al.* [1] suggested resilient geographic routing in sensor networks with two approaches — location verification algorithm and trust-based route selection. They suggested the use of fixed increment and decrement trust values for awarding and punishing node forwarding behaviors. In route decision, similar to Tanachaiwiwat *et al.*, they suggested the formation of a forwarding set of neighbor nodes in which the forwarding set is a set with neighbor nodes geographically closer to destination and with trust value larger than a certain threshold value. Each candidate neighbor node is assigned with selecting probability based on weighted trust values of all neighbors in the set. Then, K neighbor nodes will be selected for forwarding the packet with the use of roulette wheel selection techniques. As a result, multipath routing for resilient can be resulted. Unfortunately, Abu-Ghazaleh *et al.* proposed the security framework in sensor networks without any performance analysis or comparison. The value of K in neighbor selections may result in large number of duplicate packets in the network. Also, the suggestion of verification with the use of signature may not be energy efficient and practical.

On the other hand, there are quite a large number of researches focusing on prolonging network lifetime of the sensor networks. Different researches have different definition of the network lifetime. Some defines lifetime as the first unsuccessful fulfillment of the requests, the first death of a node in the network, the cluster of nodes from the network, the unsuccessful coverage of points, areas, etc. Most researches focused on the sleeping schedules to prolong lifetime. In this paper, we define lifetime as the time that the network can maintain a certain percentage of area being covered and similar definition can also be found in [20], [6].

III. System Model

A. Assumptions

The following assumptions are made in our design:

- Every node knows its physical location and packet destinations are specified as physical locations.
- Nodes are assumed to have limited energy for routing messages in terms of number of messages sent.
- Nodes are fully connected.
- Nodes are assumed to have authenticated neighboring nodes through some sensor network authentication measures e.g., μTestla [14] or the variations of it [2].
- Nodes are assumed to be available all of the time.
- Neighbor nodes will broadcast their neighbors' locations in their beacons.

B. Network Lifetime

In sensor networks, the main functions of sensors are sensing the environment. The sensor nodes should have certain sensing range with radius r. They usually cooperate to sense events. In order to have correct functioning of the sensor networks, sensor nodes should have certain threshold of aggregate sensing coverage area A_T which is a fraction of total area of interest. This area may change due to different kind of applications which may have certain requirements on aggregate sensing coverage for accuracy or usability of the networks. Aggregate sensing coverage area A is defined as the $\bigcup_i A_i$ where i is the normal sensor node which is still alive and A_i is the sensing coverage area of node i. We define the lifetime of the network as the time of maintaining the aggregate sensing coverage area over A_T.

C. Trust Value and Packet Trust Requirement

In our design, *trust value* [19], [1] specifies the degree of trustworthiness in forwarding packets. Trust is defined on the link level, that is, between each pair of neighbors. Trust is defined as the level of trustworthiness of a particular node j as perceived by its neighbor node i. In other words, trust value of j w.r.t. i indicates how likely j would forward a packet sent by i. Consequently, the perceived trust value of node j could be different for two different nodes i and k, even though nodes i and k are both neighbors of j. Trust value is adjusted based on packet delivery and packet dropping events. The adjustment mechanism will be described in detail in the next subsection.

Packet Trust Requirement is used to quantify the level of trustworthiness of the links of the message forwarding path.

This is adapted from [15] in which routers are set to a certain threshold to filter the untrustworthy route replies. In our design, each packet carries its own requirement in its header. All the links on a forwarding path must have a trust value no less than the packet trust requirement. In other words, a node i cannot forward a packet of requirement r to neighbor j if the trust value of j w.r.t. i is less than r.

D. Basic System Components

There are three components involved in our system as shown in Figure 1. *Trust and Sensing Coverage Database* keeps the trust values and sensing coverage values of neighbors and provides information for finding a neighbor for packet forwarding in *Route Computation*. After identifying the next hop neighbor for forwarding a packet, *Overhearing* module overhears whether the packet is forwarded and relays the information back to the trust and sensing coverage database to adjust the trust value of the neighbor accordingly.

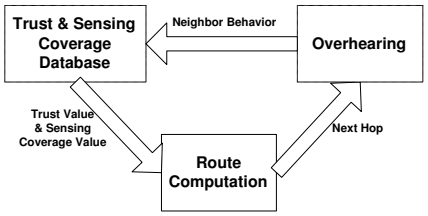

Fig. 1. Proposed architectural model

1) Neighbor Trust and Sensing Coverage Database: The trust and sensing coverage database is kept in each node i within the network. The trust database keeps all necessary information for trust value adjustment and path forwarding. There are two main pieces of information stored in the trust database for each neighbor j.

- Trust Value $T_{i,j}$: the trustworthiness of j perceived by i
- Packet History: a history of the packets sent to j. The necessary information for packet k is the packet trust requirement ($P^i_{j,k}$) and whether j has sent it out ($S^i_{j,k}$). $S^i_{j,k}$ = 1 if j sent out packet k; otherwise, $S^i_{j,k}$ = 0.

Let N be the history size, trust value $T_{i,j}$ which will be updated after getting the feedback from the overhearing module is defined as

$$T_{i,j} = \frac{\sum_{k=1}^{N} P^i_{j,k} \times S^i_{j,k}}{\sum_{k=1}^{N} P^i_{j,k}} \quad (1)$$

Equation 1 is developed based on the observation that every message carries a packet trust requirement to indicate the security level that the packet needs in the whole path from the source to destination. Both trust value and packet trust requirement range from 0 to 1. When compared with the equations in [19], [1], [17], the major difference is that packet trust requirement is also considered in the calculation of the trust value of the neighbors. The messages which require higher security level are expected to embed with a higher packet trust requirement value, indicating the importance of the messages. As a result, in successful forwarding of a more important message, trust value on a certain neighbor node is expected to increase higher and vice versa. On the other hand, if j sends out more packets, $T_{i,j}$ will be higher. Therefore, $T_{i,j}$ reflects how likely j would forward a packet sent by i.

On the other hand, the sensing coverage of the neighbor can also be calculated based on the beacons of the neighbors. Since neighbor nodes are assumed to forward their neighbors list, the current node can then calculate the sensing coverage of that particular neighbor based on the location information of the neighbors list. However, if a particular neighbor is untrustworthy, it will be removed from the list. Suppose node i has two neighbor nodes j and k, from the beacon of node k, node i knows that node k has no other neighbors and so the sensing area of node k that is uncovered by other nodes is A_k as shown in Figure 2 and this value will be stored in i's sensing coverage database.

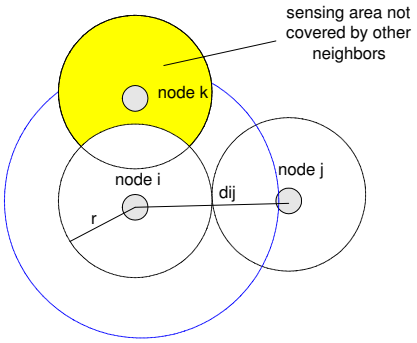

Fig. 2. Uncovered neighbor sensing coverage area

However, if the neighbor's trust value is below certain threshold, the information provided by that neighbor is assumed to be untrustworthy and the sensing coverage of the untrustworthy neighbor will become zero.

2) Route Computation: The main function of this component is to make decision on which next hop to forward a packet based on the trust value and neighbor sensing coverage value provided by the neighbor trust and sensing coverage database and the packet trust requirement value. Based on GPSR [8], a node should forward the packet to a neighbor which is closer to the destination than itself. Of course, this neighbor should have a trust value larger than the trust requirement value which is defined by the application. On the other hand, intuitively, the sensing coverage of this neighbor is preferred to be as minimal as possible, so the effect of node out of energy will be minimal to the whole network. However, selection of next hop bases only on the trust value or neighbor sensing coverage value may result in selection of the next hop very close to the current node. This results in higher hop count to the destination and more risky to the packet.

Therefore, the basic idea of our neighbor selection algorithm is to find a neighbor closer to the destination and with trust value higher than the packet trust requirement. Then, a filter distance will be calculated to filter the nodes that are too close to the source nodes. The initial value of the filter distance

is calculated based on the intuition that in order to achieve the expected shortest hop count from the current node to the destination node, each hop should pass through an average distance. Suppose the distance of the current node to the destination is 30, with the transmission range of 18, the lowest possible hop count will be 2. In this case, the filter distance will be 15. i.e. every hop is expected to be 15 units closer to the destination. With the fulfillment of this criterion, the nodes with minimal sensing coverage value will be chosen as the next hop. If no neighbor nodes can fulfill this requirement, the expected hop count will be incremented by 1 and a new filter distance will be calculated. This process will be continued until a hop fulfilling the criterion is found or all the nodes closer to destination are considered. In the first case, the message will be forwarded to the next hop. In the latter case, a modified trust-based perimeter mode will be used to select the next hop. Since [19], [1] do not explicitly suggest how to handle void, trust-based perimeter mode is modified by forming trust RNG graph, i.e., RNG links will be formed only with those neighbors which have trust value over packet trust requirement value. In this case, the next hop will be the neighbor in trust RNG graph according to right hand rule.

3) Overhearing: After the next hop neighbor is selected for the message, the message will be forwarded to the neighbor based on the decision made by the route computation component. Then, the overhearing component will track on the neighbor's transmission to see whether the message has been forwarded. If the message is not forwarded within a certain time, the message is assumed to be dropped. The trust database component will then be informed with the neighbor's current behavior.

IV. Simulation Results

We generate 10 random network topologies in our self-written simulator, each with 400 nodes. The nodes are randomly distributed in a board size of 125×125 with transmission range of 18. A certain percentage of nodes are chosen to be malicious nodes randomly. The malicious nodes do not contribute to the aggregate sensing coverage and will drop messages probabilistically from 50% to 100%. The nodes are generated with about 22 neighbors on average. In most researches, the sensing range of each node is assumed to be halved that of the transmission range. i.e. 9 in this case.

All nodes are assumed to be able to send or forward at most 500 messages. After that, nodes will become out of energy and will be disconnected from the network. Each time, a source node and a destination node will be chosen for a message. For fair comparison, all algorithms will use the same set of topologies under the same source-destination pairs until the network lifetime of that algorithm ends. The source-destination pairs are ensured to be connected by a path by Dijkstra's algorithm. As stated, the network lifetime is defined to be the number of messages that can be transferred in the network until the aggregate sensing coverage is below a certain threshold. In this simulation, 0.75 of the board size will be used as the threshold.

We simulate four routing algorithms in our simulation. The first one is the normal GPSR [8] algorithm. The second one is the algorithm adapted from [19]. This algorithm will only route the message over links with trust values above the threshold from a source node to a destination node using GPSR. We refer this algorithm as *Trust Threshold* in the figures. The third one is the algorithm adapted from [1]. This algorithm would first find the forwarding neighbor set which refers to a neighbor set which has distance closer to the destination than the current node. Then, a neighbor node will be selected with a probability based on the relative trust value in the forwarding neighbor set (i.e. a higher trust value has a higher probability to be selected). In this algorithm, we set $K = 1$ [1], i.e., single path only. We refer this algorithm as *Trust Probabilistic* in the figures. The fourth one is the algorithm proposed in this paper. We refer our algorithm as *Trust Lifetime* in the figures. Since we are interested in the effects of the routing algorithms to the network lifetime, all the trust-based routing algorithms suggested are using the trust calculation formula suggested in Equation 1 with the same packet trust requirement value.

In order to have fair comparison, all trust-based algorithms, except GPSR, used the same modified trust-based perimeter mode in case void occurs. However, due to the possibility of asymmetric trust values between nodes[2], a loop detection technique is implemented to detect these cases. The source-destination pairs will be redrawn whenever a loop occurs. In the simulations, the packet trust requirement is set to be 0.35 which is equal to the trust threshold value. At the same time, for better fit the sensor network memory constraints, the history window size N is set to 10.

Figure 3 shows that GPSR can achieve the best lifetime if delivery ratio is not considered. Since GPSR does not detect any malicious dropping, the higher the percentage of malicious nodes, the larger the amount of malicious dropping of the messages and this will result in longer lifetime as the number of nodes involved are minimal. However, malicious nodes do not contribute to aggregate sensing coverage, and the higher the percentage of malicious nodes, the lower the percentage of normal nodes out of energy will result in the failure of the whole network. With these two balancing effects, GPSR's lifetime is increasing with the increase in the malicious nodes percentage at the beginning and decreasing when the malicious nodes percentage further increases. Also due to the reason of the dropping of messages, GPSR can have longest lifetime if delivery ratio is not considered. However, if delivery ratio is considered, our algorithm outperforms other two algorithms in most of the malicious nodes percentages. It is because our algorithm considers the effect of out of energy of the neighbor node to that of aggregate sensing coverage and gives preference to the trusted node with the minimal sensing coverage. As a result, it performs better than *Trust*

[1]Refer to Section II for the meaning of K

[2]The solution of asymmetric problem in face routing is out of our current context and left for future work

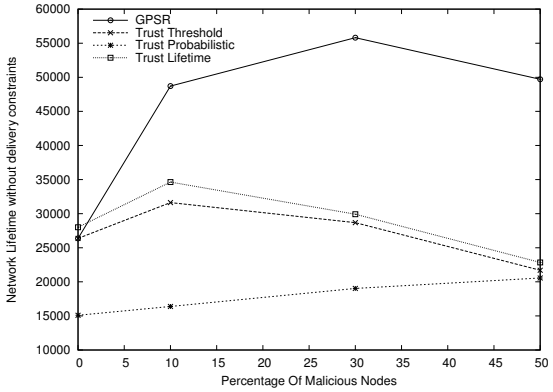

Fig. 3. Network lifetime without delivery constraints vs. percentage of malicious nodes in 0.75 fractional coverage

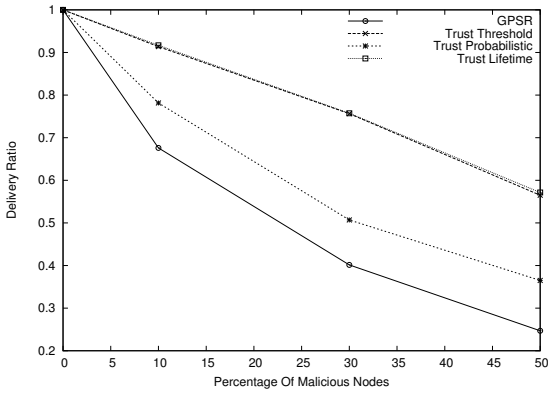

Fig. 4. Delivery ratio vs. percentage of malicious nodes in 0.75 fractional coverage

Threshold. For *Trust Probabilistic*, the algorithm will select the neighbor nodes closer than the current node to the destination probabilistically based on their trust values. As a result, it will select some nodes close to the source node and this results in longer hop counts in the path, so the energy of the nodes in the path will use up faster. This results in the minimal lifetime in this algorithm.

Figure 4 shows that the performance of GPSR is poorest with only 65% of successful delivery ratio in 10% of malicious node, as no mechanisms are used to detect malicious nodes. However, both our algorithm and *Trust Threshold* algorithm achieve similar performance and can have delivery ratio of more than 90% in 10% of malicious nodes. At the same time, both our algorithm and *Trust Threshold* algorithm outperforms the *Trust Probabilistic* algorithm as the *Trust Probabilistic* algorithm will probabilistically select neighbor nodes based on their trust value which is unknown at the beginning. This results in higher probability of selecting malicious nodes which are unnecessary to explore throughout the lifetime of that node.

V. Conclusions and Future work

In this paper, we describe a trust-based routing scheme that finds a forwarding path based on the trust requirement of a packet, the trust level of neighbor nodes and the sensing coverage of neighbor nodes. Our proposed algorithm achieves better performance in terms of network lifetime as it has considered the effect of certain nodes out of energy to the whole sensing function of the networks. Also, with the help of packet trust requirement, it is possible to allow applications to enjoy the flexibility of security level adjustment so as to meet their requirements. In the future, we would like to explore how to identify and defense more sophisticated malicious behaviors and include more factors (e.g., energy) in consideration of forwarding node. We would also like to investigate how to intelligently assign trust requirements to packets in order to balance the effects of delivery ratio and network lifetime.

References

[1] N. Abu-Ghazaleh, K. Kang, and K. Liu, “Towards Resilient Geograhic Routing in WSNs,” *ACM MSWiM*, Oct. 2005.

[2] M. Bohge and W. Trappe, “An Authentication Framework for Hierarchical Ad Hoc Sensor Networks,” *ACM WiSE*, 2003.

[3] S. Buchegger and J. L. Boudec, “Self-Policing Mobile Ad Hoc Networks by Reputation Systems,” *IEEE COMM. Mag.*, Jul. 2005.

[4] S. Buchegger and J.-Y. Le Boudec, “Performance Analysis of the CONFIDANT Protocol: Cooperation of Nodes Fairness In Dynamic Ad-hoc Networks,” *Proc. IEEE/ACM Symp. Mobile Ad Hoc Net. and Comp.*, Lausanne, Switzerland, June 2002.

[5] Q. He, D. Wu, and P. Khosla, “SORI: A Secure and Objective Reputation-Based Incentive Scheme for Ad Hoc Networks,” *WCNC 2004*, Atlanta, GA, Mar. 2004.

[6] C. Hunag, L. Lo, T. Tseng, and W. Chen, “Decentralized Energy-Conserving and Coverage-Preseving Protocols for Wireless Sensor Networks”, *ACM Tran.On Sensor Networks*, May 2006.

[7] D. B. Johnson and D. A. Maltz, “Dynamic Source Routing in Ad-Hoc Wireless Networks,” *Mobile Computing*, T. lmielinski and H. Korth, Eds., Kluwer, 1996, pp. 153-81.

[8] B. Karp and H. T. Kung, “GPSR: Greedy Perimeter Stateless Routing for Wireless Networks,” *ACM MOBICOM*, 2000.

[9] S. Marti, T.J. Giuli, K. Lai, and M. Baker, “Mitigating Routing Misbehavior in Mobile Ad Hoc Networks,” *ACM MOBICOM*, 2000.

[10] P. Michiardi and R. Molva, “CORE: A Collaborative Reputation Mechanism to Enforce Node Cooperation in Mobile Ad Hoc Networks,” *6th IFIP Conf. Sec. Commun. and Multimedia*, 2002.

[11] V. D. Park and M. S. Corson, “A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks,” *INFOCOM*, 1997.

[12] K. Paul and D. Westhoff, “Context Aware Detection of Selfish Nodes in DSR based Ad-hoc Networks,” *Proc. IEEE GLOBECOM*, 2002.

[13] C. E. Perkins and E.M. Royer, “Ad-hoc On-Demand Distance Vector Routing,” *Proc. 2nd IEEE Wksp. Mobile Comp. Sys. and Apps.*, Feb. 1999.

[14] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, “SPINS: Security Protocols for Sensor Networks,” *Wireless Network*, Kluwer, 2002.

[15] A. A. Pirzada and C. McDonald, “Deploying Trust Gateways to Reinforce Dynamic Source Routing,” *INDIN*, 2005.

[16] A. A. Pirzada, C. McDonald, and A.Datta, “Performance Comparison of Trusted-Based Reactive Routing Protocols,” *IEEE Trans. On Mobile Computing*, June 2006.

[17] A. A. Pirzada and C. McDonald, “Establishing Trust In Pure Ad-Hoc Networks,” *ACSC*, 2004.

[18] A. A. Pirzada, A.Datta, and C. McDonald, “Trust-Based Routing For Ad-Hoc Wireless Networks,” *ICON* , 2004.

[19] S. Tanachaiwiwat, P. Dave, R. Bhindwale, and A. Helmy, “Location-centric Isolation of Misbehavior and Trust routing in energy-constrained sensor networks,” *IPCCC*, 2004.

[20] M. Ye, E. Chan, G. Chen, and J. Wu, “Energy Efficient Fractional Coverage Schemes for Low Cost Wireless Sensor Networks,” *ICDCSW*, 2006.