A Swarm Intelligent Secured Routing for Wireless Sensor Network

D.Usha
Assistant Professor/CSE Dept
National Engineering College K.R.Nagar, Kovilpatti
Tamil Nadu, India

B.Paramasivan, PhD.
Professor and Head/CSE Dept
National Engineering College K.R.Nagar, Kovilpatti
Tamil Nadu, India

ABSTRACT

Rapid development of sensor technology, within short span has attracted extensive attention in wide range of applications. But the adhoc nature of the Wireless Sensor Network (WSN) does not support any traditional routing or security scheme and hence the problem of finding a minimum cost, secured and reliable path between the source and destination still becomes a crisis. Therefore, a route maintenance algorithm for WSN should tolerate any intermittent failures and reach destination through shortest and minimal latency path. This thrust has motivated to design the proposed algorithm that provides a probabilistic multi-path routing which constantly updates the goodness of choosing a particular path by measuring the link quality in those paths and also prioritizes the forwarding candidates based on its truthfulness in addition to the shortest path metric. Moreover this proposed system also concentrates on security of a node by assessing its truthfulness. The Prejudice value (propensity to trust/mistrust its neighbors) and deserve value (information of other nodes about this node) are computed to detect the trustworthiness of the node. The effect of packet delivery was analyzed and it is found that perfection is brought about by the participation of the forwarding candidates.

Keywords

Wireless Sensor Networks, Reliability, Swarm intelligence, Secured routing, Minimal Latency, Link Quality, Shortest Path.

1. INTRODUCTION

The rapidly changing and unpredictable nature of ad-hoc Wireless Sensor Networks pose a wide range of challenges like efficient routing, load distribution, congestion avoidance, energy consumption, etc. Though a number of routing protocols exists, which aim to provide effective routing in WSN, very little address the need or would provide a plausible solution for overall network management. Some proactive routing protocols like DSDV and on-demand routing protocol like AODV and DSR, only attempt to solve simply one challenge such as effective routing or minimal end-to-end delay or maximum throughput etc. Hence the primary goal in an ad-hoc wireless network is to efficiently design a routing protocol with some crucial parameters such as congestion avoidance, minimizing energy consumption through load balancing, packet delivery ratio, percentage of optimal routes taken, the average end-to-end delay, secured transmission etc. This thrust has motivated to develop a fault-tolerant routing algorithm which tolerates various types of failures and would confer the overall network management in addition to just packet routing.

Multi-objective problem requires solutions that incorporate optimization techniques like swarm intelligence, where resource constraints are minimized with maximized performance. The natural adaptability of insects and other creatures to survive and thrive through cooperative behavior has astounded the researchers to analyze the swarm behavior of ants, bees, birds etc and attempt to artificially replicate it. These special features make swarm intelligence [12] play important roles in many engineering applications such as formation control of multi-robot system, massive distributed sensing using mobile sensor networks, warfare using cooperative unmanned aerial vehicles etc. The swarm behavior of the ants when implemented in engineering problems is capable of solving many complicated and dynamic optimization issues.

The proposed algorithm provides a probabilistic multi-path routing called swarm intelligent secured routing, where agents move towards an optimal solution by sharing their knowledge with their neighbors. The nodes study the link weights based on link quality and minimum hop count metrics to drive an optimal value for each route. Based on this information, a sender selects the default path and a list of forwarding nodes that are eligible for forwarding the data. Furthermore at each hop, the node that forwards the packet will check its neighbor list to see whether the nodes are in the path to the destination. By performing such identification check in addition to its location information before forwarding, the effect of the path divergence can be very much alleviated. Thus, the route taken by the node is both energy aware and time sensitive at hop level. The computation time is defined as the amount of time the algorithm takes to obtain an optimal solution.

Hijacking of nodes and extraction of cryptographic material is quite easy and gives the attacker the possibility to add malicious nodes or inject bogus data into the network. Instead of hardening the security in WSNs, assessing the trustworthiness of sensor data is an alternative solution. The goal of the trustworthiness assessment is to determine whether the application can rely on the data or not. Trustworthiness allows applications to separate erroneous sensor data from trustworthy ones and it also supports for energy optimization because data transmission is the most energy consuming task in WSN.

Reputation systems have been developed in order to identify compromised nodes, based on their behavior. Reputation is based on a collection of evidence of good and bad behavior undertaken by other entities. In this approach, only well behaving nodes can get access to other nodes’ information. In [16], a framework trust-and-clustering based on public key authentication for mobile ad hoc wireless networks is introduced, where each node monitors and rates each other. The goal is thus to discover and isolate dishonest nodes. But in [15], authors use subjective logic for evaluating direct, observation and recommended opinions on the nodes involved in a MANET (Mobile Ad Hoc Network) based on sensor data origin. Nevertheless, all existing approaches barely address erroneous data processing.

Zhang et al proposed a trust evaluation model, where every node participates locally and independently in evaluating all other nodes. The node's movement and its routing table updates are traced to build an anomaly detection model. Upon detecting a malicious node, the local detector broadcasts the message to the network, and every node makes a final decision based on its neighbors report. Such a global computation for all the other nodes cannot be accomplished in practical resource-constraint sensor networks. The computation is very high.

The proposed algorithm uses a trust prediction acuity, finding the most reliable or trusted node in a local sub graph. The forwarding list of nodes is confirmed for its truthfulness. The source node calculates the Prejudice and Deserve values of the forwarding nodes by using the trust score, which is the edge weight of the network. These values determine the trustworthiness of the nodes. Unlike other methods this algorithm works even when the scores are negative. It converges to a unique value very quickly with limited iterations itself. So calculations are bounded with only little processing. This algorithm is based on the idea that the opinions of trustworthy node weigh more but a highly prejudiced node should weigh less. Hence this algorithm allows the data packets to be broadcasted in swift, reliable and secured way in wireless sensor network.

2. SELECTION OF FORWARDING CANDIDATES

The proposed algorithm detects multiple flows by strategically learning the set of nodes and selecting a list of forwarding candidates at each hop level. Then it chooses the best sequence of nodes from source to destination and forwards each packet through that sequence. The next hop forwarder with the largest positive progress towards the destination which is trustworthy and having less radio loss link is found and listed with its priority. This routing mechanism is implemented with minimum modification to MAC protocol and results in high expected progress per transmission.

2.1 Prioritization of Forwarding Nodes Based on Distance and Link Quality

Geographic routing uses location information to forward data packets in a hop-by-hop fashion [1]. The proposed algorithm directly uses the location information is used to exploit the multiple forwarding candidates' on-the-fly on a per packet basis. The node which is located in the forwarding direction to the destination and having better link quality has the first chance to forward. If the best forwarder does not forward the packet in certain time slots, suboptimal candidate will take turn to forward the packet according to a locally formed order [7].

Initially the sender selects a forwarding list based on its distance. Then this list is limited by the measure of link qualities of the nodes and its trustworthiness (calculated as in chapter IV). The nodes that have more than 50% of packet loss and which is less trusted are removed from the list.

The Link Quality (LQ) is measured [8] by the expected number of transmissions required to reliably send a packet across the link. Each node measures the loss rate of its links to and from its neighbors by broadcasting one probe packet every second and counting the number of probes received in the last 10 seconds. Then, the LQ is calculated by assuming independent packet losses.

$$LQ = \frac{1}{((1 - p_f)x(1 - p_r))} \text{ ----------------------------(1)}$$

P_f - loss probability in forward direction

P_r - loss probability in reverse direction

The Source then broadcasts a small control packet including the list of forwarding nodes. The nodes not in the forwarding list that hear the transmission will discard the packet. Nodes on the forwarding list store the packet and set forwarding timers based on their proximity to the destination. This control packet is used to record the optimal route by forwarding it from source to destination. A node closer to the destination uses a smaller timer and forwards the packet earlier. Also the link quality and the timer value of that path are recorded in that packet and in all the intermediate nodes. Three nodes in the forwarding list are allowed to transmit the packet based on their priority. Upon hearing the transmission for three times, other nodes will remove the corresponding packet from their queues to avoid multiple duplicate transmissions. Due to the broadcast nature of wireless media which does not support long distance transfer, heavy packet loss may occur and this problem is rectified by the LQ measure. Since the forwarding table only depends on local information, it takes much less time to be constructed. The table records only the current active flows and the route expire time decreases with time to rebuild the table for other routes.

3. PATH SAMPLING USING LINK QUALITY

This route maintenance phase combines with reactive forwarder to update routing table information using a proactive route maintenance algorithm. Reactive behavior means that an algorithm only gathers routing information in response to an event, usually for new routes or the failure of an existing route. Proactive behavior means that the algorithm also gathers information at other times, so that routing information is readily available when any event happens. Hence this algorithm is a hybrid routing where an ant-based path sampling and information bootstrapping is executed periodically in each route discovery phase. Bootstrapping is a characteristic of dynamic programming, in which the forwarder also calculates the combined predictable quality of the path based on estimates made by neighboring nodes.

3.1 Calculation of Path Efficiency

Path sampling is done by small control packets that travels from source to destination through the nodes of the forwarding list [2] and collects information about the quality of path. This information is retraced back from destination to source to update the routing tables at intermediate nodes [4]. These values are updated according to the quality of the paths sampled by the ants.

The LQ of the path is calculated by its congestion metric which is initially set as one and broadcasted [3]. All the intermediate nodes multiply its individual congestion measure with the previous Node Congestion (NC) value to obtain the Path Efficiency (PE) value.

Then the PE can be defined as

$$PE = \prod(NC \quad of \quad IN\, from \quad S \quad to \quad D) \text{---------------------------(2)}$$

IN-Intermediate nodes, S-Source, D-Destination

These linear measure of congestion does not show any variation between high values of NC and low values of NC. Therefore the linear measure of the NC is converted as non-linear as

$$\text{New } NC = \left(1-(1-NC)^3\right) \text{--------------------------------------(3)}$$

The higher values of NC have less impact and as NC increases, it has much greater impact. These entries are a measure of the goodness of going over that neighbor on the way to the destination. Less good paths can be occasionally utilized and hence they are maintained as backup in case of failure or sudden congestion.

The PE of a particular path is also based on its hop count (shortest path) with its total NC for that path. The modified equation for PE based on hop count measure is

PE of a Route (using hop count) = $(NC)^L$ --------------------(4)

Where L=Length of the route.

Similarly the PE count is calculated for all the routes. Next Probabilities (P) are assigned to the individual routes based on their PE values.

P(Route) = PE of the Route / Σ (PE count of all routes) ----(5)

Once the probabilities are assigned to the individual routes, the node then chooses a route as per this probability. Hence good routes (based on length and NC) have higher probability of being chosen while all alternate routes are still kept fresh [5]. This algorithm also avoids the delayed delivery of packets due to reconstruction of routes whenever there is any transmission interruption in the selected path.

4. IDENTIFICATION OF TRUSTFULL NODES

Many sensor network systems developed so far have lacked security during their initial design phase, paving way for intruder actions and security breaches that reduced system and application performance. A single attack could endanger partial or full coverage in any network [6]. Hence, an in-depth knowledge of the network helps in designing appropriate and efficient security measures.

The proposed system uses a trust based network [9] where, Watchdog and Pathrater method is initially used to calculate the trust scores of individual nodes. In this method, every node ensures that the neighbor node in the path also forwards the correct packet within the time slot and increases or decreases its trustworthiness accordingly [11]. Based on this the Prejudice and Deserve values of individual nodes are calculated. The prejudice of a node denotes its propensity to trust/mistrust its neighbors and if the node trusts all its neighbors, its recommendation of another node as trustworthy should weigh less. The deserve value is the information given by other nodes about that node depended on the trust scores.

4.1 To form a Trusted Link using Cognitive Intelligence

Let $d^o(i)$ denotes the set of all outgoing links from node *i* and likewise, $d^i(i)$ denotes the set of all incoming links to node *i*. The attributes prejudice or Trustworthiness (expected weight of an out-link) and Deserve (expected weight of an in-link from the prejudice nodes) are measured for each node. Thus, the propensity or prejudice of a node to trust/mistrust other nodes can be measured by the difference between the rating a node provides to another node (i.e., the edge weight) and the “ground” truth, i.e., what the second node truly deserves (this takes into account the trust by other nodes). The prejudice of a node *i* is given by

$$\Pr ejudice(i) = \frac{1}{2\,|\,d^o(i)\,|}\left(\sum_{j \in d^o} w_{ij} - deserve(j)\right) \text{---------------(6)}$$

Normalization is done to maintain the value of prejudice in the range of [−1, 1]. A node is assumed to be truthful if it has a prejudice value of 0. A node has a positive prejudice if it has a propensity to give positive out links, and a negative prejudice otherwise. A node giving a positive rating to other nodes that do not deserve such ratings values would gain a high prejudice value. Using prejudice, the inclination of a node toward trusting/mistrusting is measured. It can also be used to understand the true nature of a node. If a highly prejudice node (either positive or negative) gives a rating, then such score should be given less importance by reducing the effect of prejudice from each out link a node gives.

Similarly, negative weights from a negatively prejudice node are reduced. However, if a node has an edge whose weight has an opposite sign of that of the prejudice, then the value is not changed. For example a positive (negative) prejudice has an edge with negative (positive) weight, and then the value of edge weight is not changed. An auxiliary variable X_{kj} is used to measure the effect of prejudice of node *k* on its outgoing edge to node *j* per unit edge-weight.

$$X_{kj} = \begin{cases} 0 & if\left(\Pr ejudice(k) \times w_{kj}\right) \le 0 \\ \Pr ejudice(k) & otherwise \end{cases} \text{-----------(7)}$$

From the above expression, it is observed that when prejudice and edge weight are of opposite signs, X_{kj} becomes zero and there is no effect of the prejudice. Otherwise, X_{kj} becomes the absolute value of the prejudice. There is an equivalent formulation of X_{kj}:

$$X_{kj} = \max\left(0, \Pr ejudice(k) \times sign(w_{kj}) \le 0\right) \text{----------------(8)}$$

Thus the edge weight is reduced using the effect of prejudice, i.e., X_{kj}. The new weight w_{kj} is scaled from the old weight as follows:

$$w'_{kj} = w_{kj}\left(1 - X_{kj}\right) \text{-------------------------------------(9)}$$

If edge-weight and prejudice are of opposite signs, the new weight remains the same, otherwise it is reduced. The deserve value of a node represents the true trust a node deserves. The prejudice values are used to define the deserve value of a node. Deserve is the expected weight of an incoming link from an unprejudiced node. The deserve value depends on the quality of the in-links and not on the quantity. The deserve of a node *j* is given by

$$Deserve(j) = \frac{1}{|\,d^o(i)\,|}\left(\sum\nolimits_{k \in d^i(j)} \left(W_{kj}\left(1 - X_{kj}\right)\right)\right) \text{--------------(10)}$$

The deserve value lies in the range [−1, 1]. Ranking is based on the principle that a node receiving positive ratings from unprejudiced or negatively prejudice nodes can be trusted more than the node receiving positive links from prejudice (positive) nodes. Thus, the deserve value of a node can be directly used for ranking.

4.2 Computing Prejudice and Deserve

Genuine of a node depends on deserve of its neighbors which in turn depends on the prejudice of their neighbors and so on. The method called fixed-point iteration is used. The prejudice and deserve of node *i* at iteration *t* are denoted by prejudice(*i*) and *deserve*^{t}(*i*) respectively. The values obtained from iteration *t* used to compute the values for iteration *t*+1. Then, using those values, the prejudice values are re-estimated. Thus, deserve^{t+1}(*i*) depends on prejudice(*) (actually, X^t(*)), which in turn is computed using *deserve*^{t}(*).The system converges to a unique solution irrespective of the initial values.

$$Prejudice^{t+1}(i) = \frac{1}{2\,|\,d^{o}(i)|}\left(\sum\nolimits_{j\in d^{o}(i)} w_{ij} - deserve^{t+1}(j)\right) \text{--------(11)}$$

$$Deserve^{t+1}(j) = \frac{1}{|\,d^{o}(i)|}\left(\sum_{k\in d^{i}(j)} \left(w_{kj}\left(1 - X^{t}_{kj}\right)\right)\right) \text{-------------------(12)}$$

C. RANKING

Ranking a node in a graph is an important problem and has attracted a wide attention. It is based on the principle that a node receiving positive ratings from prejudice or negatively [10] prejudice nodes can be trusted more than the node receiving positive links from prejudice (positive) nodes. Thus, the deserve value of a node can be directly used for ranking.

5. EXPERIMENTAL RESULTS

The proposed system has been simulated by using the simulation tool NS2(Network Simulator). A network with 14 nodes is created and node 0 is decided to be the source node.

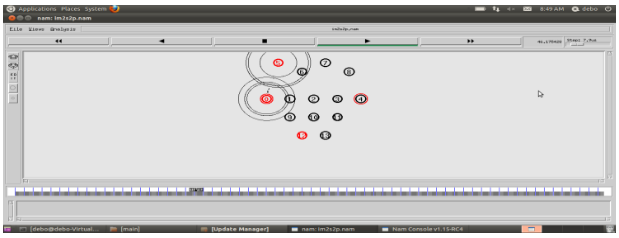

Fig 5.1: Scrutiny of nodes 5 and 12 for truthfulness

In the above figure, the source node checks the nodes 5 and 12 for truthfulness, whether to trust the nodes 5 and 12 or not.

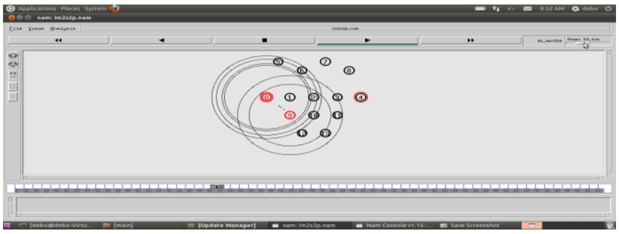

Fig 5.2: Inspection of node 9 and routing.

In the figure 4.2, the source node 0 checks the node 9 for truthfulness at the same time it sends acknowledgement to the same node 9.

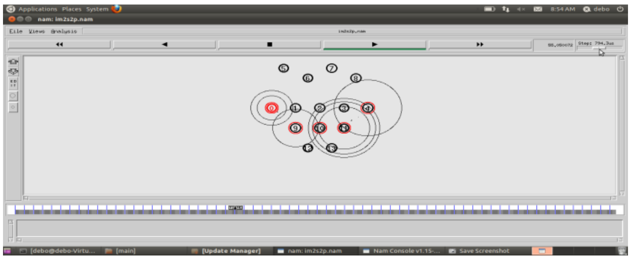

Fig5.3: Sending packets through shortest path.

In figure 4.3, the source node now found that the shortest route from source to destination(base station) is 0-9-10-11-4 and the node 0 now start sending packets to node 9.

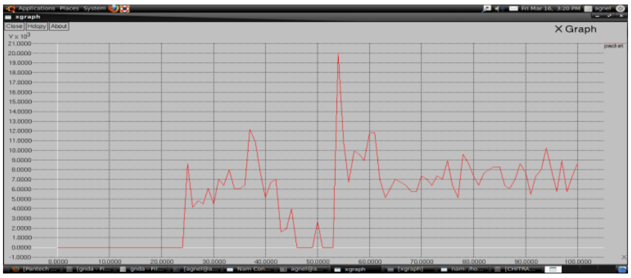

Fig 5.4: Packet flow Analysis

In the above graph X axis represents number of data packets and Y axis represents time period. Initially the system is idle, so there is no fluctuations in the graph. Then it selects a random path to send the packets, while the source node is sending the packets through the random path the time needed to send a packet is abruptly changing, so there is abrupt fluctuations in the graph. After the source node finds the shortest path there is no abrupt changes in the graph.

6. CONCLUSION

In this paper, an alternate approach inspired by concepts of emergence and self-organization in biological systems, has been discussed and implemented. This approach tries to optimize routing by finding the best shortest path, optimize congestion by incorporating Link Quality metrics in route discovery and maintenance and also optimize energy and load balancing by using the probabilistic route choosing algorithm.

This algorithm also uses the Cognitive Intelligence to form trust based link by computing the prejudice and prestige of the nodes in the networks where the edge weight denotes the trust score. Unlike other methods this method works even when the weights are not necessarily positive. These values converge fast to unique values and the errors at any iteration are bounded. This makes the system to Rank the trustworthiness of the nodes with limited processing effort.

7. FUTURE WORK

In future, the routing can be enhanced with signal strength incorporated as an additional route metric, so as to predict the link breaks before they actually occur and to redirect to other routes. Also other forms of attacks can be tackled such as adversarial nodes, colluding groups etc.

8. REFERENCES

[1] Karp and H.T. Kung, “GPSR: Greedy Perimeter Stateless Routing for Wireless Networks,” Proc. ACM MobiCom, pp. 243- 254, 2000.

[2] Mesut Gunes, Udo Sorges, Imed Bouazizi, “ARA – The Ant-colony based routing algorithm for MANETs”, International workshop on Ad-hoc Networking (IWAHN 2002)

[3] Deb, S. Bhatnagar, and B. Nath, “ReInForM: Reliable Information Forwarding Using Multiple Paths in Sensor Networks,” Proc. Ann. IEEE Int’l Conf. Local Computer Networks (LCN ’03), pp. 406- 415, 2003.

[4] Frederick Ducatelle, Gianni Di Caro and Luca Maria Gambardella, “Using ANT Agents to combine reactive and proactive strategies for routing in mobile ad-hoc networks”,2004

[5] Felemban, C.-G. Lee, E. Ekici, R. Boder, and S. Vural, “Probabilistic QoS Guarantee in Reliability and Timeliness Domains in Wireless Sensor Networks,” Proc. IEEE INFOCOM, pp. 2646-2657, 2005.

[6] A Trust-Based Geographical Routing Scheme in Sensor Networks Ka-Shun Hung; King-Shan Lui; Yu-Kwong Kwok Wireless Communications and Networking Conference, 2007. Page(s): 3123 – 3127

[7] IEEE Conference PublicationsChen, J. Deng, and P. Varshney, “Selection of a Forwarding Area for Contention-Based Geographic Forwarding in Wireless Multi-Hop Networks,” IEEE Trans. Vehicular Technology, vol. 56, no. 5, pp. 3111-3122, Sept. 2007.

[8] Rozner, J. Seshadri, Y. Mehta, and L. Qiu, “SOAR: Simple Opportunistic Adaptive Routing Protocol for Wireless Mesh Networks,” IEEE Trans. Mobile Computing, vol. 8, no. 12, pp. 1622-1635, Dec. 2009.

[9] *Trustworthiness* Assessment of Wireless *Sensor* Data for Business Applications Gomez, L.; Laube, A.; Sorniotti, A. Advanced Information Networking and Applications, 2009. AINA '09. International Conference

[10] J. Leskovec, D. P. Huttenlocher, and J. M. Kleinberg. Predicting positive and negative links in online social networks. In WWW, pages 641–650, 2010.

[11] Implementing a Trust-Aware Routing Protocol in Wireless Sensor Nodes Zahariadis, T.; Trakadas, P.; Leligou, H.; Karkazis, P.; Voliotis, S. Developments in E-systems Engineering (DESE), 2010 Publication Year: 2010 , Page(s): 47 - 52 IEEE Conference Publications

[12] Yan-fei Zhu,Xiong-min Tang.”Overview of Swarm Intelligence”,In ICCASM(IEEE),Pages 400-402,2010.