Wireless Sensor Network, **2011, 3, 125-134**
doi:10.4236/wsn.2011.34015 Published Online April 2011 (http://www.SciRP.org/journal/wsn)

A Direct Trust Dependent Link State Routing Protocol Using Route Trusts for WSNs (DTLSRP)

Shaik Sahil Babu, Arnab Raha, Mrinal Kanti Naskar
Department of Electronics and Telecommunication Engineering, *Jadavpur University*, *Kolkata*, *India*
E-mail: {*sksahilbabu*419, *arnabraha*1989}*@gmail.com*, *mrinalnaskar@yahoo.co.in*
Received February 19, 2011; *revised March* 3, 2011; *accepted March* 23, 2011

Abstract

The traditional cryptographic security techniques are not sufficient for secure routing of message from source to destination in Wireless Sensor Networks (WSNs), because it requires sophisticated software, hardware, large memory, high processing speed and communication bandwidth. It is not economic and feasible because, depending on the application, WSN nodes are high-volume in number (hence, limited resources at each node), deployment area may be hazardous, unattended and/or hostile and sometimes dangerous. As WSNs are characterized by severely constrained resources and requirement to operate in an ad-hoc manner, security functionality implementation to protect nodes from adversary forces and secure routing of message from source node to base station has become a challenging task. In this paper, we present a direct trust dependent link state routing using route trusts which protects WSNs against routing attacks by eliminating the untrusted nodes before making routes and finding best trustworthy route among them. We compare our work with the most prevalent routing protocols and show its benefits over them.

Keywords: Wireless Sensor Network (WSN), Geometric Mean (GM), Direct Trust, Indirect Trust, Route Trust (RT), Base Station (BS), Benevolent Node, Malicious Node, Homogeneous Nodes, Packet Latency, Packet Transmission Rate, MATLAB

1. Introduction

Trust establishment among nodes is a must to evaluate the trustworthiness of other nodes and is one of the most critical issues in WSNs. Trust is dependent on time; it can increase or decrease with time based on the available evidence through direct interactions with the node or recommendations from other trusted nodes. Trust-modeling is mathematical representation of node's opinion of another node in a network. We need mathematical tools to represent trust and reputation, update these continuously. Maintaining a record of the transactions with other nodes, directly as well as indirectly, from this record a 'trust' value will be established [1].

Security and trust are two tightly interdependent concepts. Generally these terms are used interchangeably when defining a security system [2]. However, security is different from trust. Security is more complex and the overhead is high. In other words, security means no one is trusted and requires authentication all the time which leads to high overhead, *i.e.* encryption and decryption with secret key [3]. Trust means everybody is trusted somehow and does not require any authentication (less overhead). It tells the degree of reliability. Every node finds the trust of all other nodes, based on previous experience and recommendations in fulfilling its promises.

Trust management system for wireless sensor networks (WSNs) is a mechanism that can be used to support the decision-making processes of the network [4]. It aids the members of WSN (trustors) to deal with uncertainty about the future actions of other participants (trustees). As WSNs are highly application oriented, these various applications bring various security needs. In WSN, sensor nodes have limited communication bandwidth, processing resources, memory space and battery capacity [5]. Hence, the trust management system should enable the WSN to be secure while significantly reducing computing and communication overheads. The WSNs can be established without any existing infrastructure, which is a major feature exploited in most applications, they rely on the mutual cooperation among nodes to route traffic towards sink or base station. Hence, trust establishment among the nodes is a must to evaluate the trustworthiness of other nodes and is one of the most critical issues in

WSNs. Survival of a WSN is dependent upon the cooperative and trusting nature of its nodes. Hence, the trust establishment between nodes is must. Trust is dependent on time; it can increase or decrease with time based on the available evidence through direct interactions with the same entity or recommendations from other trusted entities [2].

Trust aware routing framework for WSNs is proposed by [6], to secure multi-hop routing in WSNs against intruders exploiting the replay of routing information. With the idea of trust management, their proposal enables a node to keep track of the trustworthiness of its neighbours and thus to select a reliable route. Their proposal can also be implemented for large-scale WSNs deployed in wild environments. Many security attacks have been presented in ([7,8]) with a significant subset targeting the routing process [9]. If an adversary force manages to capture the node, it participates in the network, and it can damage the routing process by simply dropping the packets it receives for forwarding. Another attack easy to implement is packet modification. In [10] an approach that the human society follows proposed to defend against the majority of routing attacks. Although the design of mechanisms to enhance security at all layers of the networking protocol stack has attracted the interest of the research community (e.g. [11,12]), very limited implementation effort has been reported. In [13], the implementation of a link-layer security architecture is presented, while in [14] experience regarding the implementation of hash-based encryption schemes in tinyos operated sensor nodes is reported. In [15], the efficiency of a set of routing protocols is compared based on real testbed experiments. In [16], very limited information regarding the implementation of a trust model is provided. Finally, in [17] presented results and experience gained through the implementation of a location-based trust-aware routing solution. A distributed trust model is incorporated in the routing solution which relies on both direct and indirect trust information.

In this paper, we present simulated results of a new link state routing protocol based on trust by eliminating the malicious nodes from the network (lsrp based on trust). This protocol incorporates a trust computational model with direct and indirect experiences based on traditional weighting approach of the qos characteristics such as packet forward, data rate, power consumption reliability, etc. using beta probability distribution [2]. The trust management system at the node computes trust table for network nodes, and then using a threshold find out the benevolent nodes of the network. Then, using link state routing it finds all available paths by eliminating the malicious nodes.

The rest of this paper is organized as follows: first in section 2 we present the related work on WSN routing protocols based on trust and in section 3 the designed lsr protocol based on direct trust while in section 4 performance evaluation. In section 5 the simulation results and in section 6 conclusions.

2. Related Work

Trust Based Routing methods: Enhancements in the routing related protocols based on the trust have been widely addressed in the literature. The following are the most important research results in this direction:

2.1. ARIADNE

It is a very efficient protocol, using highly efficient symmetric cryptographic primitives and per-hop hashing function [18]. It prevents the attackers or compromised nodes from tampering with uncompromised routes consisting of uncompromised nodes, and also prevents a large number of types of Denial-of-Service attacks.

2.2. ATSR (Ambient Trust Sensor Routing)

A fully distributed Trust Management System is realized in ATSR [5] in order to evaluate the reliability of the nodes. Using this approach, nodes monitor the behavior of their neighbors in respect to different trust metrics and finds direct trust value per neighbor. It also, takes into account indirect trust information, *i.e.* trust information from its neighbors, also called reputation. Direct and indirect trust information is combined to reach the Total Trust information. Finally, the routing decisions are based on geographical information (distance to the base-station) and Total Trust information. The trust model presented has been integrated with a location-based routing protocol. If no malicious node exists in the network, *i.e.* the Total Trust is almost equal to 1, the ATSR behaves simply the Greedy Perimeter Stateless Routing (GPSR) protocol.

2.3. Trusted AODV

It is an extended AODV routing protocol to perform routing by taking trust metrics into account [19]. First, a trust recommendation mechanism introduced and then the routing decision rules of AODV are modified to take trust into account. A set of policies is derived for a node to update its opinions towards others, because it is necessary to design a trust information exchange mechanism when applying the trust models into network applications.

2.4. Trusted GPSR

The Greedy Perimeter Stateless Routing [20] is modified

to take trust levels of node into account. Each time a node sends out a packet it waits until it overhears its neighboring node forwarding it. Based on this correct and prompt forwarding information it maintains a trust value for its neighbors. This information is then taken into account in the routing decisions.

2.5. SPINS (A Suite of Security Protocols Optimized for Sensor Networks)

This [21] has been designed to provide data authentication, data confidentiality and evidence of data freshness. In this protocol two security blocks SNEP and µTESLA are involved. The first block introduces overhead of 8 bytes and maintains a counter fro achieving semantic security. µTESLA provides authentication for data broadcasting. Though SPINS claim to provide trusted routing ensuring data authentication and confidentiality, but it does not deal with Denial of Service Attacks.

2.6. Trust- Aware DSR

The watchdog and Pathrater modules has been designed and incorporated in the Dynamic Source Routing protocol for security [22]. The watchdog module is responsible for detecting selfish nodes that do not forward packets. For this, each node in the network buffers every transmitted packet for a limited period. During this period each node enters into promiscuous mode in order to overhear whether the next node has forwarded the packet or not. And based on the feedback that Pathrater receives from the watchdog, it assigns different ratings to the nodes. These ratings are then used to select routes consisting of nodes with the highest forwarding rate.

2.7. CONFIDANT (Cooperation Of Nodes, Fairness In Dynamic Ad-hoc Networks)

This [23] protocol adds reputation system and a trust manager to the Watchdog and Pathrater scheme. The trust manager evaluates the events reported by the Watchdog and issues signals to other nodes regarding malicious nodes. The signal recipients are maintained in a friends-list. The reputation system maintains a black-list of nodes at each node and shares them with friends-list nodes. In one way it is a punishment based scheme by not forwarding packets of nodes whose trust level drops below the certain threshold.

2.8. TRANS (Trust Routing for Location Aware Sensor Networks)

TRANS [24] routing protocol selects routes based on trust information not on hop count to avoid the insecure locations. This protocol assumes that the sensors know their locations and that geographic routing is used. A sink sends a message only to its trusted neighbors for the destined location. Those corresponding neighbors forward the packet to their trusted neighbors that have the nearest location to destination. Thus the packet reaches the destination along a path of trusted sensors. Here the important feature of TRANS, the sink identifies misbehavior by observing replies, probes potential misbehaving locations and isolates insecure locations. On discovery of such locations, the sink records and advertises to the neighboring nodes.

2.9. Traditional Weighting Approach Using Beta Probability [1]

Momani introduced one algorithm for trust calculation and risk assessment based on trust factors and dynamic aspects of trust. As shown in **Figure 1**, he assumed that trust is computed using traditional weighting approach of the QoS characteristics such as packet forward, data rate, error rate, power consumption, reliability, competence, etc.

A is direct trust (*experience*), B is indirect trust (*recommendations*), C is total trust.

T is required trust, R is risk, Total trust $C = F(A, B)$.

$T_Y(X)$ means trust at Y on X *i.e.* (trust that Y is having on X).

$T_{Y_i}(X)$ means trust at Y on X for ith category.

$A = \sum_{i=1,2,\cdots,n} T_{Y_i}(X)$. Sum of trust values at Y on X for n different events.

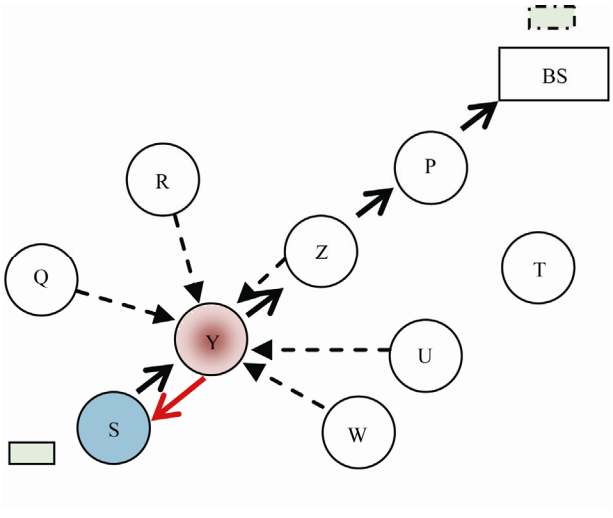

Figure 1. General trust computational model in brief (Traditional weighting approach using Beta probability distribution).

$B = \sum_{Y=1,2,\cdots,m} T_Y(X)/m$. Average of sum of indirect trust values at Y on m nodes. Trust reported from all the surrounding nodes that have had previous experience with the node.

Total trust $C = F(A,B)$ and it can be

$$\begin{aligned} C &= A*W_A + B*W_B \\ &= \sum_{I=1,2,\cdots,n} T_{Y_i}(X)*W_A \\ &\quad + \sum_{Y=1,2,\cdots,m} T_Y(X)*W_B/m \end{aligned}$$

The weights W_A and W_B can be assigned using different approaches.

CASE 1: Some nodes may be given more weightage in direct trust; others may be given more weight in recent indirect trust, *i.e* $W_A > W_B$ or $W_A < W_B$.

CASE 2: Weights to the direct trusts of some events may be given more importance, and others are less importance. Similarly, for indirect trusts nearby nodes may be given more importance and others is less importance as shown in figure.

$$\begin{aligned} C = &\left[T_{Y_1}(X)*W_1 + T_{Y_2}(X)*W_2 \right. \\ &\left. + T_{Y_3}(X)*W_3 + T_{Y_4}(X)*W_4 + \cdots\right] \\ &+\left[T_1(X)*W_a + T_2(X)*W_a + T_3(X)*W_b \right. \\ &\left. + T_4(X)*W_b + \cdots\right] \end{aligned}$$

Risk can be calculated as the difference between trusts required and total trust.

$$R = T - C, \text{ and } C = A*W_A + B*W_B$$

A traditional weighing approach to calculate Trust and asses Risk (Risk assessment algorithm) is introduced. These weights W_A, W_B can be assigned using different approaches. Some nodes might give more weight to direct trust, others might give more weight to recent indirect trust.

3. A Direct Trust Dependent Link State Routing Protocol Using Route Trusts for WSNs (DTLSRP)

Our model uses the Momani's model of assigning discrete trust values to the sensor nodes. Most of the work present in the literature establishes different techniques to calculate trust of individual nodes with respect to its one hop neighbor (direct trust) or with respect to the one-hop neighbors of the one-hop neighbor used initially to measure the direct trust (known as the indirect trust). We present this concept in brief.

Let us first present a network topography depicting the location of nodes and the path through which the packet must be transmitted from one node(which may be called the source) to the Base Station or another node(which may be called the destination) as is shown in **Figure 2**. This diagram represents a network of 25 nodes deployed in the form of a square area. Now suppose the source wants to send a packet to the destination. Let us for our ease of understanding we first assign some trust values to the nodes color marked in this diagram. We represent the same in the form of a table. In this paper we are only interested the transmission and reception of data packets and hence our concern will be only in the direct communication trusts measured particularly on the basis of first-hand experience of percentage of successful reception of packets. Suppose node *A* wants to send packets of data to *B*. The trust map is shown in **Figure 2**. Now in this case, we are only interested in the direct trust parameters which node B is having on node A such as percentage of successful reception of data packets, latency of transmission, relative power levels etc. First of all here $W_B = 0$ as we are not concerned with indirect trusts. So $W_A = 1$ and the calculation of $T_B(A)$ which is the trust value that *B* is having on *A* will only be based upon the direct trust parameters which is also known as the one-hop trust metric.

The non-inclusion of indirect trusts is obvious from the fact that as we are only interested in communication trust that is transmission of data through the network, such emphasis on indirect trust is redundant more so because the paths are determined beforehand and a receiving node is only concerned whether the data it is receiving from the sender is at all trustworthy or not irrespective of what other nodes think about it. It also decreases the memory and processing capability of the nodes.

In the following table we represent some practically calculated trust values assigned to the 10 nodes as shown in the network topology described earlier. It is to be noted in this case that for this table the direct trust value assigned to a single node is with respect to its direct one hop neighbor in the routing path. If a node belongs to more than one routing path then individual trust values must be taken in to account. For example, in case of node 6 if two separate paths exist and its 2 next hop neighbors are respectively 7 and 10 then $T_7(6)$ and $T_{10}(6)$ both are to be calculated for calculation of route trusts of the two paths. We divide our routing protocol into several steps as shown in the following **Figure 3**.

Step*1: *Calculation of direct trust of the indivudual nodes

As mentioned previously we are only concerned with the direct trusts in this case. So if our topology consists

Figure 2. Simple trust map.

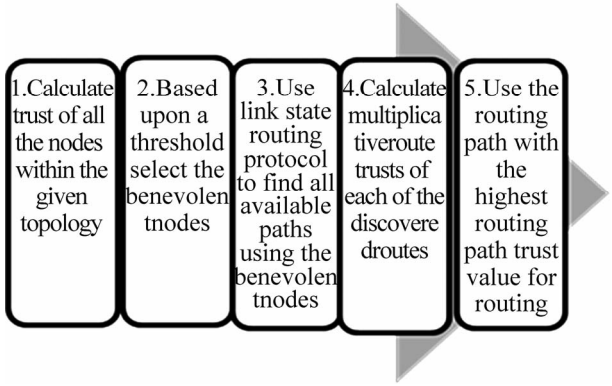

Figure 3. Proposed protocol structure.

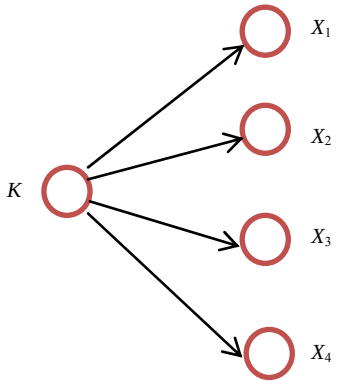

Figure 4. Trust relationship with individual nodes.

of N nodes then all of $T_{mX_i}(K)$ *i.e.* the trusts of one-hop neighbors (X_i) to the node K, are calculated where $i = 1$ to N, and m is the QoS parameters for different metrics. This is shown in **Figure 4**. The parameters used for such trust calculation are a) $T_{1X_i}(K)$ probability of successful reception of packets of node X_i from node K (here $m = 1$); b) $T_{2X_i}(K)$ ratio of minimum latency possible and mean latency of packets sent from i to X_i (here $m = 2$); c) $T_{3X_i}(K)$ ratio of the power level (or battery life) of node K and the maximum power level (or battery life) possible for that WSN node (here $m = 3$), etc. More number of direct trust parameters can be added such as $T_{mX_i}(K)$ etc.

We calculate overall direct trust as

$$T_{X_i}(K) = \left[T_{1X_i}(K) * T_{2X_i}(K) * T_{3X_i}(K) \cdots * T_{mX_i}(K)\right]^{(1/m)}$$

which is the Geometric Mean(GM) of all the parameters. For m different metrics, Trust of node K at X_1, X_2 and X_3 may be

$$T_{X_1}(K) = \left[\pi\left(T_{1X_1}(K), T_{2X_1}(K), T_{3X_1}(K), \cdots, T_{mX_1}(K)\right)\right]^{(1/m)}$$

$$T_{X_2}(K) = \left[\pi\left(T_{1X_2}(K), T_{2X_2}(K), T_{3X_2}(K), \cdots, T_{mX_2}(K)\right)\right]^{(1/m)}$$

$$T_{X_3}(K) = \left[\pi\left(T_{1X_3}(K), T_{2X_3}(K), T_{3X_3}(K), \cdots, T_{mX_3}(K)\right)\right]^{(1/m)}$$

In our case we assume $W_A = 1$ and $W_B = 0$. So $T_{X_i}(K)$ is the set of trust values assigned to the node K by one-hop neighbors (X_i) .

This is different from the Mohammad Momani's model as he has calculated the trust as the arithmetic sum or mean of the different parametric probabilities which can lead to some serious false values. Suppose at least one parameter (say % successful packet transmission or packet latency) gives a trust value of 0 but others have high values. So a high value of trust may be assigned even though packet transmission is 0 or latency is infinite. This can be avoided if we calculate the product or the geometric mean of the trusts as suggested in this model. This is also proved by an example later.

It is to be noted that since trust is a probabilistic value its range must always remain within 0 to 1 with a higher value of indicating higher trust in the positive sense. Hence it is to be noted that while calculating the trust if there is some parameter which is better if less, then it has to be multiplied after subtracting it from 1.

Step 2: Calculation of threshold (t_{th}) direct trust value and subsequent selection of benevolent nodes

The value of T_{TH} is application-dependent and needs to be determined according the accuracy, precision, reliability, risk acceptable to be operated in the present network. It should be calculated upon the consumer or application needs. In course of this paper we assume the $T_{TH} = 0.5$ to be the balanced and appropriate value. Hence all the subsequent formulations, deductions and derivations involve T_{TH} to be equal to 0.5. Now if $T_{X_i}(K) \geq T_{TH}$ then node K is trusted and it is assigned to be a benevolent node with respect to the X_i^{th} node.

Else if $T_{X_i}(K) < T_{TH}$ then node K is not trustworthy and it is assigned to be a malicious node with respect to the X_i^{th} node. This process is repeated for $i = 1$ to N.

Step 3: Application of link state routing protocols (lsrp) for finding all available paths using the benevolent nodes only

Link state routing protocols are the most widely used static routing protocols. We are only interested in the basic features of the LSRP and are not mentioning the wide details of it or whether OPSF, IS-IS, MOPSF, MLSRP etc. are used in this case. Applying anyone of these LSRPs are possible depending upon other network needs.

The basic features of LSRP in brief are:

a) Discovery of the neighbors of the nodes and learning their network addresses.

b) Measurement of the delay or cost to each of its neighbors.

c) Construction of a packet telling all the information

learnt by it.

d) Transmission of this packet to all the router nodes.

One of the main advantages in our algorithm is that it doesn't require the LSRP to apply Dijsktra's algorithm or any other algorithm to find the shortest path from the source to the sink. It gets automatically evaluated from determination of Route Trust Values.

The above 4 steps help in completion of the adjacency database in which each node stores all the information regarding its neighbours. The only addition for our trust based routing protocol is that an extra field will be added to this database in our case that is of the Trust values. Data packets or acknowledgement packets may be modified for inclusion of this field. For example, according to the given topology if the DESTINATION node has 3 one hop neighbours 3, 5 and 9 then it should have a table embedded in its memory as shown in **Table 1**. Now such a table will be formed only in those nodes and only those nodes will be used for LSRP (*i.e.* included in the routing table) which are determined to be benevolent nodes with the method given in the previous step. Hence valid routing paths will only be decided by LSRP involving only the benevolent nodes eliminating the malicious nodes.

Step* 4: *Evaluation of multiplicative route trusts (rt)

Suppose that upon the implementation of the LSRP protocol the paths that are found out are shown by bold continuous arrows in the following **Figure 5**.

Table 1. Different Trusts in node's memory.

N_ID	Trust Value
3	0.9
5	0.75
9	0.7

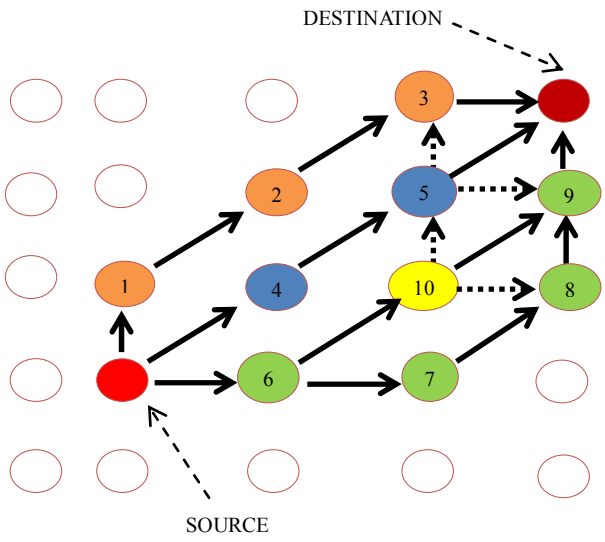

Figure 5. Evaluation of multiple routes in WSN.

Assuming the all the colored nodes have trust valuesabove the threshold value the paths we get are the following:

Route 1 = *S*->1->2->3->*D*,

Route 2 = *S*->6->7->8->9->*D*,

Route 3 = *S*->4->5->*D*,

Route 4 = *S*->6->10->9->*D*, with the source(*S*) and the destination (*D*) be included implicitly.

Now the Route Trusts are calculated as follows:

Trust for Route 1 *i.e.* $RT_1 = \pi\,(T_1(S), T_2(1), T_3(2), T_D(3))$

Trust for Route 2 *i.e.* $RT_2 = \pi\,(T_6(S), T_7(6), T_8(7), T_D(9))$

Trust for Route 3 *i.e.* $RT_3 = \pi\,(T_4(S), T_5(4), T_D(5))$

Trust for Route 4 *i.e.* $RT_4 = \pi\,(T_6(S), T_{10}(6), T_9(10), T_D(9))$

So the Route Trusts (*RTs*) are calculated by multiplying the direct trusts of all the nodes belonging to the path with each other. Such a method provides plenty of advantages.

Step*5: *Selection of the most appropriate path and subsequent transmission of data through it

In the fifth and final step, data will be routed only through that path whose *RT* value is the highest.

So if in the previous case $RT_3 > RT_1 > RT_4 > RT_2$, then data under normal conditions will be transmitted through the 3*rd* route or through *S*->4->5->*D*. The priority order will be same as the decreasing values of *RT*. In case $RT_L = RT_M$, where *L* and *M* are different routes. Then both the routes will be given the same priority and transmission can take place through any of them or may be through some prior tie-breaking rule as per network needs like less number of hops route.

This method provides many advantages as compared to the existing ones: Firstly, it allows us to find the shortest path without applying Dijsktra's algorithm. It also allows us to find out the correct path even when the T_{TH} value cannot be decided or evaluated and hence the separation of benevolent and malicious nodes is not possible. Even if it is done this method provides more accu- rate and precise choice of the suitable path. Examples of these cases are given next:

In the **Table 2** given below we represent 4 distinct, random but important cases which will show the benefit of using our protocol for routing purposes compared to the conventions followed presently although no clear process of routing using trust values exist in the available literature.

In accordance with the network topology given, we assign the following unique trust values to each node. For simplicity, if multiple instances of a single node exist in each routing path then instead of assigning 2 different trust values we have given a single value, e.g., $T_7(6) = T_{10}(6)$.

Table 2. Trust levels for different cases.

Node trust	Trust level			
	Case I	Case II	Case III	Case IV
T(1)	0.3	0.8	0.85	0.75
T(2)	0.85	0.5	0.4	0.75
T(3)	0.9	0.6	0.3	0.75
T(4)	0.6	0.75	0.75	0.75
T(5)	0.75	0.2	0.5	0.75
T(6)	0.9	0.4	0.55	0.75
T(7)	0.35	0.9	0.6	0.75
T(8)	0.5	0.9	0.4	0.75
T(9)	0	0.7	0.2	0.75
T(10)	0.30	0.9	0.9	0.75

These cases show one of the major advantages of our protocol over the existing ones. In case of the existing protocols there is no existence of routing path trusts. They suffer in different aspects:

Calculation of direct trust of a single node with respect to another node based on number of parameters is accomplished by taking the average of the individual single parametric trusts as shown in Momani's model ([1,2]). So if say the trust due to successful transmission of packets is 0 and the rest have a high value of it, there will not be any successful transmission of packet although the overall trust of the node will be quite high which denotes definitely a false value.

Another deficiency of the protocols is that as the routing is based upon the trust values of the nodes based only upon the one-hop neighbours they remain largely unaware of the rest of the network topology. Decisions on routing being taken largely and only upon the one-hop neighbours the probability of choosing the best path is very low.

If all the nodes are homogeneous and they have the same levels of trust probabilities assigned then the selection of routes may be random and the routing may not take place through the shortest path. It will be evident from the following cases that our algorithm does not suffer from these 3 difficulties but instead provide solutions of these. We take each case and give the different choices of routing paths with respect to the existing method and our mode as follows:

	Case 1	Case 2	Case 3	Case 4
Other methods	Route 2	Route 1	Route 1	Any Route
Our method	Route 3	Route 4	Route 3	Route 3

4. Explanations

CASE 1: The existing method will choose Route 2 which is evident as $T(6) > T(4) > T(1)$ and $T(7) > T(10)$. But this decision is totally wrong since all the packets will be entirely dropped at node 9. But our method will choose Route 3 which not only provides the highest reliability but also the shortest path. In the first step as shown below, these nodes are eliminated based upon $T_{TH} = 0.5$ and with it the paths and then the RTs are calculated. This is shown in the following **Figure 6**.

Even in case the threshold value is indeterminate, $RT_2 = RT_1 = 0$. Hence they will not be chosen.

CASE 2: When similar type of process is followed Momani method chooses Route 1 and our method Route 4 in case of separation of benevolent and malicious nodes take place on the basis of $T_{TH} = 0.5$. This method signifies the importance of the threshold trust level in our protocol because in its absence the path chosen would have been R4. It would have been an incorrect path as in this path the first node itself is behaving maliciously although the RT_4 value is the highest. But due to prior elimination of the node 6 as a malicious node we can overcome the problem. Setting the T_{TH} value to suit the network's QoS needs appropriate path can be selected.

CASE 3: In this case upon application of the first and second steps of our algorithm, only one path remains valid so calculation overhead is decreased.

CASE 4: This case is very interesting. When the trust levels of all the nodes are equivalent existing protocols will be choosing randomly one of the valid paths. So if there exists K number of paths the probability that the most appropriate is chosen is only 1/*K*. However if we apply our model, it will always choose the shortest path

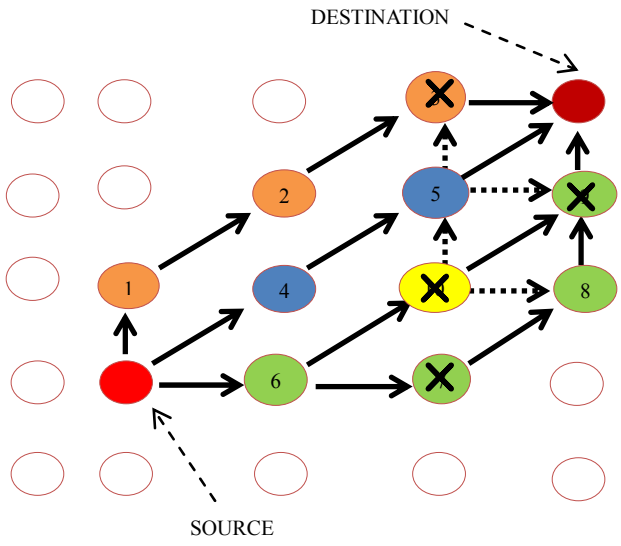

Figure 6. Appropriate route selection in WSN.

that is available between the SOURCE node and the DESTINATION (SINK) node. It can be proved as follows:

Suppose $T(i) = p$, where $T(i)$ is the trust level of the node i and p is the desired probability. If there are three paths P_1, P_2, P_3 consisting of n_1, n_2 and n_3 number of nodes respectively then $RT_{P1} = p * n_1$, $RT_{P2} = p * n_2$ and $RT_{P3} = p * n_3$. If $n_1 > n_2 > n_3$ then it is evident that $RT_{P3} < RT_{P2} < RT_{P1}$, as $0 \le p \le 1$ and vice-versa.

Figure 7 shows the advantage of our protocol as compared with the ATSR (LSRTP denotes DRTLSRP). Although in a few cases the performance of both are quite similar but in others our model scores over the ATSR one.

Figure 8 shows the plot of transmission latency with random trust assignment. Although we can't clearly decide which one is better, it's possible to conclude that in the long run our model behaves better than their one especially in the case of equal trusts and when the number of nodes in the network is very large.

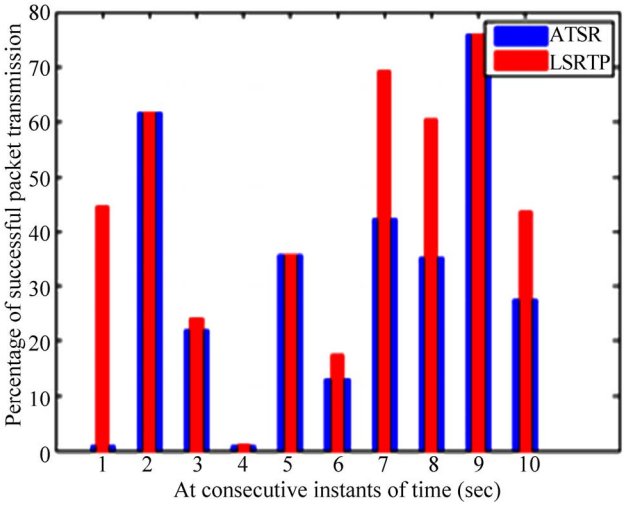

Figure 7. Packet transmission rate.

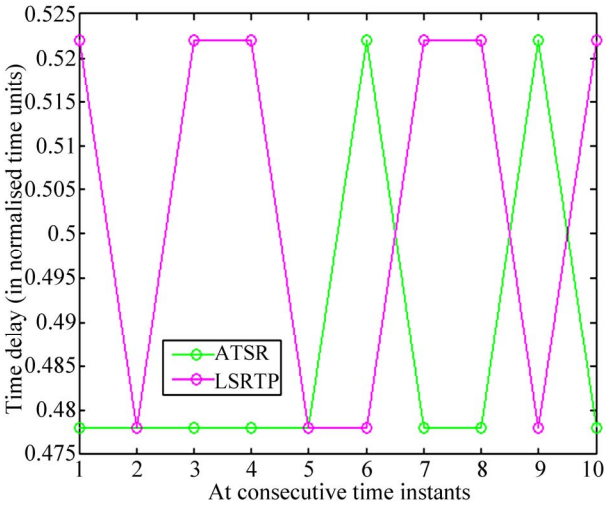

Figure 8. Packet delay.

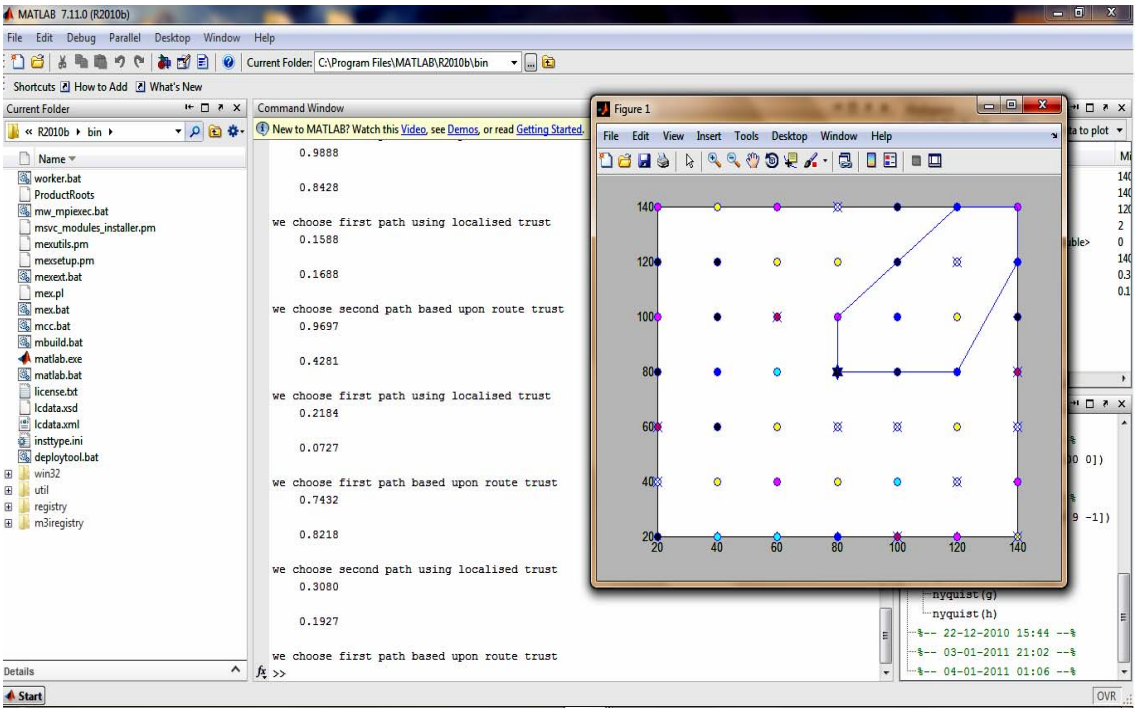

Figure 9. Matlab simulation model.

5. Conclusion and Future Work

It can be ultimately concluded from this simulation results (as shown in **Figure 9**) that our model performs better with respect to the available protocols such as ATSR, CONFIDANT etc. Future work includes inclusion of indirect trusts and calculation and assignment of route trust based on fuzzy logic.

6. References

[1] M. Momani, J. Agbinya, G. P. Navarrete and M. Akache, "A New Algorithm of Trust Formation in Wireless Sensor Networks," *Proceedings of the* 1*st IEEE International Conference on Wireless Broadband and Ultra Wideband Communications*, Sydney, March 2006.

[2] M. Momani, "Bayesian Methods for Modeling and Management of Trust in Wireless Sensor Networks," Ph.D. Thesis, University of Technology, Sydney, July, 2008.

[3] A. Sorniotti, L. Gomez, K. Wrora and L. Odorieo, "Secure and Trusted In-network Data Processing in Wireless Sensor Networks: A Survey," *Journal of Information Assurance and Security*, Vol. 2, No. 3, 2007, pp. 189-199.

[4] J. Lopez, R. Roman, I. Agudo and C. F. Gago, "Trust Management Systems for Wireless Sensor Networks: Best Practices," *Computer Communications*, Vol. 33, No. 9, 2010, pp. 1086-1093. doi:10.1016/j.comcom.2010.02.006

[5] T. Zahariadis, H. C. Leligou, P. Trakadas and Stamatis Voliotis, "Mobile Networks Trust Management in Wireless Sensor Networks," *European Transactions on Telecommunications*, Vol. 21, No. 4, 2010, pp. 386-395.

[6] G. X. Zhan, W. S. Shi and J. Deng, "TARF: A Trust-Aware Routing Framework for Wireless Sensor Networks," *Proceedings of the European Conference on Wireless Sensor Networks*, Coimbra, 17-19 February 2010, pp. 65-80.

[7] T. Kavitha, D. Sridharan, "Security Vulnerabilities in Wireless Sensor Networks: A Survey," *Journal of Information Assurance and Security*, Vol. 5, No. 1, 2010, pp. 31-44.

[8] J. Sen, "A Survey on Wireless Sensor Network Se- curity," *International Journal of Communication Net- works and Information Security*, Vol. 1, No. 2, August 2009, pp. 59-82.

[9] C. Karlof and D. Wagner, "Secure Routing in WSNs: Attacks and Countermeasures," *Ad Hoc Networks Journal*, Vol. 1, No. 2-3, September 2003, pp. 293-315. doi:10.1016/S1570-8705(03)00008-8

[10] A. A. Pirzada, C. McDonald and A. Datta "Performance Comparison of Trust-Based Reactive Routing Protocols," *IEEE Transactions on Mobile Computing*, Vol. 5, No. 6, June 2006, pp. 695-710.

[11] S. Zhu, S. Setia and S. Jajodia, "LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks," *Proceedings of the* 10*th ACM Conference on Computer and Communications Security*, Washington DC, 27-30 October 2003.

[12] A. Perrig, R. Szewczyk, V. Wen, D. Culler and J. D. Tygar, "SPINS: Security Protocols for Sensor Networks," *Proceedings of the* 7*th Annual International Conference on Mobile Computing and Networking*, Rome, 16-21 July 2001.

[13] C. Karlof, N. Sastry and D. Wagner, "TinySec: A Link Layer Security Architecture for Wireless Sensor Networks," *Proceedings of the* 2*nd International Conference on Embedded Networked Sensor Systems*, Baltimore, 3-5 November 2004.

[14] H.-R. Lee, Y.-J. Choi and H.-W. Kim, "Implementation of TinyHash Based on Hash Algorithm for Sensor Network," *Proceedings of the World Academy of Science, Engineering and Technology*, Saint Louis, Vol. 10, August 2005.

[15] M. Becker, S. Schaust and E. Wittmann, "Performance of Routing Protocols for Real Wireless Sensor Networks," *Proceedings of the* 10*th International Symposium on Performance Evaluation of Computer and Telecommunication Systems*, San Diego, July 2007.

[16] M. J. Probst and S. K. Kasera, "Statistical Trust Establishment in Wireless Sensor Networks," *Proceedings of the* 13*th International Conference on Parallel and Distributed Systems*, Hsinchu, Vol. 1, 5-7 December 2007.

[17] T. Zahariadis, H. Leligou, P. Karkazis, P. Trakadas, I. Papaefstathiou, C. Vangelatos and L. Besson, "Design and Implementation of a Trust-Aware Routing Protocol for Large WSNs," *International Journal of Network Security & Its Applications*, Vol. 2, No. 3, July 2010, pp. 52-68.

[18] Y. C. Hu, A. Perrig and D. B. Johnson, "Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks," *Proceedings of the* 8*th Annual International Conference on Mobile Computing and Networking*, Atlanta, 23-28 September 2002, pp. 12-23. doi:10.1145/570645.570648

[19] X. Q. Li, M. R. Lyu and J. C. Liu, "A Trust Model Based Routing Protocol for Secure Ad-Hoc Networks," *Proceedings of the IEEE Conference on Aerospace*, Big Sky, Montana, Vol. 2, 6-13 March 2004.

[20] A. A. Pirzada and C. McDonald, "Trusted Greedy Perimeter Stateless Routing," *Proceedings of the* 15*th IEEE International Conference on Networks*, Adelaide, 19-21 November 2007, pp. 19-21. doi:10.1109/ICON.2007.4444087

[21] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen and D. Culler, "SPINS: Security Protocols for Sensor Networks," *ACM Journal of Wireless Networks*, Vol. 8, No. 5, September 2002, pp. 521-534. doi:10.1023/A:1016598314198

[22] S. Marti, T. Giuli, K. Lai and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad-Hoc Networks," *Proceedings of the* 6*th Annual International Conference on Mobile Computing and Networking*, ACM Press, Boston, 6-11 August 2000, pp. 255-265. doi:10.1145/345910.345955

[23] S. Buchegger and J. Boudec, " Performance Analysis of the CONFIDANT Protocol: Cooperation of Nodes – Fairness in Distributed Ad-Hoc Networks," *Proceedings of the* 3*rd ACM International Symposium on Mobile Ad-Hoc Networking and Computing* (MobiHoc), ACM Press, Lausanne, 9-11 June 2002, pp. 226-236.

[24] S. Tanachaiwiwat, P. Dave, R. Bhindwale and A Helmy, "Location-Centric Isolation of Mis-behavior and Trust Routing in Energy-Constrained Sensor Networks," *Proceedings of the IEEE International Conference on Performance*, *Computing and Communications*, Phoenix, April 2004, pp. 463-469.

