Wireless Sensor Network, 2014, 6, 157-172
Published Online August 2014 in SciRes. http://www.scirp.org/journal/wsn
http://dx.doi.org/10.4236/wsn.2014.68016

Trust Evaluation Based on Node's Characteristics and Neighbouring Nodes' Recommendations for WSN

Shaik Sahil Babu[1], Arnab Raha[2], Mrinal Kanti Naskar[1]

[1]Department of Electronics and Telecommunication Engg., Jadavpur University, Kolkata, India
[2]School of Electrical and Computer Engineering, Purdue University, West Lafayette, USA
Email: sksahilbabu419@gmail.com, arnabraha@gmail.com, mrinaletce@gmail.com

Received 19 June 2014; revised 21 July 2014; accepted 19 August 2014

Abstract

In Wireless Sensor Networks (WSNs), the traditional cryptographic mechanisms for security require higher consumption of resources such as large memory, high processing speed and communication bandwidth. Also, they cannot detect faulty, malicious and selfish nodes which lead to the breakdown of network during packet routing. Hence, cryptographic security mechanisms are not sufficient to select appropriate nodes among many neighbouring nodes for secure packet routing from source to sink. Alternatively, trust management schemes are tools to evaluate the trust of a node and thereby choosing a node for routing, and also detecting their unexpected node behaviour (either faulty or malicious). In this paper, we propose TENCR: a new Trust Evaluation method based on the Node's QoS Characteristics (trust metrics) and neighbouring nodes' Recommendations. The proposed new technique detects the malicious and selfish nodes very efficiently than the arithmetic mean based methods, and allows trustworthy nodes in routing, thereby eliminating malicious/selfish nodes. Our proposed trust evaluation method is adaptive and energy efficient that separates the trustworthy nodes and qualify them to take the participation in routing, and disqualify the other nodes as malicious/selfish. Hence, trustworthy nodes only be allowed in routing, and malicious/selfish nodes will be eliminated automatically.

Keywords

Wireless Sensor Network, Trust, Security

How to cite this paper: Babu, S.S., Raha, A. and Naskar, M.K. (2014) Trust Evaluation Based on Node's Characteristics and Neighbouring Nodes' Recommendations for WSN. *Wireless Sensor Network*, **6**, 157-172.
http://dx.doi.org/10.4236/wsn.2014.68016

1. Introduction

A WSN consists of large number of spatially distributed autonomous wireless sensor nodes to cooperatively monitor physical or environmental conditions, such as temperature, humidity, light intensity, sound, vibration, pressure, motion etc. The nodes are densely deployed in a hazardous/unattended environment with the capability of sensing, computing and communicating wirelessly to the base station (BS), via neighbouring benevolent nodes [1]-[4]. The traditional security systems proposed in [5]-[9] such as public-key cryptographic, authentication and other mechanisms cannot be applied directly because they are computation intensive and each sensor node in WSN, is limited in its memory, battery life, computation and communication capabilities. Also, WSNs are susceptible to many varieties of attacks [10]-[13] like node capture, eavesdropping, worm-hole, Sybil attack, sink-hole and denial of service etc. Due to these attacks, total breakdown of network may take place. Hence, cryptographic and authentication methods are not sufficient to detect the malicious and/or selfish nodes and to separate the appropriate nodes among many neighbouring nodes for secure message routing from source to destination. Therefore, a new way of security mechanism that consumes less resources of network and detects malicious nodes easily, is required. *Trust*, the degree of reliability [14] of other node in performing actions, *i.e.* in QoS characteristics is a new mechanism for secure packet routing. The trustworthiness of a node on any neighbouring node can be evaluated based on its previous transaction records.

In this paper, we extend our previous work presented for the calculation of trust, based on direct trust only as in [15]; to a new approach of trust calculation considering both direct trust and indirect trust. The direct trust is evaluated based on node’s performance in QoS characteristics, *i.e.* in trust metrics such as data packets and control packets forward, data rate, power consumption, reliability, etc. is presented. Similarly, the indirect trust is evaluated based on the recommendations given by node's neighbouring nodes. The direct trust is combined with indirect trust using the traditional weighting approach to calculate total trust of a node. We have presented simulated results graphically, showing the trustworthiness and risk formed among the nodes for different trust thresholds and for different percentage of malicious nodes. Also, we have shown simulated results for the detection of malicious nodes and percentage of packet loss in the network. Rest of the paper is organized as follows: we present existing trust evaluation methods in Section 2, the proposed TENCR: Trust Evaluation method for WSNs based on Node’s QoS Characteristics and neighbouring nodes Recommendations in Section 3, while performance evaluation is included in Section 4 and finally, Section 5 concludes the paper.

2. Trust Models for Sensor Networks

There are many trust evaluation techniques available in the literature [14]-[18] for WSNs. In [14] Mohammad Momani defined the trust in different ways, as per him trust is the degree of reliability on other node/s in performing actions and can be formed by maintaining a record of the transactions with the nodes directly as well as indirectly. Trust can also be stated as a level of confidence that the assigned work can be done at a particular moment. This level of confidence can be extracted from the past history of transactions. 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 neighbouring nodes. We need mathematical models to represent trust and reputation and update these continuously.

Trust management system for WSNs is a mechanism that can be used to support the decision-making processes of the network [13]. It aids the members of WSN to deal with uncertainty about the future actions of other participants. As WSNs are highly application oriented, these applications bring various security needs. Survival of a WSN is dependent on the cooperative and trusting nature of its nodes. Hence, the trust establishment between nodes is must. Overview of the existing trust evaluation techniques proposed in different trust management systems and trust based routing protocols are given below.

1) Mohammad Momani [14] introduced a computational model for trust in his doctoral thesis. He proposed different methods for modelling and managing trust to enable WSN to be secure and reducing the computing and communication overheads. He proposed one algorithm for trust calculation and risk assessment based on trust factors and dynamic aspects of trust. He modelled the direct trust computation with direct experiences, and indirect trust with recommendations given by the neighbours. He modelled total trust using traditional weighting approach for direct trust and indirect trust as shown in following equations. DT is direct trust (experience), IT is indirect trust (recommendations), T is total trust. The direct trust DT of A on B is given by the following equation.

$$DT^{A,B} = \sum_{m=1}^{k} W_m \times tm_m^{A,B}$$

The indirect trust IT of node A on node B is given by the following equation.

$$IT^{A,B} = \frac{1}{n}\sum_{j=1}^{k}\left(W_{A,N_j} \times T^{N_j,B}\right)$$

where W_{A,N_j} is weight for j^{th} neighbouring node recommendation.

The total trust T is function of DT and IT, and is given by the following equation where W_D and W_I weights assigned to each.

$$T = W_D \times DT + W_I \times IT$$

2) A novel flexible trust management system proposed in [16] defined the trust as the ratio of successful transactions *S* to the total transactions made by the node. The proposed model is a decentralized trust scheme, *i.e.* the trust management functionality is distributed over the network nodes. In this model, each node is responsible for computing its own trust value per relation in the network, collecting direct and indirect information. The both direct and indirect trust values are used to evaluate each node's trustworthiness. The proposed model has inherent reputation scheme of getting trustworthiness of any node, when direct evidences does not suffice, *i.e.* the number of direct evidences remain under threshold. Node A calculates trust value regarding node B based on the following equation:

$$T_m^{A,B} = S_m^{A,B} \Big/ \left(S_m^{A,B} + F_m^{A,B}\right)$$

where *S* is successful and *F* is unsuccessful transactions.

$$DT^{A,B} = C^{A,B} \times \sum_{m=1}^{k}\left(W_m \times T_m^{A,B}\right)$$

where W_m is weight of each trust metric.

$$IT^{A,B} = \sum_{j=1}^{n}\left(W_{A,N_j} \times DT^{N_j,B}\right)$$

where W_{A,N_j} is weight for recommendation made by j^{th} neighbour.

$$T^{A,B} = W_D \times DT^{A,B} + W_I \times IT^{A,B}$$

3) In trust management scheme "Securing Geographic Routing in WSN" proposed in [17], honest nodes are favoured by giving them the credit for each successful packet forwarding, while penalizing suspicious nodes that exaggerate their contribution to routing. Initially, trust value 0.5 will be assigned to the node. Then, it monitors the behaviour of the nodes (one hop neighbouring nodes) to which it forwards packets.

$$T_{i_{\text{new}}} = T_i + \Delta t, \quad \text{if } \left(T_i + \Delta t\right) \text{is} \le 1$$

= 1 otherwise. Δ*t*, step size is 0.01.

$$T_{i_{\text{new}}} = T_i - \Delta t, \quad \text{if } \left(T_i - \Delta t\right) \text{is} > 0$$

= 0 otherwise. Δ*t* is predefined penalty.

4) Trust establishment system "Defense of Trust Management Vulnerabilities in Distributed Networks" proposed in [11] has two ways to establish trust in computer networks. First, when the subject (first party) can be directly observing the agent (second party). Second, when the subject receives recommendations from other entities about the agent, indirect trust can be established.

Direct Trust *DT* = (*S* + 1)/(*S* + *F* + 2) where S, F are the number of successful and unsuccessful interactions respectively between the Subject and Agent. Initially *DT* = 0.5 because *S* = *F* = 0.

Indirect Trust IT = Recommended by others.

5) A trust model for WSN using fuzzy logic is proposed by Tae Kyung Kim, and Hee Suk Seo in [18] for centralized network. Reputation defined as a perception of a party, created through past actions about its inten-

tions and norms. The different components in the suggested trust modelling are minimum trust, maximum trust and un-trust. The proposed model calculates the trust level of a sensor node, trustworthiness T, untrustworthiness U assuming that the wireless sensor network has the reputation value of each sensor node. In reputation components, minimum T, maximum T, minimum U and maximum U are defined between each pair of nodes. From this, trust T and un-trust U are acquired. Using the T and U, the trust evaluation level is acquired.

The trust evaluation between two nodes I and J is shown below, where T_i is trust of node I on node J and T_j is trust of node J on node I and U is the corresponding un-trust.

$$T = \operatorname{avg}\left(T_i, T_j\right) \Big/ \left[1 - \left(\operatorname{avg}\left(T_i, U_j\right) + \operatorname{avg}\left(T_j, U_i\right)\right)\right], \quad U = \operatorname{avg}\left(U_i, U_j\right) \Big/ \left[1 - \left(\operatorname{avg}\left(T_i, U_j\right) + \operatorname{avg}\left(T_j, U_i\right)\right)\right]$$

Trust evaluation between I and $J = T/(T + U)$.

3. Proposed Technique for Trust Evaluation

In this section, we propose a new trust evaluation method TENCR for WSNs. By applying this technique, any node of WSN can evaluate how much trust it is having on its neighbouring nodes. Here, neighbouring nodes means, nodes those can be connected by node’s radio signal. The trust, which is the level of confidence, is a time dependent entity. That means the trust may vary as time goes on based on the nodes’ behaviour in transactions performed among them. The trust can be evaluated from the history of transactions with the node and from the recommendations given by the other neighbouring nodes. Here, the history means the behaviour of the node in different aspects, *i.e.* trust metrics, also called Quality of Service Characteristics [10] [16]. The level of confidence extracted from trust metrics is called direct trust (DT). The indirect trust (IT) can be extracted from the recommendations, called indirect information given by the neighbouring nodes. The overall Trust (T) on any node can be formed by manipulating these direct and indirect trusts. As shown in **Figure 1**, node A is evaluating trust on node B. It evaluates direct trust from its direct experiences and indirect trust from the information given by the neighbouring nodes.

Some of the useful trust metrics of WSNs are listed in **Table 1**. Each of the nodes of WSN shall update the trust metrics of its neighbouring nodes for every event occurred in the network. The indirect trust (IT) on any neighbouring node can be calculated by gathering the information about that node from all other neighbours.

Every node in the network keeps an eye on the behaviour of their neighbouring nodes and maintains a record on them for every event happening in the network. This record contains the information about neighbouring node QoS characteristics. These trust metrics data will be helpful for calculating the direct trusts on them. Also, as and when requested by neighbouring nodes, trust of one node, can be transferred to other nodes, and there it helps in evaluating the indirect trust. In our trust evaluation method, trust of any node on its any neighbouring node is a function of DT and IT. The trust metrics [16] listed in **Table 1**, can defined as below:

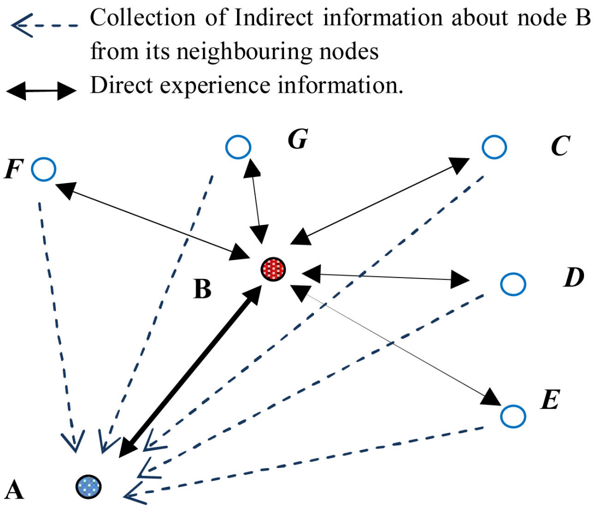

Collection of Indirect information about node B from its neighbouring nodes
Direct experience information.
G
C
F
B
D
E
A

Figure 1. Node A evaluating trust on node B.

Table 1. Trust metrics.

1)	Data packets forwarded
2)	Data packet/message precision
3)	Control packet/message forwarded
4)	Control packet/message precision
5)	Availability based on beacon/hello messages
6)	Routing protocol execution (routing actions)
7)	Message Cryptography
8)	Consistency of reported (sensed) values/data
9)	Sensing communication
10)	Reputation
11)	Packet address modified
12)	Battery lifetime
13)	Packet Delay

Availability to hello messages: Detection of neighbouring nodes those are available in the radio range of any node, and to examine them for packet forwarding.

Packet forwarding: Detection of nodes that deny to forward packets, acting in malicious manner, each time a source node sends a packet to a neighbour for further forwarding, it enters the promiscuous mode and overhears the wireless medium to check whether the packet was actually forwarded by the selected neighbour.

Acknowledgements: Detecting the successful end-to-end forwarding of the messages, we suggest that each source node waits for a acknowledgement per transmitted message to check whether the message has successfully reached the Sink node.

Packet delay: Detection of the time delay taken by the packet to reach the Sink node successfully, and to find network congestion status from source-to-end.

Packet precision: Each time a source node transmits a packet for forwarding and then overhears the wireless medium to ensure that the packet was forwarded, it additionally processes it to check the packet's integrity, *i.e.* that no unexpected modification has occurred.

Routing protocol execution: All the nodes may not having the routing function. Some nodes meant for only monitoring the environment, and some may be only for forwarding packet and some for both the functions.

Reputation: In the trust evaluation technique the node's indirect information will be requested from the node's neighbouring nodes. It will be helpful when there is no direct trust information regarding that node.

Remaining Energy: Although the energy level of each neighbour is not a pure trust metric, taking into account the remaining energy level, apart from extending the network lifetime, contributes towards load balancing (partially defending against the traffic analysis attack). In our novel routing protocol, the remaining energy value is piggy-backed in the Beacon message.

In our trust evaluation method, we are dividing trust metrics into two categories called higher priority category, and lower priority category. This is because, we do not want to compromise in some fundamental functional characteristics of a node. A node's main functionality can be seen by its higher priority trust metric levels. Hence, node's higher priority trust metrics shouldn't fall below the threshold level. For example, the trust metrics Data packets forwarded, Control packets/Messages forwarded and Reputation values should not be less than the higher priority threshold because node's functionality is hidden in these trust metrics. All other trust metrics can be considered as of lower priority category.

The direct trust on any neighbouring node can be evaluated from higher priority trust metrics and lower priority trust metrics. A node should maintain minimum trust threshold in the higher priority trust metrics hence, they are multiplied or geometric mean value will be taken. Similarly, all other trust metrics (of lower priority), are averaged or arithmetic mean value will be taken. To evaluate the direct trust, these two are combined by giving some weight to each one. Here, a node can become malicious/selfish/un-trusted at two different stages, one is, when any one of the higher priority trust metrics are falling below the threshold, and second, over all direct trust is falling down by direct trust threshold. The indirect trust on any node, actually depends on the neighbouring nodes recommendations on that node. The indirect trust (IT) on any neighbouring node can be evaluated by gathering the information about that node from all other neighbouring nodes. The neighbouring nodes are divided into two categories, most trusted neighbour and normal neighbour. Every node not only maintains the trust metric data but also, maintains their trusts history. Based on their history, some neighbouring nodes are most trustworthy and can be separated into most trusted/high reputation neighbours and some are into less trusted neighbours. The most trustworthy neighbouring node must recommend positively. Similarly, all other neighbouring nodes are of lower priority, and their recommendations will be averaged. To evaluate the indirect trust, these two are combined by giving some weight to each one. Here, a node can become malicious/selfish/un-trusted at two different stages, one is, when any one of the higher priority neighbouring node is not recommending, *i.e.*, most trusted neighbouring node recommendation is falling below the threshold, and second, over all indirect trust is falling down by indirect trust threshold.

The overall trust or total trust T of any node on any neighbouring node is again a function of direct trust (DT) and indirect trust (IT). Our proposed model uses the traditional weighting approach for combining direct trust and indirect trust to form the total trust (T) per relation in the network.

3.1. Direct Trust (DT)

One of the most important aspects of trust management schemes is the process of data collection for trust calculation. The Direct Trust value of a neighbouring node can be determined by the different trust metrics of that particular node in different events occurred in the network. The trust metrics, *i.e.* the QoS characteristics that can be taken into account are given in [16]. The Direct Trust is a function of trust metrics. The listed trust metric data for different events are essential and can provide a useful feedback to the system, towards the proper decision making by the trust management system in the node. Here, depending on the application, we can insist the minimum level (threshold) to all the trust metrics, or we can have different thresholds to different groups of trust metrics. Once one/more trust metric threshold/s are fixed, our trust management system considers that no node is trusted unless it is having minimum threshold level in a given trust metric strictly. This is the main advantage of our proposed trust management model compared with other models. And, this is where our proposed Trust Evaluation model filters the malicious and selfish nodes in participating in the packet routing in WSN.

As already mentioned, all the trust metrics are divided into two categories called higher priority and lower priority trust metrics as shown in **Figure 2**. Let $tm_m^{A,B}$ be the set of different higher priority trust metrics on node A on node B for different trust metrics, where m = 1 to k. And Let $tm_n^{a,b}$ be the set of different lower priority trust metrics on node A on node B for different trust metrics, where n = 1 to l. Also, let W_H^{DT}, W_L^{DT} are weights given to higher priority trust metrics and lower priority trust metrics respectively in finding the Direct Trust. Here, $W_H^{DT} + W_L^{DT} = 1$.

In our trust management system, the direct trust of any node on any other neighbouring node is weighted sum of the geometric mean of all higher priority trust metrics, provided every higher priority trust metric is greater than or equal to threshold, and arithmetic mean of all lower priority trust metrics. The set of trust metrics are called trust metric vector. Every node consists of set of metric vectors called trust metric matrix of its neighbours. From these records, Direct Trust (DT) of neighbouring node is calculated, and is shown below.

$$DT^{A,B} = W_H^{DT} \times \left[\prod \left(tm_1^{A,B}, tm_2^{A,B}, tm_3^{A,B}, \cdots, tm_k^{A,B} \right) \right]^{(1/k)} + W_L^{DT} \times \frac{1}{l} \left[\sum \left(tm_1^{a,b}, tm_2^{a,b}, tm_3^{a,b}, \cdots, tm_l^{a,b} \right) \right]$$

$$DT^{A,B} = W_H^{DT} \times \left[\prod_{m=1 \text{ to } k} tm_m^{A,B} \right]^{1/k} + W_L^{DT} \times \frac{1}{l} \left[\sum_{n=1 \text{ to } l} tm_n^{a,b} \right] \quad (1)$$

3.2. Indirect Trust (IT)

The indirect trust of any node on any neighbouring node can be evaluated from the indirect information given by the neighbouring nodes. Again, as in trust metrics, the neighbouring nodes also divided into most trusted and normal other neighbouring nodes. The geometric mean will be applied to the information given by the most trusted neighbouring nodes and arithmetic mean will be applied to the information given by the other normal neighbouring nodes. Let us assume for the calculation of IT the set of nodes (say 8) are situated in some nearby area in the WSN field as shown in **Figure 3**. Their names are A, B, C, D, E, F, G, H and I. Here, we are interested to find the IT of node A on the neighbouring node B, *i.e.* $\text{IT}^{\text{A, B}}$. The node A, first collects the recommendations from their neighbouring nodes. In this case the nodes C, D, E, F, G, H and I are neighbours. Assume that

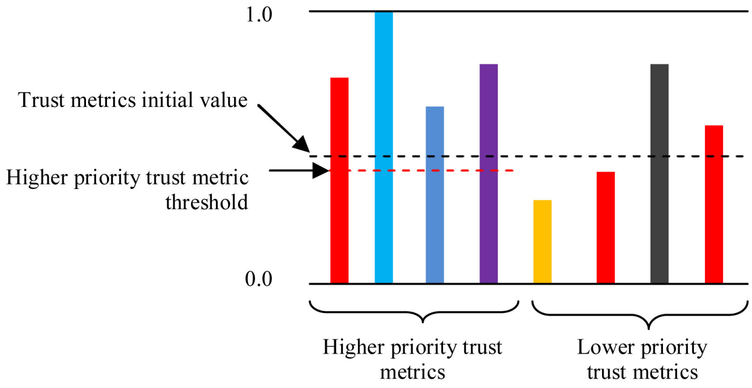


Figure 2. Different trust metrics levels.

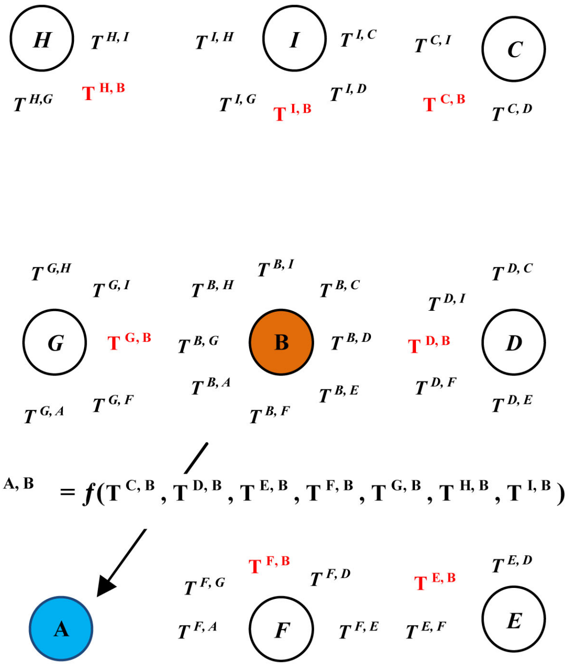


Figure 3. Indirect trust of node A on node B.

collected recommendations are $T^{C,B}$, $T^{D,B}$, $T^{E,B}$, $T^{F,B}$, $T^{G,B}$, $T^{H,B}$ and $T^{I,B}$ respectively. Among the neighbouring nodes, let us say G and F are most trusted neighbours. The indirect trust is a function of indirect information given by most reputed nodes (higher priority node trusts) and ordinary neighbour nodes. Here, the geometric mean function is applied to most reputed category and arithmetic mean is applied to less reputed category with weights W_H^{IT}, W_L^{IT} respectively. From **Figure 3**, the following three equations can be extracted.

IT = Geometric mean of trusts given by higher priority neighbour nodes + Average of trusts given by lower priority neighbour nodes.

In general, the indirect trust IT of node A on node B is given by the following equation.

$$IT^{A,B} = W_H^{IT} \times \left[\prod_{i=1 \text{ to } r} W_{A,N_i} \times T^{N_i,B} \right]^{1/r} + W_L^{IT} \times \frac{1}{s} \sum_{j=1}^{s} \left(W_{A,N_j} \times T^{N_j,B} \right) \quad (2)$$

where, W_{A,N_j} is weight for recommendation made by i^{th} neighbour.

Selection of higher priority neighbouring nodes:

As already explained, every node of WSN shall maintain the database for its every neighbouring node, related to the transactions occurring in the network. Among the neighbouring nodes, some nodes may be the most trusted neighbouring nodes. If any neighbouring node is maintaining its trust (T) more than or equal to the trust threshold level on average all the time or say for the last x events/transactions, then it can be the most trusted neighbour node. **Table 2**, shows the history of 8 neighbouring nodes of node A for last x events. It is the node's responsibility to check its database, and selects few most trusted nodes if any. The indirect information given by such neighbouring nodes would be applied to the geometric mean function in the equation of indirect trust (IT). The indirect information given by all other neighbouring nodes would be applied to the average function in the calculation of IT.

3.3. The Trust (T)

The overall trust or total trust (T) of any node on any neighbouring node is again a weighted some of direct trust (DT) and indirect trust (IT). $T = W_D \times DT + W_I \times IT$.

The W_D is a weight given to direct trust and W_I to the indirect trust where $W_D + W_I = 1$. Weights can be assigned using different approaches. Sometimes DT may be given more weight, and IT may be given less weight *i.e.* $W_D > W_I$. Hence, we define a new equation for trusting one sensor node by the other in Wireless Sensor Networks as shown below.

$$T^{A,B} = W_D \times DT^{A,B} + W_I \times IT^{A,B}$$

$$\begin{aligned} T^{A,B} &= W_D \times \left\{ W_H^{DT} \times \left[\prod_{m=1 \text{ to } k} tm_m^{A,B} \right]^{1/k} + W_L^{DT} \times \frac{1}{l} \left[\sum_{n=1 \text{ to } l} tm_n^{a,b} \right] \right\} \\ &\quad + W_I \times \left\{ W_H^{IT} \times \left[\prod_{i=1 \text{ to } r} W_{A,N_i} \times T^{N_i,B} \right]^{1/r} + W_L^{IT} \times \frac{1}{s} \sum_{j=1}^{s} \left(W_{A,N_j} \times T^{N_j,B} \right) \right\} \end{aligned} \quad (3)$$

where, A is the node that is evaluating its trustworthiness on node B,

k is higher priority trust metrics, $\boldsymbol{l}$ is lower priority trust metrics,

r is most trustworthy neighbours, $\boldsymbol{s}$ is ordinary neighbours,

$tm_m^{A,B}$ is m^{th} higher priority trust metric of node A on node B,

$tm_n^{a,b}$ is n^{th} lower priority trust metric of node A on node B,

W_H^{DT} is higher priority trust metric weight, W_L^{DT} is lower priority trust metric weight.,

W_H^{IT} is higher priority neighbour weight, W_L^{IT} is lower priority neighbour weight,

W_{A,N_j} is weight for recommendation made by j^{th} neighbour,

$T^{N_i,B}$ is trust given by neighbour N_j about node B, and

W_D, W_I are weights of direct and indirect trusts respectively.

Figure 4, shows the relation among the direct trust, indirect trust and the trust graphically for $W_D = 0.75$ and $W_I = 0.25$ respectively. From **Figure 4**, we can also see the trustworthy area where trust quantity will be towards 1.0 and risky area where trust quantity will be towards 0.0.

Table 2. Node A's neighbouring node trusts for different events.

Neighbour	Event e_0	Event e_1	Event e_2	Event e_3	Event e_4	...	Event e_x
C	$T_{e_0}^{A,C}$	$T_{e_1}^{A,C}$	$T_{e_2}^{A,C}$	$T_{e_3}^{A,C}$	$T_{e_4}^{A,C}$	...	$T_{e_x}^{A,C}$
D	$T_{e_0}^{A,D}$	$T_{e_1}^{A,D}$	$T_{e_2}^{A,D}$	$T_{e_3}^{A,D}$	$T_{e_4}^{A,D}$	...	$T_{e_x}^{A,D}$
E	$T_{e_0}^{A,E}$	$T_{e_1}^{A,E}$	$T_{e_2}^{A,E}$	$T_{e_3}^{A,E}$	$T_{e_4}^{A,E}$	...	$T_{e_x}^{A,E}$
F	$T_{e_0}^{A,F}$	$T_{e_1}^{A,F}$	$T_{e_2}^{A,F}$	$T_{e_3}^{A,F}$	$T_{e_4}^{A,F}$	...	$T_{e_x}^{A,F}$
G	$T_{e_0}^{A,G}$	$T_{e_1}^{A,G}$	$T_{e_2}^{A,G}$	$T_{e_3}^{A,G}$	$T_{e_4}^{A,G}$	...	$T_{e_x}^{A,G}$
H	$T_{e_0}^{A,H}$	$T_{e_1}^{A,H}$	$T_{e_2}^{A,H}$	$T_{e_3}^{A,H}$	$T_{e_4}^{A,H}$	...	$T_{e_x}^{A,H}$

3.4. Adaptive Trust Evaluation Algorithm

The proposed trust evaluation method is an adaptive trust evaluation method. First, it tries to find the trustworthy neighbouring node from the available trust metric database. If any node's trust value greater than or equal to trust threshold then that neighbouring node will be selected for packet transmission for that moment. But, if no node found trustworthy then it adapts another way to find and evaluate the trustworthy node for packet routing. Node makes the transactions with neighbouring nodes and gets the indirect information from all neighbours those are situated in its radio range. Then it updates the database of the node, and evaluates the trusts and chooses the best node, if not better node with risk factor. If trustworthy node found as a result of first step, *i.e.* benevolent node from already available database then much energy of a node can be saved. When no trustworthy neighbour node found from its radio range, the node increases its radio communication range to find good neighbouring nodes through indirect information. This is an energy consumption process which is adaptive and occurs in risky situations. Hence, this evaluation method saves the energy of node as long as the trusts of neighbouring node is greater than or equal to trust threshold. This increases the life of the node as well as lifespan of the entire WSN.

When node wants to make the communication with its neighbouring nodes for indirect information data, it simply sends a request command to its neighbours. This command may take only one byte or two byte in size. But, when it is getting indirect information the size of the message from each neighbouring node may be approximately 20 to 25 bytes by assuming that each node's information takes 2 bytes and it may have 10 neighbours. The communication among nodes for indirect information will take place only when there is risky situation and not all the time as explained in the following algorithm. Hence, this adaptive TENCR method is not only energy efficient but also decreases the communication overhead.

```
Algorithm for node selection for packet handing over:
    while (data packet is ready) {
            evaluate trusts of all neighbouring nodes from the database;
            for all neighbouring nodes ($n_i$) {
                if (node $n_i$ trust  ≥  $T_{th}$){
                    handover the packet to neighbouring node $n_i$ ;
                    trustworthy_node_found = 'yes'; return;}
                trustworthy_node_found = 'no';}
            if(trustworthy_node_found == 'no'){ //node energy consuming operation
                    get new record of indirect information from neighbouring nodes;
                    update the database; evaluate trusts of all neighbouring nodes;
                    for all neighbouring nodes ($n_i$){
                        if (node $n_i$ trust  ≥  $T_{th}$ ){
                            handover the packet to neighbouring node $n_i$ ;
                            trustworthy_node_found = 'yes'; return;}
                    trustworthy_node_found = 'no';}}
            if(trustworthy_node_found == 'no'){//max. node energy consuming operation
                    increase the radio range r;
                    get the new record of indirect information from neighbouring nodes;
                    update the database; evaluate trusts of all neighbouring nodes;
                    for all neighbouring nodes ($n_i$){
                            if (node $n_i$ trust  ≥  $T_{th}$){
                                handover the packet to neighbouring node $n_i$ ;
                                trustworthy_node_found = 'yes'; return;}
                            trustworthy_node_found = 'no'; }}
            if(trustworthy_node_found == 'no'){//max. node energy consuming operation
                    handover the packet to highest trust neighbouring node with risk;
                    return;}
            update the database;
    } //endwhile
```

3.5. Advantages

This method provides many advantages as compared to the existing ones. First, it allows us to find out the Trust (T) levels of all neighbouring nodes, even when the trust threshold (T_{th}) cannot be decided or evaluated, and hence the separation of benevolent and malicious nodes is possible before selection of any neighbouring node for data/packet transmission. The second advantage is that this method allows us to give more weight to certain trust metrics, also to some neighbouring nodes, depending on the requirement of the application. The third and

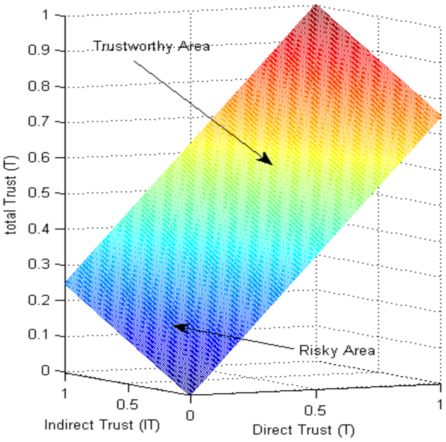


Figure 4. Relation between direct trust, indirect trust and trust.

important advantage of this Trust Evaluation method is that the calculation of direct trust and indirect trust of any neighbouring node is not on the basis of the average of the trust metrics as in the case of other trust models. If it is on the basis of average function, then the neighbouring node can be trustworthy even if some major trust metrics of neighbouring node like, *data packets forwarded*, *control packets forwarded* are fallen down to 0 or less than the T_{th}, which should not be the case. This situation occurs when WSN is attacked by the Sybil node or malicious node. But, in our proposed trust evaluation model, a node is treated as faulty or malicious when one of the trust metrics is failed to form trustworthy relation. Hence, in this Trust Evaluation method, the malicious and/or selfish nodes can easily be detected and discarded from making transactions with the other nodes.

4. Performance Evaluation

To evaluate the performance of the proposed TENCR technique, we have taken some assumptions in the Sensor Network. The Sink node sends a test signal to the all nodes of the network periodically. The period of the test signal is not defined and it can be application dependent. This test signal will be received by all nodes and it may contain any information from Sink node to a node/group of nodes/all nodes. Also, the test signal may be an instruction or can be node/s behaviour information in a particular period/transaction. Based on the application, the WSNs functionality can be centralized or decentralized. In a decentralized model, the trust evaluation and management functionality is distributed over the network nodes. Each node is responsible for computing trust value per relation in the network, collecting events from direct relations, and collecting trust values from other nodes in the network. The trust formation, evaluation and updating can be centralized where trust will be computed by one single node (may be Sink) periodically and will be communicated to all network nodes. Our proposed trust evaluation technique, the trust computation will be at the Sink node because of the resource constraints at sensor node.

Also, we assume that there are two main functions of WSN, initialization phase or setup phase and the normal routing or run phase. The setup/initialization phase is primarily to detect the malicious nodes of the WSN if any using our proposed trust evaluation method and the run phase is a normal packet routing phase with the benevolent nodes after eliminating the detected malicious nodes.

Initialization phase is also called trial phase. In this phase, all nodes will be deployed and will be initialized. For this, the Sink node will be sending appropriate signal to nodes. To detect the status, working conditions and maliciousness if any of all network nodes, Sink node instructs all the nodes to send some dummy packets from every node to Sink. In turn, these dummy packets from all nodes will be collected at Sink node. The Sink node collects these dummy packets and analyze them to know the working conditions of all nodes. As Sink node will be containing the central database for all nodes as well as for neighbouring nodes with respect to every node, it finds the malicious nodes of the network. Finally, the Sink node detects the all malicious nodes in the network, and informs to all benevolent nodes. Then in the second phase, normal routing will be performed without considering the malicious nodes. But, the Sink node will be evaluating the trusts of all nodes periodically and it will inform malicious node information if any to all nodes.

The performance of the proposed trust evaluation technique has been evaluated through computer simulations. Using MATLAB, a new simulation package for routing has been developed based on distance of neighbouring nodes towards the Sink node. We have used our trust evaluation mechanism TENCR in this protocol. The other settings and assumptions are given below.

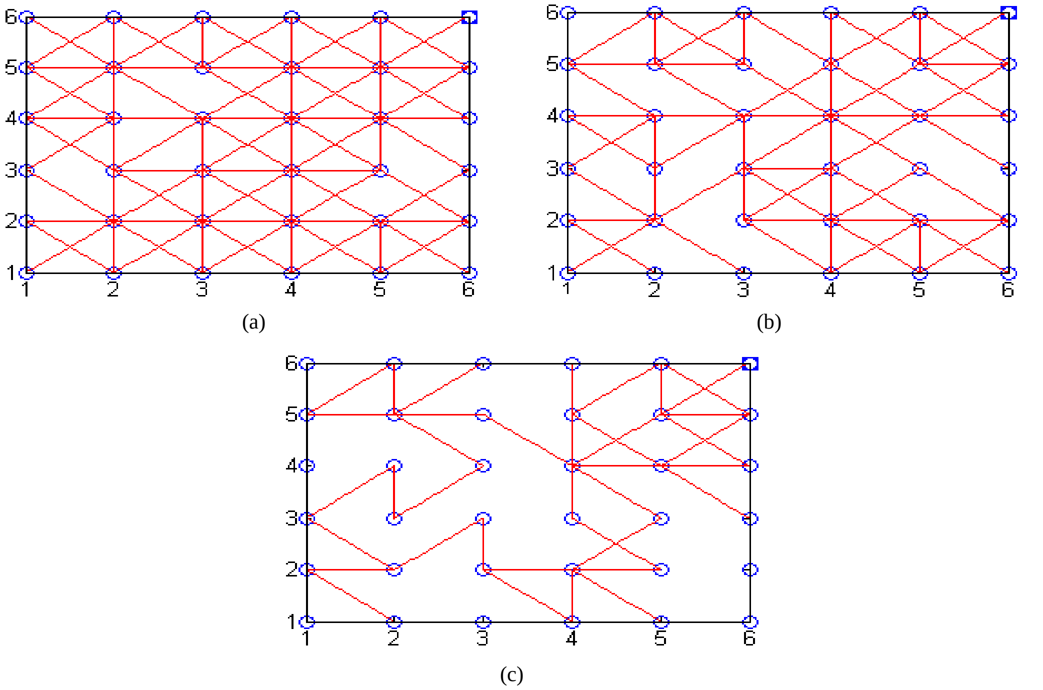
4.1. Trustworthy Relations for Randomly Taken Trust Metrics

4.1.1. Node Location Is Fixed Equally Spaced WSN in Square Area

Figure 5, shows trustworthy relations formed for randomly taken trust metric quantities. Here, WSN with 36 nodes deployed in square area as 6 × 6 nodes in X and Y directions. The radio range of the node here is assumed as it covers the one-hop nodes. That means one hop nodes (8 nodes) are neighbour nodes to any node. Here, our main aim is to find how the trustworthy relations are being formed among the nodes, depending on the trust threshold level. Both direct trust and indirect trust are considered for the calculation of total Trust. The number of trust metric categories are taken is 10, out of which 4 are taken as higher priority category and 6 are lower priority category. Similarly, every node is having 8 neighbour nodes, out of which 3 are most trusted neighbouring nodes and rest are ordinary neighbouring nodes. Finally, we have shown how the Trustworthy relations changed for different trust threshold levels graphically for four different cases.

4.1.2. Random Deployment and Fixed Location in a Square Area

WSN with random deployment with fixed node location in a square area as shown in **Figure 6**. Here, our assumptions are, a WSN with 100 nodes deployed in square area of 100 × 100 meters randomly in X and Y directions. The radio range of the node assumed here is 15 meters. The number of trust metric categories are taken is 8, out of which 3 are taken as higher priority category and 5 are lower priority category. The neighbour nodes of any node are dependent on the locations. Because locations are random, they vary node to node. Here, we are assumed 2 are most trusted neighbouring nodes and rest are ordinary neighbouring nodes. The other assumptions are same as the manual deployment WSN. **Table 3**, shows the detection of malicious nodes in two different trust evaluation systems with some randomly taken trust metric data and neighboring node recommendations. For cases 2 and 4, as shown in **Table 3**, though the higher priority trust metrics are below the threshold level,

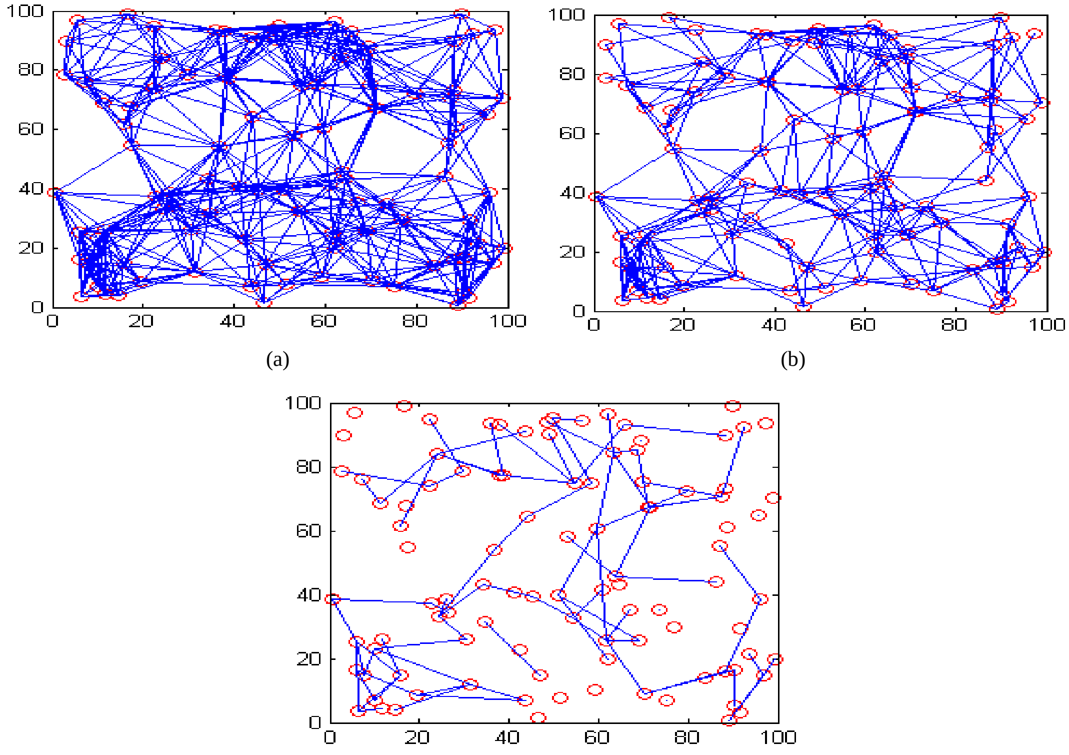


(a)

(b)

(c)

Figure 5. Trustworthy relations among the nodes. (a) Trusted relations for trust threshold ≥ 0.35; (b) Trusted relations for trust threshold ≥ 0.4; (c) Trusted relations for trust threshold ≥ 0.45.



(a)

(b)

(c)

Figure 6. Trustworthy relations among the nodes for random deployment. (a) Trusted relations for trust threshold ≥ 0.25; (b) Trusted relations for trust threshold ≥ 0.30; (c) Trusted relations for trust threshold ≥ 0.35.

Table 3. Detection of malicious nodes by TENCR for $T_{th} = 0.35$.

case	Trust metrics								Neighbouring nodes Recommendations					***Malicious node detection***			
	Higher priority			Lower priority					Higher priority		Lower priority			Arithmetic mean		**TENCR**	
	***tm*$_1$**	***tm*$_2$**	***tm*$_3$**	*tm*$_4$	*tm*$_5$	*tm*$_6$	*tm*$_7$	*tm*$_8$	**C**	**E**	**D**	**F**	**G**	**Trust**		**Trust**	
1	0.75	0.85	0.45	0.38	0.90	0.73	0.75	0.56	0.19	0.39	0.29	0.38	0.87	0.56		**0.52**	
2	0.57	0.34	**0.27**	0.36	0.62	0.72	0.18	0.56	**0.07**	0.19	0.49	0.29	0.38	0.41	Not detected	**0.30**	*Detected*
3	0.75	0.72	0.34	0.09	0.91	0.38	0.19	0.72	0.73	0.75	0.49	0.29	0.38	0.50		**0.39**	
4	0.35	0.56	**0.08**	0.28	0.72	0.91	0.45	0.38	0.67	**0.09**	0.45	0.51	0.78	0.50	Not detected	**0.32**	*Detected*
5	0.75	0.54	0.49	0.29	0.38	0.87	0.38	0.54	0.51	0.78	0.56	0.29	0.35	0.51		**0.41**	

the arithmetic/average based trust evaluation systems cannot detect the malicious nodes, whereas the TENCR detects these malicious nodes. Similarly, the most trusted neighboring node recommendations also, cannot be recognized in the arithmetic mean based trust evaluation systems.

4.2. Detection of Malicious Nodes in Setup Phase for Different Weights

During the setup/initialization phase, the malicious nodes can be caught by our trust evaluation method running in the Sink node. The performance of the proposed trust evaluation technique for different settings and assumptions of the network has been evaluated and shown graphically. For this, a new multi-hop routing package has been developed in which the next neighbouring node will be selected for handing over the packet based on distance of neighbouring nodes towards the Sink node. We have evaluated the performance of our trust evaluation mechanism TENCR with this multi-hop routing protocol. The other settings and assumptions are given below.

WSN deployment: Random in a square area

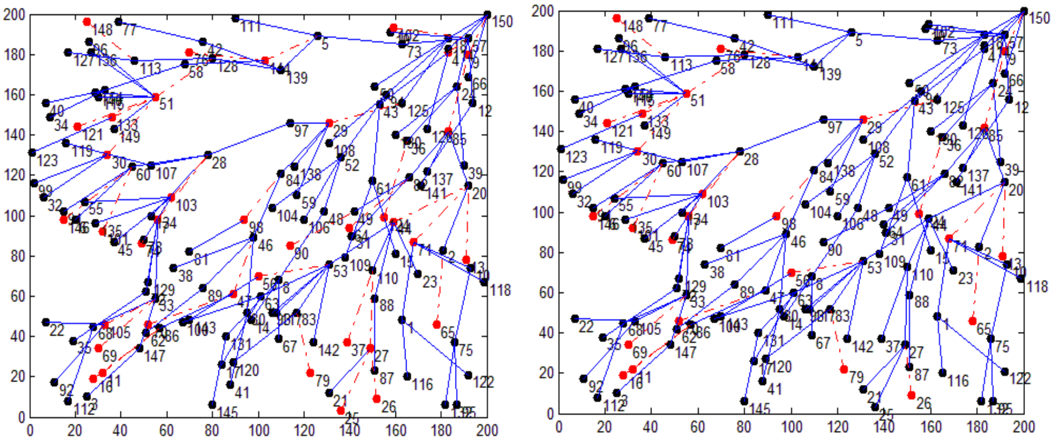
Mobility of nodes: Immobile
WSN area: 200 × 200 square meter
Sink location: top right corner
No. of nodes: 150
Node Radio Range: 30 meter
Trust metrics: 8 categories
Neighbouring nodes: max. 10% of total nodes
Initial Trust: 0.5 (initially all nodes are trusted)
Malicious nodes: 0% to 30% (red color marked)
Trust threshold (T_{th}): 0.48 to 0.5
Direct Trust weight: varies from 0.75 to 0.5,
Indirect Trust weight: varies from 0.5 to 0.75
Packet generation: randomly with Poisson probability of 0.3

Figure 7(a) and **Figure 7(b)** shows, the graphical view of WSN for random deployment, and packet routing from nodes to Sink for 30% of malicious nodes and 15% of malicious nodes respectively. This pictorial view contains total 150 nodes, out of which black colored are benevolent nodes and red colored are malicious nodes. **Figure 8**, shows, the percentage of packet loss in the network versus percentage of malicious nodes. The packet loss values in **Figure 8**, are the average of ten values taken for different deployments. **Figures 9(a)** and **Figure 9(b)** shows, malicious node detection by TENCR in a network for 20% and 30% of malicious nodes respectively for different weights given to direct trust (geometric mean) and indirect trust (arithmetic mean). We can observe the increment in percentage of malicious nodes detection as weight to the geometric mean increases from 0.5 to 0.75; Hence, the application of geometric mean instead of arithmetic mean in the calculation of direct and indirect trust evaluation, captures the malicious nodes efficiently in WSNs.

4.3. Calculation of Direct and Indirect Trusts, and Identification of Trustworthy Node

We have implemented our trust evaluation technique practically, using nesC programming language and TinyOS which is very suitable operating system for wireless sensor network nodes. We have used four nodes, *iris motes* whose IDs are 0, 1, 2 and 3. Every node in this case will be having three neighbouring nodes and can supply indirect information of one neighbouring node to rest two neighbouring nodes vice versa.

Every node evaluates direct trust on all neighbours from trust metrics those are evolved with direct interactions and stored in its database. Before calculating the trust, it brings the indirect information from neighbours, then it evaluates the trust. **Figure 10**, shows snapshot of a manual calculation of direct trust, indirect trust and the total trust by all four nodes on their neighbouring nodes. Here, we have assumed only two trust metrics and their initial values are shown in brown colored columns and manually calculated direct, indirect and total trust are also shown in **Figure 10**.



(a) (b)

Figure 7. WSN random deployment without and with malicious nodes. (a) WSN with 30% malicious nodes; (b) WSN with 15% malicious nodes.

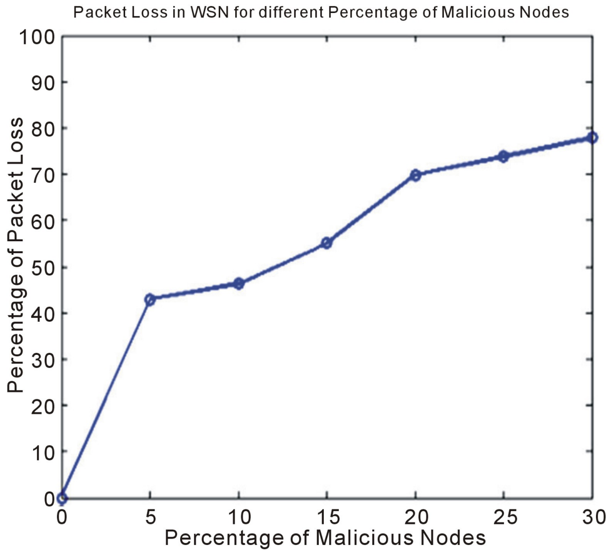
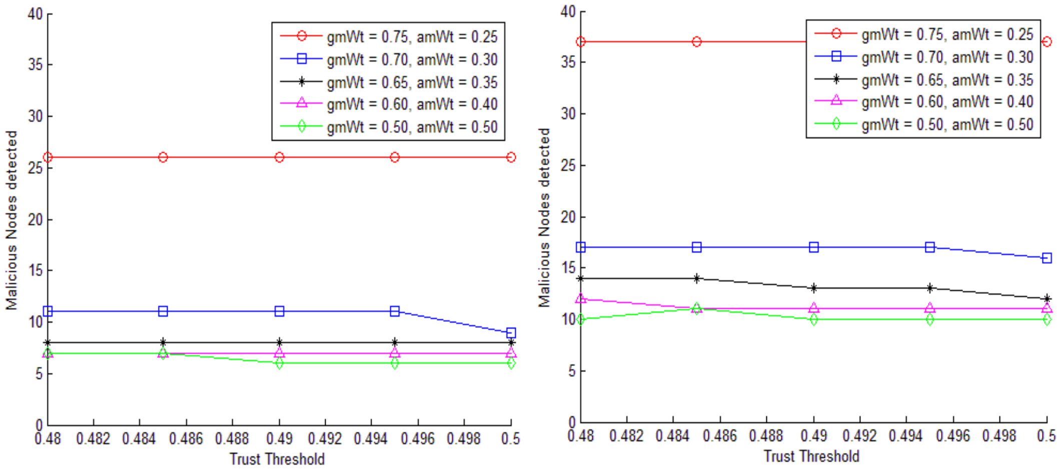


Figure 8. Packet loss by malicious nodes.



(a)

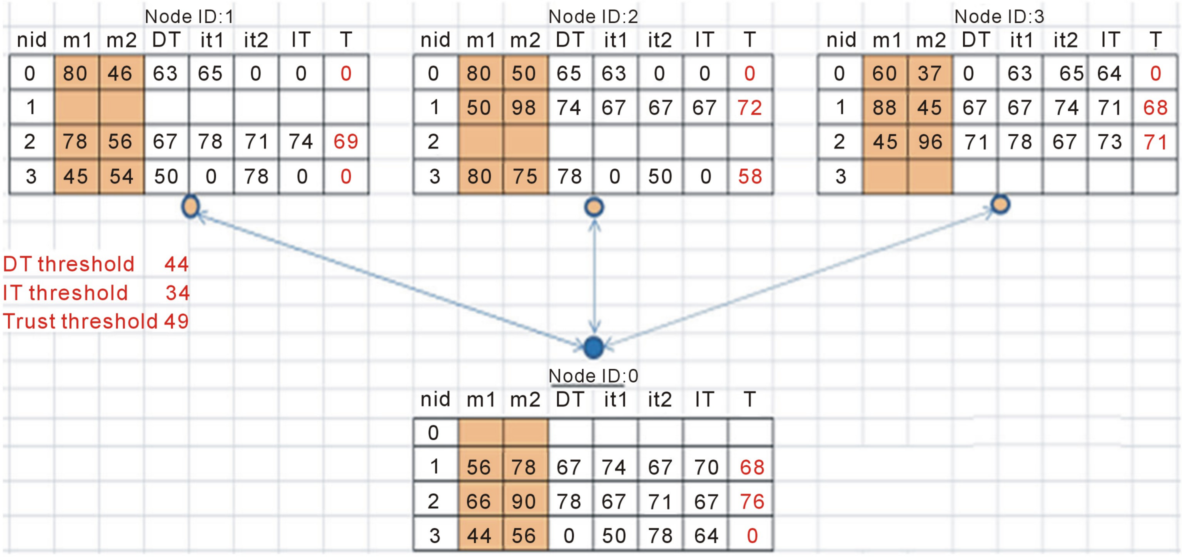
(b)

Figure 9. Detection of malicious nodes for (a) 20% and (b) 30% of malicious nodes.

Figure 11, shows the practical implementation results experimented with *iris motes*. Every node is displaying the node ID whose trust is large among their neighbours. Node is having three LEDs, yellow, green and red (LSB).

5. Conclusion

The TENCR trust evaluation method presented in this paper detects the malicious and/or selfish nodes in the network if any, and provides the trustworthiness among the node and its neighbours based on the trust metrics evolved by previous transactions in the network and the recommendations provided by the neighbouring nodes. The direct trust is formed by trust metrics (QoS characteristics) and indirect trust is formed by neighbour nodes recommendations. In this paper, we argue that few basic and fundamental functional QoS characteristics can be categorized as higher priority and few (others) are lower priority, and the categorization of trust metrics is the application specific. Once the classification of the trust metrics, most trusted neighbour nodes, and their threshold levels are fixed by the application, our proposed model see that they are maintained at the node in finding the trust on any neighbour node. We have shown graphically as well as in tabular form how malicious/selfish node can be captured by TENCR Trust evaluation method for different cases. We have also shown in algorithm



Node ID:1

nid	m1	m2	DT	it1	it2	IT	T
0	80	46	63	65	0	0	0
1							
2	78	56	67	78	71	74	69
3	45	54	50	0	78	0	0

Node ID:2

nid	m1	m2	DT	it1	it2	IT	T
0	80	50	65	63	0	0	0
1	50	98	74	67	67	67	72
2							
3	80	75	78	0	50	0	58

Node ID:3

nid	m1	m2	DT	it1	it2	IT	T
0	60	37	0	63	65	64	0
1	88	45	67	67	74	71	68
2	45	96	71	78	67	73	71
3							

DT threshold 44
IT threshold 34
Trust threshold 49

Node ID:0

nid	m1	m2	DT	it1	it2	IT	T
0							
1	56	78	67	74	67	70	68
2	66	90	78	67	71	67	76
3	44	56	0	50	78	64	0

Figure 10. Manual calculation of direct, indirect and total trusts of every neighbours.

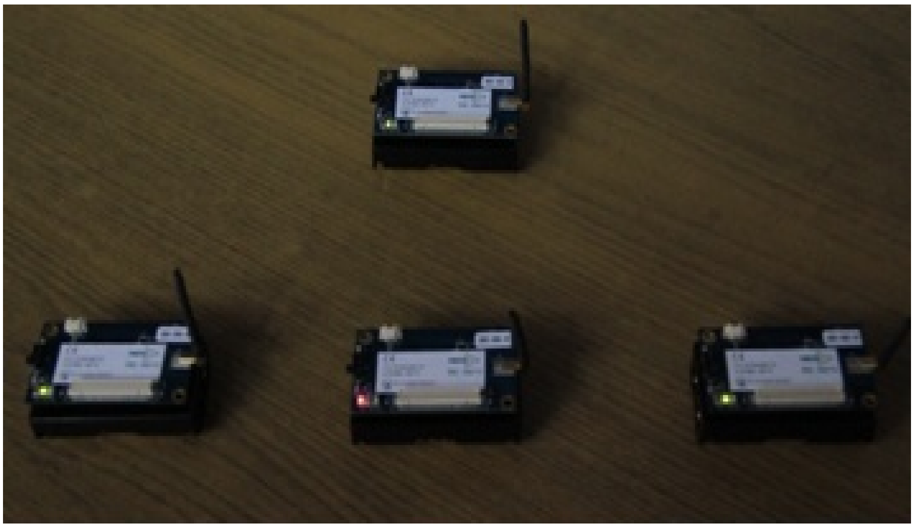


Figure 11. Implementation results for identifying trustworthy neighbouring node.

form, the behaviour of node when no trustworthy node found in its vicinity, *i.e.* adaption of energy consuming operation by increasing its radio range to find trusted node by interacting with the neighbours. In future, we want to concentrate on designing the trust management system for WSN by considering the dynamic change in a) node's location; b) neighbouring nodes; c) issues related to revocation of nodes, and also trust evaluation and malicious node detection at node level. We also have a plan to develop an algorithm for energy efficient dynamic routing protocol, which elects an indispensable node for routing based on node trust, energy level and the distance towards the sink.

References

[1] Akyildiz, I.F., Su, W., Sankarasubramaniam, Y. and Cayirci, E. (2002) Wireless Sensor Networks: A Survey. *Computer Networks*, **38**, 393-422. http://dx.doi.org/10.1016/S1389-1286(01)00302-4

[2] Culler, D., Estrin, D. and Srivastava M. (2004) Overview of Sensor Networks. IEEE Computer Society, August 2004.

[3] Chintalapudi, K., Fu, T., Paek, J., Kothari, N., Rangwala, S., Caffrey, J., Govindan, R., Johnson, E. and Masri, S. (2006) Monotoring Civil Structures with a Wireless Sensor Network. *IEEE Internet Computing*, **10**, 26-34.

http://dx.doi.org/10.1109/MIC.2006.38

[4] Mainwaring, *et al*. (2002) Wireless Sensor Networks for Habitat Monotoring. International Workshop on Wireless Sensor Networks and Applications (ACM). http://dx.doi.org/10.1145/570738.570751

[5] Karlof, C. and Wagner, D. (2003) Secure Routing in Sensor Networks: Attacks and Countermeasures. First IEEE International Workshop on Sensor Network Protocols and Applications.

[6] Perrig, A., Zewczyk, R., Wen, V., Culler, D. and Tygar, D. (2002) SPINS: Secirity Protocols for Sensor Networks. *Wireless Networks*, **8**, 521-534. http://dx.doi.org/10.1023/A:1016598314198

[7] Shaikh, R.A., Lee, S., Khan, M.A.U. and Song, Y.J. (2006) LSec: Lightweight Security Protocol for Distributed Wireless Sensor Network. Proceedings of 11th IFIP International Conference on Personal Wireless Communication (PWC'06), Spain, September 2006.

[8] Zhang, Y., Liu, W., Lou, W. and Fang, Y. (2006) Location-Based Compromise Tolerant Security Mechanisms for Wireless Sensor Networks. *IEEE Journal on Selected Areas in Communications*, **24**, 247-260. http://dx.doi.org/10.1109/JSAC.2005.861382

[9] Karlof, C., Sastry, N. and Wagner, D. (2004) TinySec: A Link Layer Security Architecture for Wireless Sensor Networks. Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, Baltimore, 162-175. http://dx.doi.org/10.1145/1031495.1031515

[10] Zahariadis, T., Leigou, H.C., Trakadas, P. and Voliotis, S. (2010) Mobile Networks: Trust Management in Wireless Sensor Networks. *European Transactions on Telecommunications*, **21**, 386-395.

[11] Sun, Y., Han, Z. and Liu, K.J.R. (2008) Defense of Trust Management Vulnerabilities in Distributed Networks. *IEEE Communications Magazine*, **46**, 112-119.

[12] Saraogi, M. (2005) Security in Wireless Sensor Networks. Department of Computer Science, University of Tennessee, Knoxville.

[13] Lopez, J., Roman, R., Agudo, I. and Fernandez-Gago, C. (2010) Trust Management Systems for Wireless Sensor Networks: Best Practices. *Computer Communications*, **33**, 1086-1093. http://dx.doi.org/10.1016/j.comcom.2010.02.006

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

[15] Babu, S.S., Raha, A. and Naskar, M.K. (2011) A Direct Trust Dependent Link State Routing Protocol Using Route Trusts for WSNs (DTLSRP). *Wireless Sensor Network*, **3**, 125-134. http://dx.doi.org/10.4236/wsn.2011.34015

[16] Trakadas, P., Maniatis, S., Zahariadis, T., Leigou, H.C. and Voliotis, S. (2009) A Novel Flexible Trust Management System for Heterogeneous Sensor Networks. *International Symposium on Autonomous Decentralized Systems*, *ISADS* 2009, Athens, 23-25 March 2009, 369-374.

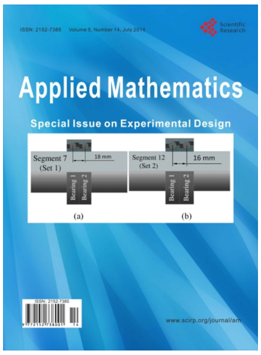
[17] Liu, K., Abu-Ghazaleh, N. and Kang, K.D. (2007) Location Verification and Trust Management for Resilient Geographic Routing. *Journal of Parallel and Distributed Computing*, **67**, 215-228. http://dx.doi.org/10.1016/j.jpdc.2006.08.001

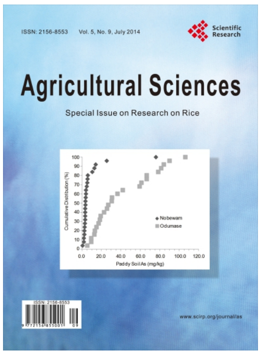
[18] Kim, T.K. and Seo, H.S. (2008) A Trust Model Using Fuzzy Logic in Wireless Sensor Network. *Proceedings of World Academy of Science*, *Engineering and Technology*, **44**, 69.

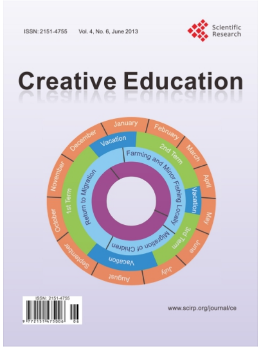
Advances in
Bioscience and Biotechnology

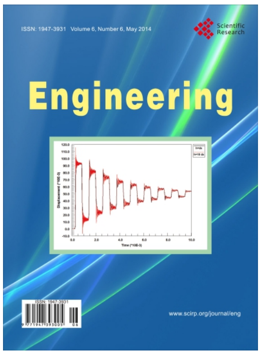
American Journal of
Analytical Chemistry

American Journal of
Plant Sciences
Special Issue on Photosynthesis

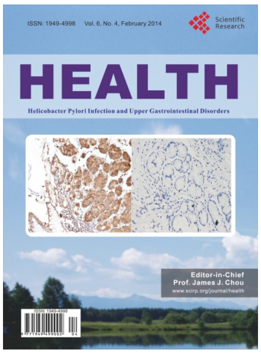

Applied Mathematics
Special Issue on Experimental Design


Agricultural Sciences
Special Issue on Research on Rice


Creative Education

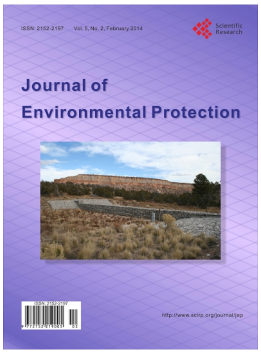

Engineering

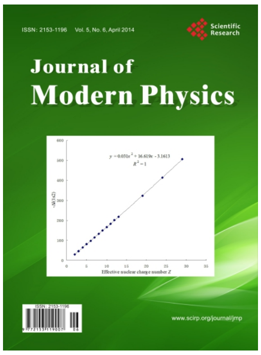
Food and Nutrition Sciences


HEALTH
Helicobacter Pylori Infection and Upper Gastrointestinal Disorders
Editor-in-Chief
Prof. James J. Chou

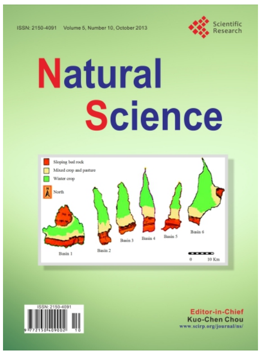
Journal of Computer
and Communications

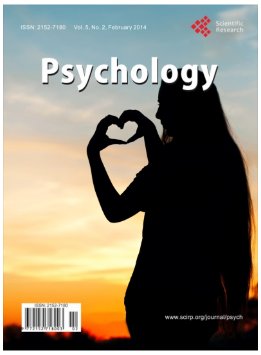
Journal of Cancer Therapy
Special Issue on Bone Tumor Research


Journal of
Environmental Protection


Journal of
Modern Physics

Modern Economy


Natural
Science
Editor-in-Chief
Kuo-Chen Chou


Psychology