

DEESR: Dynamic Energy Efficient and Secure Routing Protocol for Wireless Sensor Networks in Urban Environments

Mohammad S. Obaidat*, Sanjay K. Dhurandher**, Deepank Gupta**, Nidhi Gupta** and Anupriya Asthana**

Abstract—The interconnection of mobile devices in urban environments can open up a lot of vistas for collaboration and content-based services. This will require setting up of a network in an urban environment which not only provides the necessary services to the user but also ensures that the network is secure and energy efficient. In this paper, we propose a secure, energy efficient dynamic routing protocol for heterogeneous wireless sensor networks in urban environments. A decision is made by every node based on various parameters like longevity, distance, battery power which measure the node and link quality to decide the next hop in the route. This ensures that the total load is distributed evenly while conserving the energy of battery-constrained nodes. The protocol also maintains a trusted population for each node through Dynamic Trust Factor (DTF) which ensures secure communication in the environment by gradually isolating the malicious nodes. The results obtained show that the proposed protocol when compared with another energy efficient protocol (MMBCR) and a widely accepted protocol (DSR) gives far better results in terms of energy efficiency. Similarly, it also outdoes a secure protocol (QDV) when it comes to detecting malicious nodes in the network.

Keywords—Sensor Network, Security, Energy Efficiency, Routing, Dynamic Trust Factor

1. INTRODUCTION

Mobile devices with communication facilities are omnipresent in an urban environment. All these devices can come together to set up a network to provide interconnectivity and useful services. The mobile devices can be used as nodes in an ad hoc network which could be put to use beneficially by exploiting the processing power of these devices. An ad hoc network is heterogeneous network that consists of mobile phones, laptops and other handheld devices. As the network is ad hoc it has minimal infrastructure dependencies. But, being a part of any wireless network means putting the security of your device at risk. Moreover, an ad-hoc network requires participating nodes to forward packets meant for other nodes in the network which can consid-

※ A preliminary version of this work has been accepted in the *IEEE GLOBECOM 2010 Workshop on Web and Pervasive Security*, Miami, Florida, USA, December 6-10, 2010.

Manuscript received September 5, 2010; accepted September 6, 2010.

Corresponding Author: Mohammad S. Obaidat

* Department of Computer Science & Software Engineering, Monmouth University, NJ, USA (obaidat@monmouth.edu)

** CAITFS, Division of Information Technology, Netaji Subhas Institute of Technology, University of Delhi, New Delhi, India. (dhurandher@rediffmail.com, deepank.gupta@gmail.com, coolnidhi13@gmail.com, anupriyasthana@gmail.com)

erably drain the batteries of some nodes in the network. Thus, the protocol should be secure and energy efficient with support for heterogeneous nodes in the network.

This network will make mobile devices act as a natural extension of a user and provide relevant and context-specific information from multiple sensor nodes in the environment. The various kinds of computing devices such as laptops, palmtops, mobile phones, notebooks etc. come together along with multiple types of position sensors, heat sensors, temperature and moisture sensors, and security sensors to form a network with every node able to communicate to another. The network is scalable with the ability to add hundreds or thousands of nodes without any intervention from the network administrator. Apart from this, the network respects the security and privacy concerns of a user and makes use of only a minimal amount of resources of the participating nodes.

The network infrastructure consists of three types of nodes besides the server, namely cluster heads, sensor nodes and consumer nodes. The server is used for storing information as a database. The job of the server is to reply to information queries by clusterheads. Clusterheads are special nodes as they have the advantage of constant power supply, unlike sensor nodes and consumer nodes which are battery constrained. So, clusterheads are used for processing application layer services to exploit their computing power. Sensor nodes are planted in the urban environment which perform specialized tasks (e.g. finding out the location of a consumer node) and send the information to clusterheads. All the components of the network discussed till now are installed by the entity in charge of the urban environment. In case of the DEESR protocol, the urban environment taken under consideration is a shopping mall. So in this case, the server, the clusterheads and the sensor nodes will be installed by the owner of the mall. The mobile devices, which are owned by the customers, enter the mall and get included into the network. As the protocol respects user privacy and preference, the consumer node user may choose not to be included into the network.

2. PREVIOUS WORKS

Most of the routing algorithms can be classified as either proactive or reactive routing algorithms [1]. Proactive routing algorithms like DSDV [1], and STAR [2], are based on exchanging the knowledge about the network topology periodically. Thus, while these algorithms provide route to the destination instantly, they also require more bandwidth and may drain out the battery power of the nodes very quickly [3]. On the other hand, reactive routing protocols like AODV [4], DSR [5] and TORA [6] do not initiate a route discovery unless a route is required. These protocols do not provide security to the participating nodes. Even the energy conservation for battery powered devices needs to be implemented to make the network energy efficient. But the underlying concept behind a reactive protocol can be used in our design.

AODV is a sequence number, routing table based protocol which keeps only a single route for each source-destination couple in its routing table. DEESR protocol also uses sequence number and has a routing table, but is less bulky than the AODV. Firstly, DEESR protocol maintains a routing table only for its own population rather than for the nodes of the entire network. Another distinguishing factor is that it looks for more than one route for a specific source-destination couple, thus helping us in deciding to use the best route rather than the first one which is found, which may not always be the most efficient one. DSR, another reactive protocol, does not use

sequence numbers and encapsulates the routing information in the data or control packet only rather than storing it at the node. DSR also has a routing cache of size 1 which helps in keeping a record the latest path used. This routing cache is not sufficient, since in our case, we need to find the paths using the next-hops to find the optimal route and not the latest used. If we were to use the routing cache of size 1, we would have to find a route every time the destination changes, which would not be energy efficient.

There is also a third category of routing protocols known as cluster-based protocols. These protocols have also been studied in detail. The reference [7] discusses that clustering or hierarchical network topology can aid in energy conservation. Cluster based routing protocols such as [8] and [9] also help in making the network more scalable. In an urban environment like a shopping mall, a network will need to scale according to the number of people which may differ greatly in the span of a single day. But, the cluster based protocols have inherent overhead in case of dynamic networks when the nodes are mobile, and cluster head election and cluster maintenance can prove to be a costly affair. Also there are no guarantees that an elected cluster head might itself turn out to be malicious.

Routing protocols have also been proposed [10] which base their routing decisions on battery reserves and energy cost to transmit to the next hop. Other protocols like those referred to in [11] and [12] have also been proposed which base their routing decisions on power optimization to make the network energy efficient. Specifically we will compare DEESR protocol's performance against the Min-Max Battery Cost Routing algorithm [12]. This protocol first calculates the battery cost for each route using the battery cost function and then chooses the route with the maximum battery remaining. The MMBCR tries to avoid nodes with low battery, and with this, the hosts will be used more fairly. But, there is no guarantee that the minimum total transmission power paths will be selected, because the consumption of power can increase by the use of a different path. DEESR protocol incorporates hierarchy and routing decisions based on power optimization to achieve the goal of making the network energy efficient. Apart from this, we make use of special nodes with constant power supply, if present, to make the network more energy efficient.

As discussed earlier, ensuring security is critical in any urban ad-hoc network. There are various security based protocols [13-17] for ad hoc networks. Yi et al. [13] have discussed in their paper that if the routing protocol is compromised by changing the messages in the transit, then no security at higher layers can help. To address this problem they have proposed *Security Aware Ad-hoc Routing (SAR)*. It makes sure that data is routed through a secure route composed of trusted nodes and the security of the information in the routing protocol. Apart from this, security has been implemented at link layer in TinySec protocol given in [17].

There are two ways of ensuring security: either by encrypting the data or by identification and removal of malicious node from the network. The first approach using public-key algorithms such as the Diffie-Hellman [14] is not suitable for a wireless sensor network as it has limited resources. Using such an algorithm will mean a large amount of computation power being used to encrypt and decrypt every message making the network slow. However, other security-based protocols like SPINS [15] have been proposed keeping in mind the resource limitations. This protocol encrypts a message differently each time. SPINS comprises of two building blocks, SNEP and μ TESLA. The μ TESLA protocol is based on key chain generation, in which a key is generated from the next key, in addition to the application of a function and the last key is generated once in a time-interval. But the encryption approach still requires storage of public and

private keys. Also, the transmission of digital signatures consumes more energy. We follow the second approach in which malicious nodes are flagged and systematically removed from the network by observing the characteristics/behavior of a node in the system.

The SNEP protocol is efficient for stationary cluster based networks which have a central key assigning authority. A base station serves as a gateway to the outside world through which a node connects. All the nodes have constraint on battery and other resources. In case of a compromise of the trusted base station, the security of the whole network collapses. The communication between the node and the base station is through the keys which are specific to all the nodes. This guarantees integrity and authentication but does not prevent the malicious node from overhearing or disguising as the base station to gain access to the sensitive data though preventing active attacks.

The proposed DEESR approach is compared against the QDV Based Routing approach followed in [16] and we define a dynamic Trust Factor (DTF) for each node in the system. This protocol is used for securing the wireless sensor networks and is based on Ant colony optimization (ACO). It uses quality-of-service (QoS) and reputation to find out the trust of the node. Thus, by monitoring these two parameters the protocol is able to detect and disable the malicious nodes from gaining access and participating in the network.

The distinguishing factors about DEESR protocol is the inclusion of factors like Next-Hop Determination Factor (NHDF), node population and dynamic trust factor which help in keeping the protocol secure and energy efficient. The protocol also takes the heterogeneity of the consumer nodes into consideration while setting up the network or finding routes by giving preference to consumer nodes with properties better suited for the purpose of the protocol. We take into consideration the longevity of the mobile devices which refers to the time for which the node has been in the network in deciding how secure using that node would be.

3. MOTIVATION

While, this paper discusses a routing protocol for urban environments, the long term goal of our research is to deploy service-oriented network architecture in urban environment which provides unique services to the user. Let us consider an example scenario which illustrates the long term vision behind our research. Supposed a user enters a shopping mall. As he/she enters the mall, the mobile device of the user is immediately detected by the clusterheads located within the mall and is included into the network upon the consent of the user. Once the communication channel has been set up, the user may choose to inquire about what he/she wants to shop for or about any other query related to the mall and gets a suitable reply in the appropriate format. Now suppose there is a fire in the section of the mall the user is located in. The mobile user gets an alert on his mobile phone informing him about the same. Since this may be the first visit of the mobile user to the mall, he may require directions to get out from the nearest fire exit. So when the clusterhead alerts the user about the fire, it also gives the user the directions to the nearest fire exit. This is achieved by using sensor nodes placed all over the shopping mall to find the consumer node's coordinates and accordingly locate it on the map and finally provide directions to the mobile user to swiftly and safely exit the mall. The sensor nodes are used to provide special functions, like location detection in this case, upon being asked by the clusterheads, and subsequently reply to the query of the clusterhead with the desired information.

The example above illustrates the location based routing service, which is just one of the many applications of an ad hoc network such as the one proposed in this paper. There are many other possibilities which would make the urban environment truly responsive to the needs of the costumer. This level of responsiveness and interactivity between the user and the urban environment take the customer experience to a new level. Some of the other applications facilitated by such a network layer protocol could be advertisements and offers based on the proximity to a shop, display of user-selected content on public displays, asking a user to switch off the phone in a silent zone (e.g. movie hall), enabling a user to change the AC temperature or switch it off in a certain area and providing security messages in case the user goes in a restricted area. These applications and many more can be implemented at the application layer.

4. OUR PROPOSED WORK


4.1 Specifications

The protocol is Reactive/On-Demand to avoid unnecessary overheads of periodically flooding the network. The protocol not only works in case of heterogeneous consumer nodes e.g. cellular mobile phones, laptops, other mobile devices but also adapts to burden the more powerful nodes like laptops while lessening the burden on mobile phones. The network shall be interconnected via Wi-Fi.

4.2 Assumptions

The incoming mobiles already have the required software installed in their devices for the protocol to work. Thus, the installation of the software onto the user’s mobile will be outside the purview of the proposed protocol. The coverage of the urban environment will be 100% through the ad hoc network system. Thus, there is no area/space in the mall which cannot be communicated with and no area is inaccessible in the mall including corners. The database in the server is authentic and fully updated. The sensor nodes and cluster heads cannot be hacked and are considered as the trusted nodes in the network. Lastly, the sensor nodes and the cluster heads are placed in a way such that even without the presence of the intermediate consumer node, a packet can be transmitted back to a clusterhead by the consumer node. The administrator of the urban environment has installed the clusterheads and the sensor nodes. These nodes have already

Table 1. Summary of Table Components



Properties	Server	Cluster Head	Sensors	Mobile Nodes
Power	Unlimited	Unlimited	Limited	Limited
Transmission Range	Large	Large	Small	Small
Mobility	Static	Semi-static	Semi-static	Mobile
Reputation	Trusted	Trusted	Trusted	Non-Trusted
Number	One	Fixed	Scalable	Variable

formed an ad hoc network based on DEESR protocol before any consumer node enters the network.

4.3 Server

The server has a constant power supply. It will communicate wirelessly with all the clusterheads. The position of the server is static and centrally located with respect to the clusterheads so that it can be reached by the maximum number of clusterheads making inter-communication faster and efficient. Only one server exists in the entire network system.

The main purpose of the server is to store data, thus, it shall act as a repository of all the mall specific information. This data will be accessed by the clusterheads as and when required for processing and servicing location based and content based user requests. The server will provide the clusterheads with all the information that they require to carry out their operations.

4.4 Clusterhead

Clusterheads shall be identified by a system-specific nomenclature. A clusterhead has a constant power supply and larger transmission range as compared to sensors and consumer nodes. It is a trusted node with its dynamic trust factor as 1. The position of the clusterhead is semi-static, meaning that even though its position does not change frequently, it might move from one place to another occasionally. For instance, a stall is placed in a certain area of the mall on a contractual basis for 15 days. The stall owner installs a clusterhead in the stall and it remains stationary for 15 days. But after, the contracted period is over, the clusterhead might move to another location. The location of each clusterhead will be known to every other clusterhead and to the sensor nodes as well. The population size of the clusterhead will be large as compared to the consumer nodes and sensor devices with all the neighboring sensor nodes as part of its population initially. The clusterhead will keep the location information of the sensor nodes. The number of clusterheads shall remain constant depending on the size of the mall.

The clusterheads will be responsible for detecting new incoming nodes and starting dialogue with them. They will be responsible for querying the sensor nodes for dynamic information specific to a consumer node or its environment such as its location and the environment around a node in response to queries made by a consumer node. They will also access static information about the urban environment from the server for content-based services. The clusterhead also maintains a log of the communication between itself and the other nodes (sensor nodes and consumer nodes) for a single session where the log contains the last packet sent to and the last reply received from a particular consumer node.

4.5 Sensor nodes

The sensor nodes shall be identified according to the system specific nomenclature decided upon. Unlike the previous components, the sensor nodes will be battery operated, having a maximum energy constraint and hence a maximum transmission range. It will be within the range of a clusterhead or another sensor for the purpose of routing. There are many kinds of sensors used for different purposes. The dynamic trust factor of the sensor nodes is also 1 as they are trusted. The number of sensor nodes shall remain constant as it depends on the size of the mall. The sensor nodes will know the location of the nearest clusterheads. Sensor nodes shall

also be semi-static in position. The sensor nodes shall reply to the requests by the clusterhead by sensing specific information related to the sensor. The sensor nodes can also act as routers between the consumer nodes and the clusterheads.

The sensor nodes shall reply to the requests by the clusterhead for sending mobile node specific information. The sensor nodes will be communicating with the clusterheads using multiple hops, which may be other sensor nodes or mobile nodes. The sensor nodes shall take part in the routing protocol between the mobile nodes and the clusterheads. Special sensor nodes shall detect other information like temperature, etc. and reply to the query of the clusterhead.

4.6 Consumer nodes

The consumer nodes shall be identified with their IMEI (International Mobile Equipment Identity) [18] number or MAC Address. These nodes will be battery operated and thus will be energy constrained. These nodes will be mobile, i.e. their position will keep changing at frequent intervals. The number of consumer nodes is variable and the system should scale accordingly. The consumer nodes, like the sensor nodes, will have very limited transmission range. Consumer nodes will be heterogeneous as they can range from different models of mobile phones to laptops with different configurations. When a consumer node enters the shopping mall for the first time, its dynamic trust factor will be initialized to 0. As time passes on, the trust of a consumer node will be increased or decreased according to its behavior.

Consumer nodes shall be used for routing purposes for forwarding the messages when required. They shall also be used for initializing requests and reply packets according to the service required by the user. They may also be used to store user preferences, so that the user doesn't have to detail them to the system again the next time he/she enters the mall.

4.7 Terms and Notations

The protocol has been designed so that an ad hoc network can be set up for mobile devices securely and in an energy efficient manner. To facilitate this, certain parameters have been focused upon. These parameters are then used in a weighted manner to come to a routing decision. The following subsections present the terms and parameters used in the protocol as it is necessary to understand them before proceeding.

4.7.1 Population

Every node in the network has a set of nodes which are in its transmission range and with which it decides it can communicate securely. This set of nodes with which a node communicates is called the population of that node. Only the nodes satisfying a certain criteria are included into the population of a node. In this protocol, as security and energy efficiency are of prime importance, the parameter used for narrowing down a node's population is the Dynamic Trust Factor (DTF).

To limit the database requirements and to minimize network traffic for population maintenance, a node maintains a population restricted up to a maximum population size. The population size is dependent on the type of the node, e.g. a clusterhead would have a population size greater than that for the sensor nodes and consumer nodes and this protocol tries to find this optimal population size.

Dynamic Trust Factor (DTF):

This is given by the ratio of the sum of the packets received and generated subtracted by the number of packets transmitted to the sum of the packets received and generated. Thus, DTF gives the ratio of the number of packets dropped to the total number of packets received and generated. In DEESR protocol, a particular node is trusted if its packet dropping ratio is low. Thus, trust is inversely proportional to the number of packets dropped. We do not consider or handle the situation where the packet is tampered with at the next-hop before retransmission. This value is also extracted from the routing table.

$$\delta = (p_r + p_g - p_d)/(p_r + p_g) \quad (1)$$

Where, δ denotes the DTF

p_r = Number of packets received by that node

p_g = Number of packets generated by that node

p_d = Number of packets dropped by that node

Equation 1 represents how the DTF (δ) is calculated where δ is given by the ratio of the packets transmitted by a node to the total number of packets received and generated at that particular node. Subtracting the number of packets dropped (p_d) from the sum of the number of packets received (p_r) and packets generated (p_g) results in the number of packets transmitted. Similarly the sum of p_r and p_g results in the total number of packets at a particular node.

At the source and the destination nodes, the packets for which the node is the source/destination are not counted in the DTF calculation, which provides security against active attack, specifically, malicious packet injection.

4.7.2 Next-Hop Determination Factor

Every node maintains a routing table for the nodes in its population, which it would be using as a next-hop for a particular destination. Once the population has been identified and narrowed down, the transmission starts. The source node or the intermediate node searches its routing table for the next-hop for the desired destination. Upon searching the destination, if there is only one node which is a part of a route to that destination, the packet is forwarded to that member of the population.

In case there are two or more nodes which can be treated as next-hop for a specific destination, the protocol uses a criterion called the Next-Hop Determination Factor (NHDF) to decide which node would be more suitable. The NHDF is the weighted sum of the link quality and the node quality.

A. Link Quality (LQ)

The link quality is defined as the property of the connection between two communicating nodes. The preference of one link over another is decided upon by the following parameters:

- *Link Usage Factor (LUF)*: It is defined as the ratio of the number of times a link has already been used in the past to the maximum possible number of times it can be used. This factor gives an indication of the reliability of the link. This value is extracted from the routing table entry for that node. Since this is a desired quality, it is directly proportional to the

link quality.

- *Hop Count (HC)*: It is the number of hops required to reach the destination from the next-hop entry in the routing table. This value is extracted from the routing table as well. The greater the hop count, the greater the number of resources, in the form of consumer nodes or sensor nodes, is required and thus greater energy is being used. Since we want energy efficiency, the node quality will be inversely proportional to the hop count.
- *Distance Factor (DF)*: It is given by the ratio of the distance between the two nodes, namely the present node and the next-hop node, to the maximum possible distance between any two nodes in the network. This value is extracted from the request/reply packet received by the current node. Since this value is desired to be the minimum, it is inversely proportional to the link quality.
- *Link Normalization Factor (LNF)*: Experimentation has shown that the value which the link quality gets comes out to be greater than 1 in many situations, which is undesirable as this unavoidably gives more weight-age to link quality. To avoid this, there is a need for a Link Normalization factor. This LNF is a constant which needs to be calculated for each given physical set up, i.e. there will be only one constant for each urban environment. Since we need to make the value of the Link Quality less than 1, the LNF is the maximum value that the link quality can take for a given urban environment.

$$\alpha = 1/K * \gamma / (\rho * \sigma) \quad (2)$$

Equation 2 represents how the link quality is calculated. Link quality (α) is given by ratio of link usage factor (γ) to the product of hop count (ρ) and distance factor (σ). This ratio is then divided by the link normalization factor (K) which normalizes α to have values ranging from 0 to 1.

B. Node Quality (NQ)

Node quality is the intrinsic property of the next-hop node. The parameters by which it is decided are as follows.

- *Battery Power Factor (BPF)*: It is defined as the percentage of battery power remaining at the next-hop node. If the value of battery power factor for a node is above a certain minimum level, it may be used as a next-hop, else not. The greater the battery power factor, the more suitable is the node for usage as a next-hop. Thus, battery power is proportional to node quality.
- *Longevity Factor*: Longevity factor is defined as the ratio of the time for which the next-hop has remained in the network to the total time a node is expected to stay in the network. Longevity gives us an indication of how well the node has been providing services while it has remained in the network. Since this is a desirable quality in a next-hop, this quantity is directly proportional to the node quality.

$$\beta = \tau * \omega \quad (3)$$

Equation 3 represents how node quality (β) is calculated. β is given by the product of battery power factor (τ) and longevity factor (ω).

So, finally we get the formula for NHDF as:

$$NHDF = w_1 \alpha + w_2 \beta \quad (4)$$

Where, w_1 and w_2 are the weights assigned to α and β . In equation 4, the values of the weights w_1 and w_2 can vary from 0.0 to 1.0 such that their sum remains equal to 1.0 at all times. These weights can be adjusted according to the needs of the hour. If we want that the connection quality be more important than the security, we can keep w_1 more than w_2 . In contrast, if we want our network to be more secure, we can increase w_2 . Similarly, if we don't want to consider the node quality (NQ), we can make w_2 equal to 0 and w_1 equal to 1.

5. PROTOCOL FUNCTION

In this section, the functioning of the proposed protocol has been explained. There are four subsections in this section which explain the initiation, population maintenance, updating of next-hops and data packet transmission processes, respectively. The initiation population, maintenance and updating of next-hops are sub-routines of the data packet transmission protocol.

5.1 Dynamic Route discovery

Let us consider a scenario in which a node A acts as a source and requires to send data to the destination node E . As shown by arrows in the Figure 1, A broadcasts the route-request packet. It contains the following information: link usage factor, longevity, battery power, reputation, coordinates and hop type of node A along with the hop count till now. This packet is received by, say, node B which updates the information about A in its routing table and replaces that information in the route-request packet by its own information before broadcasting the packet again. Similarly, this goes on till the packet reaches the destination node E with the information of the previous node, say, node D while the routing tables for all the intermediate nodes get updated. In Figure 1, the dashed arrows show the route-request packets which actually reached the destination.

Since E is the destination node, it initiates a route-reply packet with this information: link usage factor, longevity, battery power, reputation, coordinates and hop type of node E along with the hop count till now and sends it to all the previous nodes from which it has received the request since there maybe more than one route between the source and destination. The first hop in the reply route updates the information of the destination node and again replaces that infor-

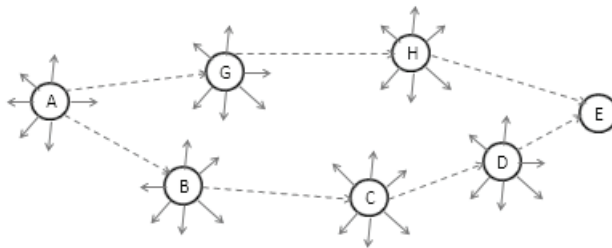


Fig. 1. Route-request packets being broadcasted

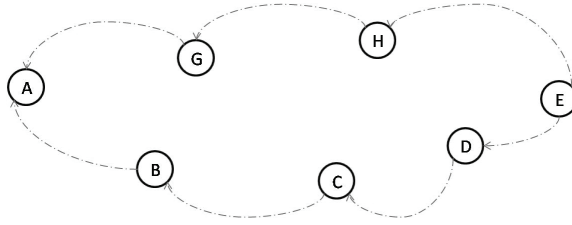


Fig. 2. Route-reply packets being uni-cast back to the source via the routes discovered

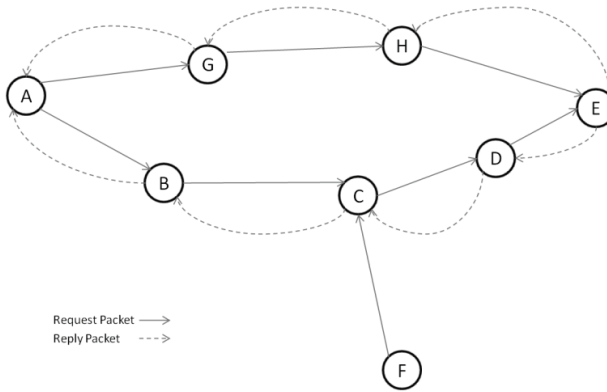


Fig. 3. Illustration of how a malicious node F is detected in the system

mation with its own information and unicasts the packet back to previous nodes in the route. When the reply packets finally reach the source node, it makes an entry in its routing table for the routes discovered and chooses the best next hop based on the NHDF. This route-reply procedure is illustrated in Figure 2.

In Figure 3, let us say a malicious node *F* injects packets into the network disguising itself as a trustworthy node, say, node *A*. Since *F* is not the actual source, so the disguised packet will be tracked down and will be counted in calculating the DTF which will indicate its untrustworthiness. Hence, DEESR protocol will be able to isolate such a node from the network.

5.2 Population Maintenance

When a node needs to send a message to any other node for the first time, it broadcasts hello packets and discovers routes to a particular destination. These discovered routes are entered into the routing table. A typical routing table entry has the following entries about the next-hop node: battery power, longevity, distance, hop type, reputation, link usage factor, packets generated, packets received, packets forwarded and hop count from the node to the destination. These next-hop nodes in the routing table thus form the population of the node. As the node communicates to send more packets over time, it discovers more routes and makes entries for more next-hop nodes making the size of the routing table increase. Thus, the number of members of its population keeps increasing with time.

The number of nodes in the population of a particular node is not equal to the number of routing table entries, but equal to the number of distinct next-hops in the routing table. This is because, there might be a single node which is the next-hop for two destinations, which results in

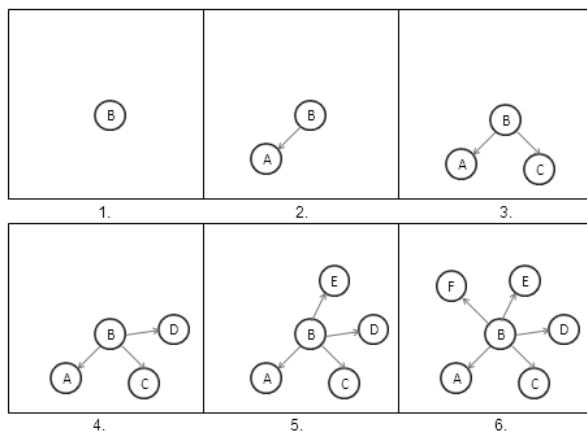


Fig. 4. Illustration on how other nodes get included into population of node B till Population Size is reached

two routing table entries but only one distinct node in the population. At some point, the members in the population may increase beyond the population size. At this juncture, the node has to eliminate certain nodes from its population which it does by evaluating the DTF of the nodes present in the routing table and keeping the nodes with the maximum DTF out of the nodes present.

Let us illustrate this with an example where the Population size is 5. In Figure 2, frame 1 shows the scenario when node B enters into the network and has no nodes in its population. Slowly, as it needs to communicate more, it starts finding more nodes as next-hops to include in its population. This is depicted in frames 2,3,4,5 and 6 of Figure 2. The node B does not check for any parameter before taking a node into its population till the population size is reached.

Now in Figure 5, frame 7, the node G is also discovered by node B but it already has the max-

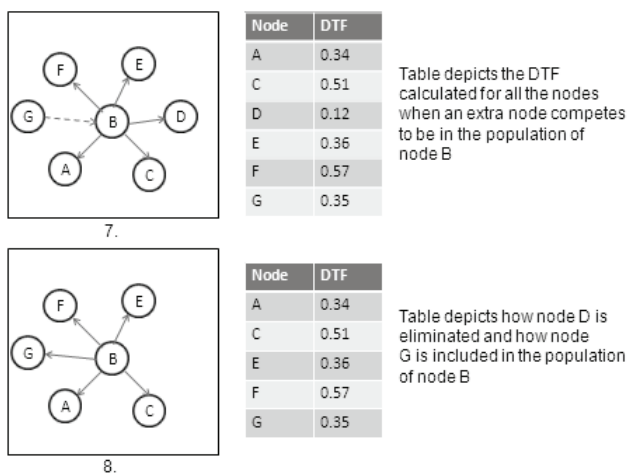


Fig. 5. Illustration on the scenario after an extra node competes to be included into node B's population

imum number of members in its population, but does not know if node G it should be included or not. To resolve this, it calculates the DTF of all the nodes in its population along with the DTF for node G . After calculation, the node B finds that the node D has the minimum DTF and thus decides to eliminate node D and give membership to node G in its population, thus maintaining its population size.

5.3 Updating of next-hops

An updating sub-routine is also used in the protocol for updating the nodes in the population of a certain node about the activity of that node. An UPDATE packet is used to transmit to the next-hop nodes in the routing table of a node whenever there is a change in its characteristic parameters namely: link usage factor, longevity, battery power, reputation, coordinates and hop type hop count, Link Usage Factor, packets generated, packets received and packets forwarded. This is unicast to only nodes which feature as a next-hop in its routing table. It is also to be noted here, that the node sending the UPDATE packet does not send it to that entry in the routing table which it used as a next-hop for data transmission. This is beneficial since all the other nodes will have updated information regarding the node and hence a fresh value of the NHDF. There is no reply for the UPDATE packet to prevent unnecessary usage of battery power of a node.

To prevent the network from getting flooded with update packets and also to reduce the energy consumption at the sending and receiving nodes of these packets, the update packets are sent after specific intervals rather than after every data packet. This interval is dictated by the amount of change in battery power left, number of packets transmitted, packets received and packets generated.

The logic is that no update is sent to the nodes in its population till the values as mentioned above change enough to affect the NHDF appreciably. So every time the node sends messages, it calculates the NHDF leaving out the distance factor and sends an update if the NHDF (without distance) has changed appreciably. Figure 7 given in the next section shows the updating function in progress.

5.4 Overall Protocol

The moment a consumer node enters into the network it is detected by either a clusterhead or a sensor node. If it is detected by a sensor node, the identifying information about this consumer node is sent to a clusterhead. Since the sensor node and the clusterheads have already formed an ad hoc network (as mentioned in the assumptions) they easily communicate with each other. If a clusterhead senses the incoming consumer node or if it receives information about the consumer node, it makes an entry for the consumer node in a separate Serving Consumer Node (SCN) database in which it keeps a record of the nodes being currently served. Also when a clusterhead, say, CH1 provides another clusterhead information about a particular consumer node which was in its domain earlier, then CH1 will remove the consumer node entry from its own SCN database and all those entries in which it features from its routing table. Now, since the clusterhead needs to communicate with the consumer node, it broadcasts hello packets (if required, in the case it has not detected the consumer node on its own) and finds a route to the consumer node. Once the route is found, it sends the data (i.e. information about the services provided by it) to the consumer node. Once the consumer node receives this data packet, it needs to reply to the

clusterhead. For replying, it also uses the route discovery sub-routine to find a route back to the clusterhead. The consumer node needs to find a route back to the clusterhead because, its transmission range is lesser than that of a clusterhead and the route discovered from clusterhead to consumer node may not be suitable for the consumer node's reply. Once a route is found for the clusterhead, the consumer node updates its routing table and sends back the data to the clusterhead. As the consumer node continues communication with the clusterhead, its routing table size keeps increasing as does the number of members in its population. If the population increases beyond population size, it eliminates a few entries based on the DTF to maintain the number of members as the population size dictates. Now suppose, in one of the route discovery sub-routines, more than one route is discovered for a particular destination. In this case, the source node would break the tie between two (or maybe more) next-hop nodes by finding out the NHDF of the two next-hops by using the information in the routing table. The node with the highest NHDF is eventually used to transmit the data packet. Suppose, a node *A* gets next-hops for a destination node *E* as nodes *B* and *G*. The node *A* now needs to calculate the NHDF to find out the optimal route. If it finds node *B*'s NHDF to be greater than node *G*'s it will choose to send the data through *B*.

It should be noted that when a clusterhead is the source, any other kind of node can be the destination. Whereas, when consumer node is the source, only a clusterhead is the destination, other nodes act only as forwarding nodes. In the case of sensor node being the source, again only a clusterhead can be the destination.

5.5 Protocol Functioning: An Example

Let us take an example and understand the entire protocol graphically. Figure 6 shows a sub-section of the ad hoc network at a certain time instant. At this time, node *A* wants to send data to node *E*. The figure also shows the state of routing table of node *B*.

In order to find a route to node *E*, the node *A* carries out the dynamic route discovery protocol as shown in Figure 7. When the sub-routine is executed, node *A* discovers that there are two nodes in its routing table for the same destination node *E*. Once it has added the nodes *B* and *F*

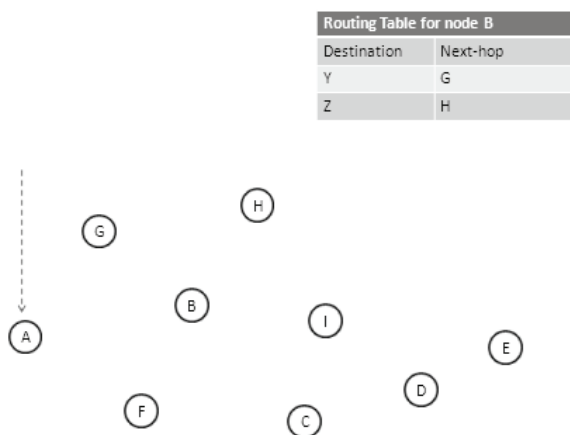


Fig. 6. A sub-section of the network; the arrow shows the source node *A*

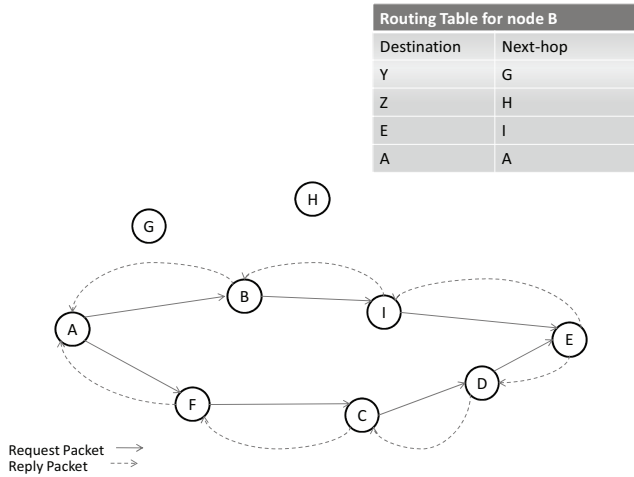


Fig. 7. Dynamic Route Discovery sub-routine

in its routing table, it needs to calculate the NHDF to break the tie and be able to decide as to which route is better.

After calculating the NHDF at node *A*, it discovers that node *B* has a higher NHDF and is hence a better option for reaching node *E*. Thus, node *A* starts data transmission for destination *E* via node *B*. Figure 8 shows that data transmission takes place from node *A* to node *E*.

After data has been sent, node *B* updates the other nodes in its population, namely nodes *G* and *H*, carries out the Updating sub-routine and updates them with the UPDATE packet. Figure 9 shows how node *B* updates only those members in its population which it hasn't communicated with for the latest activity.

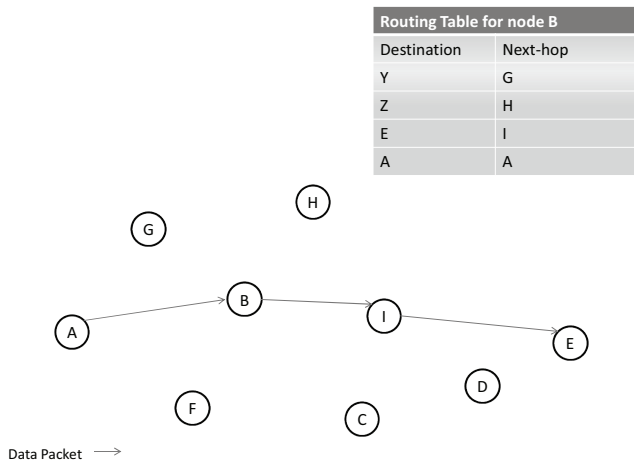


Fig. 8. Data transmission through node *B* after calculating NHDF

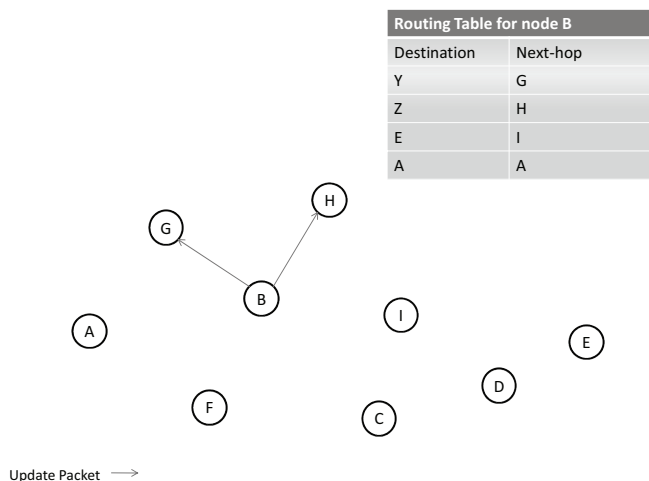


Fig. 9. Updating by node B of the other next-hops in its population

6. EXPERIMENTS AND RESULTS

The simulation experiments conducted were performed using the *Global Mobile Information System Simulator (GloMoSim version 2.03)* [19-20]. It has been designed using the parallel discrete-event simulation capability provided by Parsec [21]. Parsec is a C-based simulation language, developed by the Parallel Computing Laboratory at UCLA, for sequential and parallel execution of discrete-event simulation models. It can also be used as a parallel programming language. GloMoSim currently supports protocols for a purely wireless network.

While testing, the energy used in the transmission/reception of a data packet is kept as 100 times that required for transmitting/receiving a control packet. The network is simulated in a square area environment.

6.1 Energy Efficiency Aspect

In this section of testing, we shall be comparing the DEESR protocol with the MMBCR (Min-Max Battery Cost Routing) protocol as it is an energy efficient protocol to gauge the energy efficiency of the DEESR protocol. We also compare with another widely studied and used protocol, DSR (Dynamic Source Routing) protocol to show how it performs against the standard protocols.

6.1.1 Low Mobility Scenario

Figure 10 depicts the comparison of the energy consumption of DSR, MMBCR and DEESR protocol with the change in the number of nodes in a low mobility (node speeds ranging from 0-5m/s) and low traffic scenario. In this simulation, almost all the packets are being delivered by the network as it is a low mobility, low traffic scenario in which there is no corresponding increase in the number of packets delivered with the increase in the number of nodes in the network. The terrain area of the simulation is 2000 * 2000 square meters. The x-axis gives the num-

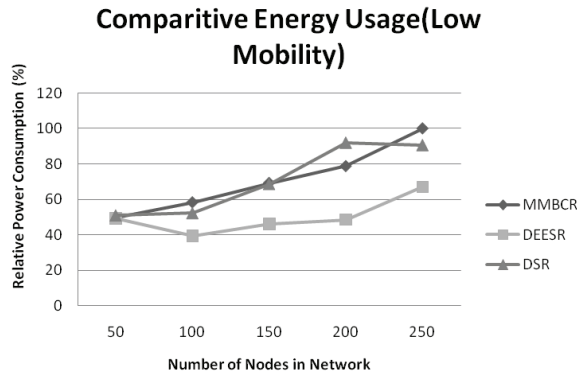


Fig. 10. Comparison between energy usages of different protocols including DEESR

ber of nodes present in the network. The y-axis represents the power consumption relatively with the maximum power consumption being marked as 100 units.

As can be seen from Figure 10, the proposed DEESR protocol is found to be more energy efficient and conserves the total amount of energy that is being utilized in the network. Also, in general the energy required by a network is more as the number of nodes increases. This is because more number of control packets gets exchanged in the network. The figure also shows that the overall energy required keeps increasing with the increase in the number of nodes.

6.1.2 High Mobility Scenario

Figure 11 shows the energy comparison between the three protocols at a high mobility scenario, where the speed of each node ranges from 5m/s to 20m/s. The x and y axes have the same units as those of Figure 10. In this figure, it can easily be seen that DEESR protocol is much more energy efficient than both MMBCR and DSR in all the given scenarios. In fact, DSR has highest energy consumption in 4 out of 5 scenarios which is marginally greater than that of

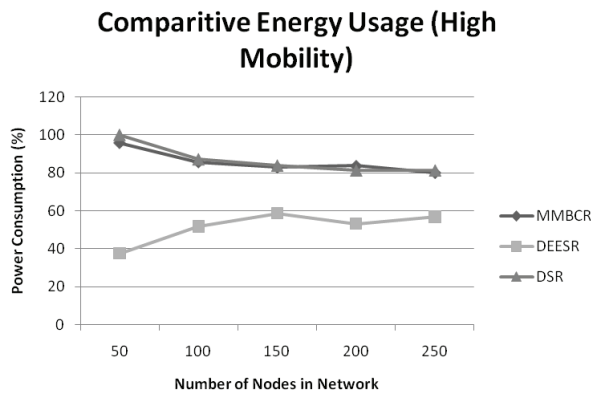


Fig. 11. Comparison between energy usages of different protocols including DESSR at high mobility scenario

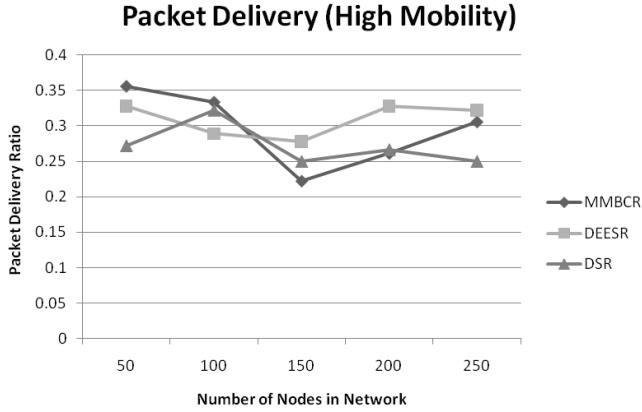


Fig. 12. Comparison of packet delivery ratio by the three protocols including DEESR in high mobility scenario

MMBCR. Though the energy efficiency of DEESR protocol is much better than both these protocols, the absolute energy used increases for DEESR while it decreases for the other two protocols.

Figure 12 shows the packets delivery ratio for the three protocols in the same scenario. The y-axis shows the packet delivery ratio which can be defined as:

$$PDR = p_r / p_t \tag{5}$$

Where,

p_r = Number of packets received.

p_t = Number of packets transmitted.

Since, this scenario has very high mobility of nodes, which is rarely found in a real-life scenario, almost all the protocols discussed struggle with it. It illustrates a very poor packet delivery ratio ranging from 0.22 to 0.36 for all the three protocols. However, our DEESR protocol is able to provide better packet delivery ratios with the increase in the number of nodes as shown by the figure and the packet delivery ratio is considerably better than the other two protocols in 4 out of the 5 results. The placement of the nodes was done randomly and in this particular case, the placement was done more favorably resulting in high packet delivery ratios.

6.1.3 Medium Mobility Scenario

Figure 13 and 14 show a scenario with medium mobility that is where the speed of each node varies from 0m/s to 10m/s. Here, to add variety, we have varied the terrain size while keeping the number of nodes in the network constant at 100. The area of the network is varied from $25 \times 10^4 \text{ m}^2$ to $625 \times 10^4 \text{ m}^2$. In Figure 13 the three protocols are again tested on energy efficiency at the medium mobility level.

The y-axis depicts the relative energy consumption as in figures 10 and 11. The x-axis represents the area of the network as a multiple of 10^4 m^2 . From the figure, we can infer that the energy efficiency of DEESR protocol although higher in small terrain size becomes better than the

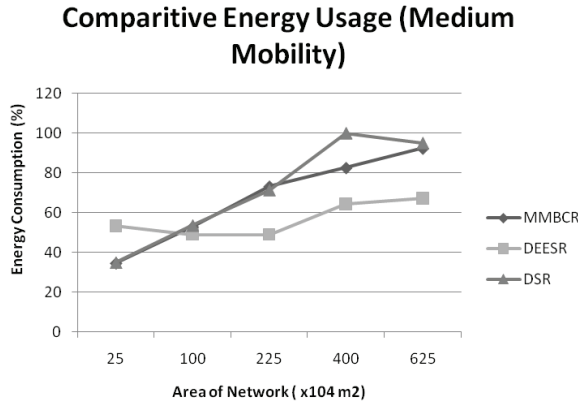


Fig. 13. Comparison of energy consumption at medium mobility with varying network area

other two protocols as the size of the terrain increases.

Figure 14 shows the comparison of the packet delivery ratio for the same scenario. At first glance, the figure tells us that there is a sharp decrease in the packet delivery ratio for the three protocols as the area increases. This is intuitive, since as the area increases, the nodes could be more spread out. If the nodes are more spread out, lower number of nodes would be present in the transmission range of a particular node which might lead to a failure of route discovery and therefore a failure to send packets to the desired destination; finally leading to a low packet delivery ratio.

From Figure 14, we can infer that the DEESR protocol follows the general trend when the nodes have medium mobility, but in the end it does marginally better than DSR and MMBCR when the area increases beyond 4,000,000 m². The packet delivery ratio varies from 0.29 to 0.98 for all the three protocols combined, whereas it varies from 0.46 to 0.98 for the proposed DEESR protocol alone.

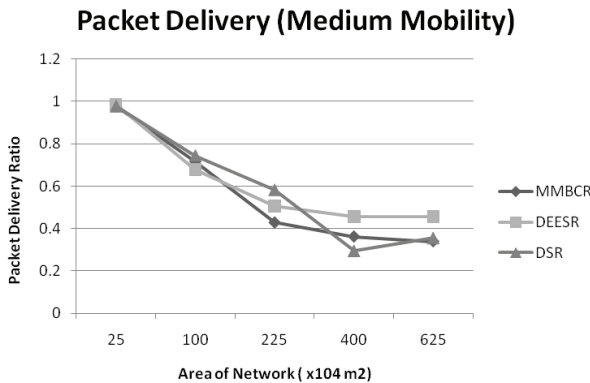


Fig. 14. Comparison of packet delivery of the protocols under varied network area size

6.2 Apparent cost of network

As the network is deployed in an urban environment, with the nodes being the mobile computing devices of real people, we need to keep the apparent cost of the network in mind. Since the nodes are not only used for the network but are also used for other services such as communication, text-messaging, gaming, and music, the network cannot be allowed to hog the battery of a node for its usage. Thus, we define a new parameter called as “Dead Nodes.” A dead node is a consumer node which used up more than 10% of its energy while being in the network and has gone beyond a certain energy threshold currently taken as 50% of the total battery power. Thus, the aim of our network is not only to maximize the energy efficiency of the network, but also to minimize the number of dead nodes in the network.

We consider a medium mobility scenario in which the node speed ranges from 0m/s to 10 m/s. The nodes have been assigned different levels of battery power from the start randomly and the network is monitored to find any dead nodes in the network.

In Figure 16, the simulation area of the network has been kept constant at 2000x2000 m² while varying the number of nodes present in the network. As can be seen from the figure, the DEESR protocol fares considerably well against the MMBCR Protocol which is also an energy aware protocol.

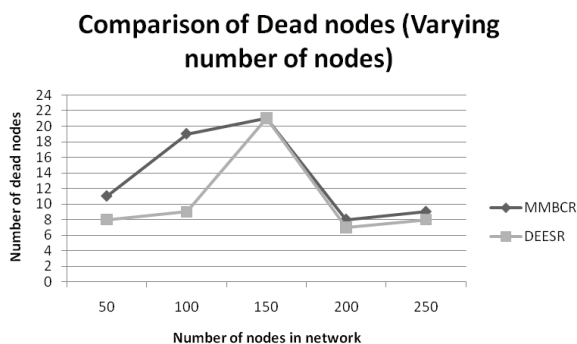


Fig. 15. Comparison of number of dead nodes with constant terrain size and increasing nodes

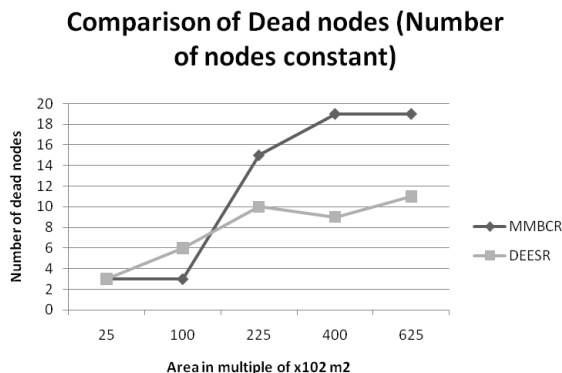


Fig. 16. Comparison of number of dead nodes with varying terrain size

Figure 16 shows the scenario in which the number of nodes has been kept constant at 100 while the terrain of the network has been varied. As the terrain size increases, MMBCR protocol shows an increase in the number of dead nodes with most of the data packets being transmitted through set routes in the network. On the other hand, the DEESR protocol performs better and is able to keep the number of dead nodes to a minimum.

6.3 Security Aspect

In this section of testing, we shall be comparing the DEESR protocol with SNEP and QDV protocols to gauge the security aspect of the DEESR protocol.

First of all, let us see the effect of malicious nodes on the packet delivery ratio in the network. In Figure 17, the x-axis represents the percentage of malicious nodes in the network. The y-axis represents the packet delivery ratio for the three protocols. As can be seen from the figure, the packet delivery ratio of DEESR protocol is considerably better than both the QDV protocol and the SNEP protocol. But, in general, with the increase in the percentage of malicious nodes, the packet delivery ratios drop significantly in the network, dropping to as low as 0.6 in the case of SNEP.

In Figure 18, we compare the detection times of the three protocols while varying the percentage of malicious nodes in the network. Thus, here also the x-axis represents the percentage of malicious nodes in the network. The y-axis represents the time taken to detect the malicious nodes in the network. It can also be seen that the time taken for detection of the malicious nodes is minimum for DEESR protocol in all the instances. For both QDV and DEESR protocol, the amount of time taken to detect malicious nodes keeps decreasing when the percentage of malicious nodes keeps increasing.

While Figure 18 gives a comparison of the detection times based on the percentage of malicious nodes, Figure 19 gives a comparison of the detection times by varying the number of nodes in the network while keeping the number of malicious nodes constant. As can be seen from the Figure, the detection times increase with the increase in the number of nodes in the network since the malicious nodes are able to hide better. However, still DEESR protocol fares well in this scenario as well.

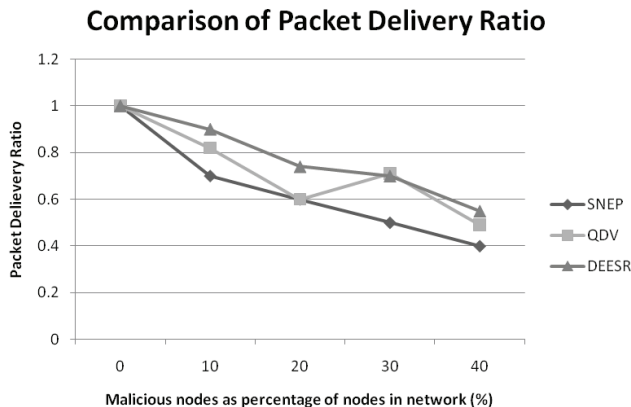


Fig. 17. Comparison of Packet Drop ratio

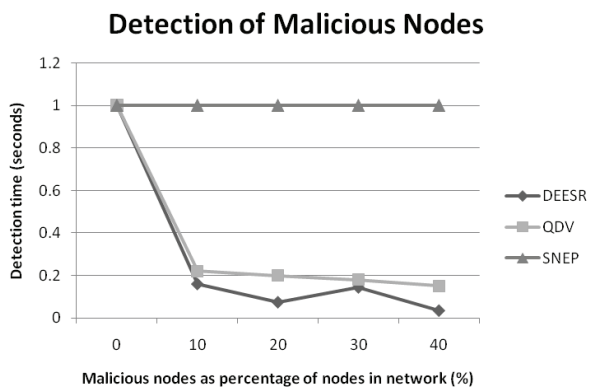


Fig. 18. Comparison of time taken to detect malicious nodes in the network

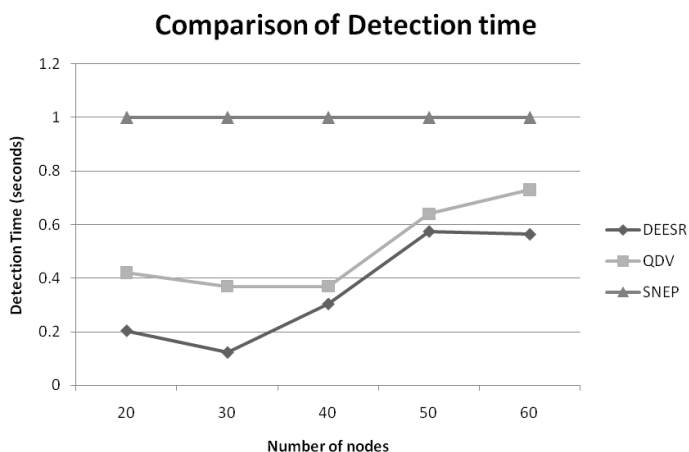


Fig. 19. Comparison of Detection Time

7. CONCLUSION

There is a need for energy efficient secure routing protocols to make wireless sensor networks be used in urban environments by real people. Most of the existing sensor networks usually have dedicated sensor devices with a single purpose in mind, but in an urban environment, the devices used by the people often have multiple tasks. Thus, it is important not only to keep the footprint of the running algorithm low, but also to enable the user to have trust in the system. The proposed DEESR protocol is an attempt geared towards the same objective. As shown in the energy efficiency testing section of this paper, the proposed protocol is not only energy efficient but is highly resilient in low mobility, medium mobility and high mobility scenarios. Apart from this, the packet delivery ratio delivered by the protocol is comparable and sometimes better than some of the well-established protocols like DSR and MMBCR.

As shown in the security testing section of this paper, we believe that the protocol is safe for communication in different scenarios, generally delivering better results in terms of the number

of dead nodes and the packet delivery percentage over the QDV and the SNEP security based protocols. Thus, we believe that the proposed DEESR protocol can provide reasonable amounts of services while keeping a minimum energy footprint along with a good level of security.

This protocol is a first step towards a future in which everybody will be interconnected with every other person through various kinds of handheld or wearable devices without using much energy which meets the goal of having Green communication and networking systems.

REFERENCES

- [1] Bhagwat, C. E., "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", *Proc of SIGCOMM94 Conf on Communications Architectures, Protocols and Applications*, Vol.24, No.4, pp.234-244, October, 1994.
- [2] C. Intanagonwiwat, R. Govindan and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks", *Proc of ACM/IEEE Intl Conf on Mobile Computing and Networking (MOBICOM)*, pp.56-67, 2000.
- [3] Chang, J.-H. T., "Energy Conserving Routing in Wireless Ad-Hoc Networks", *INFOCOM00*, pp.22-31, March, 2000.
- [4] Gerla, C. R., "Adaptive Clustering for Mobile Wireless Networks", *IEEE J. Select. Areas Communications*, Vol.15, No.7, pp.1265-1275, September, 1997.
- [5] J. Kulik, W. H., "Negotiation-Based Protocols for Disseminating Information in Wireless Sensor Networks", *ACM Wireless Networks*, Vol.8, No.2-3, pp.169-185, 2002.
- [6] Khuller, S. B., "A Clustering Scheme for Hierarchical Control in Multi-hop Wireless Networks", *Proc of IEEE INFOCOM*, Vol.2, pp.1028-1037, April, 2001.
- [7] P. Jackquet, P. M, T. Clausen, A. Laouiti, A. Qayyum, L. Viennot, "Optimized Link State Routing Protocol for Ad Hoc Network", *IEEE INMIC*, pp.62-68, 2001.
- [8] Perkins, C. E, Royer, E. M., "Ad-hoc on-demand distance vector routing", *In Proc of WMCSA99: 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pp.90-100, February, 1999.
- [9] A.P. Chandrakasan, A.C. Smith, and W.B. Heinzelman, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks", *IEEE Transactions on Wireless Communications*, Vol.1, No.4, pp.660-670, October, 2002.
- [10] David B. Johnson and David A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks", *In Mobile Computing*, Tomasz Imielinski and Hank Korth (Editor), Kluwer Academic Publishers, pp.153-181, 1996.
- [11] Zhang Li, Yu Zhen-Wei, Zhang Yang, Zhang Chun-Kai, "A Power-aware Adaptive Dynamic Routing Scheme for Wireless Ad Hoc Networks", *Proc of IEEE Intl Conf Networking, Sensing and Control ICNSC'08*, pp.966-970, April, 2008.
- [12] Suresh Singh, Mike Woo, C. S. Raghavendra. "Power-aware routing in mobile ad hoc networks", *Proc of 4th annual ACM/IEEE Intl Conf on Mobile computing and networking*, pp. 181-190, October 1998.
- [13] S. Yi, P. Naldurg, and R. Kravets, "Security-aware routing protocol for wireless ad hoc networks," *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing*, pp 299-302, Long Beach, CA, USA, October, 2001.
- [14] M. S. Obaidat, and N. Boudriga, "Security of e-Systems and Computer Networks, Cambridge University Press, Cambridge, UK, 2007.
- [15] Adrian Perrig, Robert Szewczyk, J.D. Tygar, Victor Wen and David E. Culler, "SPINS: Security Protocols for Sensor Networks", *Proceedings of 7th Annual International Conference on Mobile Computing and Networking*, pp. 189-199, Rome, Italy, July, 2001.
- [16] Sanjay K. Dhurandher, S. Misra, M. S. Obaidat and N. Gupta, "An Ant Colony Optimization Approach for Reputation and Quality-of-Service-Based Security in Wireless Sensor Networks", *Security and Communication Networks*, John Wiley & Sons, Vol.2, No.2, pp.215-224, March/April, 2009.

- [17] Chris Karlof, Naveen Sastry, David Wagner, "TinySec: A Link Layer Security Architecture for Wireless Sensor Networks", *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, SenSys 2004*, Baltimore, MD, USA, pp.162-175, November 3-5, 2004.
- [18] Christos Xenakis, Evangelos Gazis, Lazaros Merakos, "Secure VPN Deployment in GPRS Mobile Network," *Proc. European Wireless 2002*, Florence, Italy, pp.293-300, February, 2002.
- [19] M. Takai, L. Bajaj, R. Ahuja, R. Bagrodia and M. Gerla, *GloMoSim: A Scalable Network Simulation Environment*, Technical report 990027, UCLA, Computer Science Department, 1999.
- [20] GloMoSim: Global Mobile Information Systems Simulation Library. <http://pcl.cs.ucla.edu/projects/gloMosim/>.
- [21] R. Bargodia, R. Meyer, M. Takai, Y.-A. Chen, X. Zeng, J. Martin, H.Y. Song, "PARSEC: A Parallel Simulation Environment for Complex Systems", *IEEE Computer*, Vol.31, No.10, pp.77- 85, 1998.



Mohamamd S. Obaidat

Professor Mohammad S. Obaidat is an internationally well known academic/researcher/ scientist. He received his Ph.D. and M. S. degrees in Computer Engineering with a minor in Computer Science from The Ohio State University, Columbus, Ohio, USA. Dr. Obaidat is currently a full Professor of Computer Science at Monmouth University, NJ, USA. Among his previous positions are Chair of the Department of Computer Science and Director of the Graduate Program at Monmouth University and a faculty member at the City University of New York. He has received extensive research funding and has published Ten (10) books and over Four Hundred and Sixty (460) refereed technical articles in scholarly international journals and proceedings of international conferences, and currently working on three more books. Prof. Obaidat is the author of a new upcoming books: *Wireless Sensor Networks* (Cambridge University Press). He is also the editor to 2 new upcoming books: *Cooperative Networking* (John Wiley & Sons 2010) and *Pervasive Computing and Networking* (John Wiley & Sons 2010). Prof. Obaidat is the author of the book entitled: "Fundamentals of Performance Evaluation of Computer and Telecommunications Systems," by John Wiley & Sons in 2010. Dr. Obaidat is the Editor of the Book entitled, "E-business and Telecommunication Networks", published by Springer in 2008. He is the co-author of the book entitled, "Security of e-Systems and Computer Networks" published by Cambridge University Press in 2007. He is the co-author of the Best Selling Book, "Wireless Networks" and "Multiwavelength Optical LANs" published by John Wiley & Sons (2003). Obaidat is the editor of the book, *APPLIED SYSTEM SIMULATION: Methodologies and Applications*, published by Kluwer (now Springer) in 2003. Professor Obaidat has served as a consultant for several corporations and organizations worldwide. Mohammad is the Editor-in-Chief of the *International Journal of Communication Systems* published by John Wiley. He served as an Editor of *IEEE Wireless Communications* from 2007-2010. Between 1991-2006, he served as a Technical Editor and an Area Editor of *Simulation: Transactions of the Society for Modeling and Simulations (SCS) International*, TSCS. He also served on the Editorial Advisory Board of *Simulation*. He is now an editor of the *Wiley Security and Communication Networks Journal*, *Journal of Networks*, *International Journal of Information Technology, Communications and Convergence*, *IJITCC*, *Inderscience*. He served on the International Advisory Board of the *International Journal of Wireless Networks and Broadband Technologies*, *IGI-global*. Prof. Obaidat is an associate editor/ editorial board member of seven other refereed scholarly journals including two *IEEE Transactions*, *Elsevier Computer Communications Journal*, *Kluwer Journal of Supercomputing*, *SCS Journal of Defense Modeling and Simulation*, *Elsevier Journal of Computers and EE*, *International Journal of Communication Networks and Distrib-*

uted Systems, The Academy Journal of Communications, International Journal of BioSciences and Technology and International Journal of Information Technology. He has guest edited numerous special issues of scholarly journals such as IEEE Transactions on Systems, Man and Cybernetics, SMC, IEEE Wireless Communications, IEEE Systems Journal, SIMULATION: Transactions of SCS, Elsevier Computer Communications Journal, Journal of C & EE, Wiley Security and Communication Networks, Journal of Networks, and International Journal of Communication Systems, among others. Obaidat has served as the steering committee chair, advisory Committee Chair and program chair of numerous international conferences including the IEEE Int'l Conference on Electronics, Circuits and Systems, IEEE International Phoenix Conference on Computers and Communications, IEEE Int'l Performance, Computing and Communications Conference, IEEE International Conference on Computer Communications and Networks, SCS Summer Computer Simulation Conference, SCSC'97, SCSC98-SCSC2005, SCSC2006, the International Symposium on Performance Evaluation of Computer and Telecommunication Systems since its inception in 1998, International Conference on Parallel Processing, Honorary General Chair of the 2006 IEEE Intl. Joint Conference on E-Business and Telecommunications, ICETE2006. He served as General Co-Chair of ICETE 2007-ICETE 2010. He has served as the Program Chair of the International Conference on Wireless Information Networks and Systems from 2008-Presnet. He is the co-founder and Program Co-Chair of the International Conference on Data Communication Networking, DCNET since its inception in 2009. Obaidat has served as the General Chair of the 2007 IEEE International Conference on Computer Systems and Applications, AICCSA2007, the IEEE AICCSA 2009 Conference. and the 2006 International Symposium on Adhoc and Ubiquitous Computing (ISAHUC'06). He is the founder of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems, SPECTS and has served as the General Chair of SPECTS since its inception. Obaidat has received a recognition certificate from IEEE. Between 1994-1997, Obaidat has served as distinguished speaker/visitor of IEEE Computer Society. Since 1995 he has been serving as an ACM distinguished Lecturer. He is also an SCS distinguished Lecturer. Between 1996-1999, Dr. Obaidat served as an IEEE/ACM program evaluator of the Computing Sciences Accreditation Board/Commission, CSAB/CSAC. Obaidat is the founder and first Chairman of SCS Technical Chapter (Committee) on PECTS (Performance Evaluation of Computer and Telecommunication Systems). He has served as the Scientific Advisor for the World Bank/UN Digital Inclusion Workshop-The Role of Information and Communication Technology in Development. Between 1995-2002, he has served as a member of the board of directors of the Society for Computer Simulation International. Between 2002-2004, he has served as Vice President of Conferences of the Society for Modeling and Simulation International SCS. Between 2004-2006, Prof. Obaidat has served as Vice President of Membership of the Society for Modeling and Simulation International SCS. Between 2006-2009, he has served as the Senior Vice President of SCS. Currently, he is the President of SCS. One of his recent co-authored papers has received the best paper award in the IEEE AICCSA 2009 international conference. He also received the best paper award for one of his papers accepted in IEEE GLOBCOM 2009 conference. Dr. Obaidat received very recently the Society for Modeling and Simulation Intentional (SCS) prestigious McLeod Founder's Award in recognition of his outstanding technical and professional contributions to modeling and simulation.

He has been invited to lecture and give keynote speeches worldwide. His research interests are: wireless communications and networks, telecommunications and Networking systems, security of network, information and computer systems, security of e-based systems, performance evaluation of computer systems, algorithms and networks, high performance and parallel computing/computers, applied neural networks and pattern recognition, adaptive learning and speech processing. Recently, Prof. Obaidat

has been awarded a Nokia Research Fellowship and the distinguished Fulbright Scholar Award. During the 2004/2005, he was on sabbatical leave as Fulbright Distinguished Professor and Advisor to the President of Philadelphia University in Jordan, Dr. Adnan Badran. The latter became the Prime Minister of Jordan in April 2005 and served earlier as Vice President of UNESCO. Prof. Obaidat is a Fellow of the Society for Modeling and Simulation International SCS, and a Fellow of the Institute of Electrical and Electronics Engineers (IEEE).



Sanjay Kumar Dhurandher

Sanjay Kumar Dhurandher received the M. Tech. and Ph.D. Degrees in Computer Sciences from the Jawaharlal Nehru University, New Delhi, India. He is presently working as an Assistant Professor in the Division of Information Technology, Netaji Subhas Institute of Technology (NSIT), University of Delhi, India. He is also the Head of the Advanced Centre CAITFS, Division of Information Technology, NSIT, University of Delhi. Prior to this, from 1995 to 2000, he worked as a Scientist/Engineer at the Institute for Plasma Research, Gujarat, India which is under the Department of Atomic Energy, India. Dr. Dhurandher is presently also serving as the **Associate Editor** to the International Journal of Communications Systems. His current research interests include wireless ad-hoc networks, sensor networks, computer networks, network security and Underwater Sensor Networks.



Deepank Gupta

He received BE degree in Information Technology from NSIT (Delhi University) in 2009. He is now working as a Software Engineer in Test at Google India. He has also been an active contributor to various open-source projects such as OLPC and Plazi. His research interests include ad-hoc sensor networks, data mining, and machine learning.



Nidhi Gupta

Nidhi Gupta received the B.E. degree in Information Technology from Netaji Subhas Institute of Technology (NSIT), Delhi, India in 2009. She is currently into Software Development in Adobe Systems, Noida. Her research interests include database security, information security and wireless adhoc networks wherein she has published research papers and journal.



Anupriya Asthana

Anupriya Asthana received the B.E. degree in Information Technology from Netaji Subhas Institute of Technology (NSIT), Delhi, India in 2009. She is currently working as Subject Matter Expert in Amdocs DVCI, Pune. She has interests in network security, information security and wireless communication networks.