ABSTRACT: Energy conservation is a critical issue in Mobile Ad hoc Networks (MANETs) for node and network life, as the nodes are powered by batteries only. In order to maximize the lifetime of MANETs, traffic should be sent through a shortest path that does not contain nodes with low residual energy and consumes minimum total transmission power. The above objectives cannot be satisfied simultaneously by applying the existing power aware routing algorithms. For this reason, several power-aware routing approaches have been proposed in the literature attempting to maximize the network lifetime. An interesting approach called Minimum Drain Rate (MDR) aims to predict the lifetime of nodes according to the current traffic conditions. However, it does not guarantee that the total transmission power is minimized over a chosen route. In this paper, we provide an new approach called Secure Extended Optimal Energy Drain Rate (SEOEDR) able to extend the MANET lifetime and duration of the path, and also to minimize the total transmission power consumed per packet. In our approach, the traffic is routed such that the energy consumption is balanced among the nodes in proportion to their energy reserves. We also handle the selfishness of nodes by considering the node reputation while making the routing decisions. It results in selection of routes that are simultaneously energy efficient and reliable. The simulation results that we have conducted during our study show the relevance of our algorithm, in terms of energy saving and overall network lifetime, in comparison with existing ones.

Categories and Subject Descriptors
C.1.4 [Parallel Architectures]; Mobile Processors C.2.2 [Network Protocols]: Routing Protocols C.2.1 [Network Architecture and Design]; Network Communications

General Terms
Wireless Communication, Network Protocols

Keywords: DSR, Energy efficient routing protocol, NIC, MAC layer, Battery cost, MANET,

Received 23 Sep. 2006; Revised and accepted 21 August 2007

1. Introduction
Mobile Ad hoc NETwork (MANET) [1, 25] is a collection of wireless devices that come together to form a self-organizing network without any support from the existing fixed communication infrastructure. In such a network, each device plays a role of a router and has limited battery energy. In addition, the network topology can constantly change. Thus, it is widely accepted that conventional routing protocols are not appropriate for MANETs, and consequently the design of routing protocols for such networks is a challenging issue taking in particular power factor into consideration, because these devices are battery dependent. The purpose of power-aware routing protocols [10] is to maximize the network lifetime. The network lifetime is defined as the time until the first node in the network runs out of energy (i.e. it is the period from the time instant when the network starts functioning to the time instant when the first node runs out of energy). One of the major concerns in ad hoc wireless networks is reducing the node energy consumption. In fact, nodes are usually powered by batteries of limited capacity. Once the nodes are deployed, it is very difficult or even impossible to recharge or replace their batteries in many application scenarios. Hence, reducing power consumption is often the only way to extend the network lifetime. The objective is to maximize the network lifetime while guaranteeing the required traffic rate. The power-aware routing protocols should consider energy consumption from the viewpoints of both the network and the node levels. From the network viewpoint, the best route is one that minimizes the total transmission power, while from the node viewpoint, the best route should avoid the nodes with lower power [1]. It is difficult to achieve these two objectives simultaneously. Minimizing the total energy consumption tends to favor min-hop routes (i.e. route to destination with minimum number of hops). However, if the min-hop routes repeatedly include the same node, this later will exhaust its energy much earlier than the other nodes and the network lifetime will decrease [2, 3]. In [4], the authors proposed a new link cost for reliable transmission that includes the energy consumption for data packets as well as that for signaling packets in MAC layer. Once a new link cost was derived, then the Dynamic Source Routing (DSR) [7, 11, 15] can be modified with the new link cost. However, there are some drawbacks with such straightforward modification. First, the routing overhead for the route discovery is very high, which consumes a lot of energy. Second, the route setup time is very long. Third, the route maintenance scheme is not suitable for dynamic environments, such as in mobile scenarios. To address the above issue, another approach is proposed in [5] providing a progressive energy efficient routing protocol. Contrary to other energy efficient routing protocols, it attempts to find the optimal path at one shot and to maintain the route reactively. However, it targets at energy saving only with reduced overhead and does not concentrate on maximizing the network lifetime as it is needed to balance the remaining battery power at each node according to current traffic conditions.
In this paper, we combine and extend some existing approaches in a way to predict the lifetime of nodes according to current traffic conditions combined with the value of the remaining battery capacity and appropriate threshold value. In our approach, we can extend the lifetime of each node, while prolonging the lifetime of each connection. This can be achieved by detecting whether or not a node can be part of an active route, and can be applied to any existing MANET routing protocols as a route establishment criterion. We also consider node selfishness in a MANET due to the fact that some nodes may not always be in the best interest of forwarding data to others. For instance, several nodes might drop packets preferentially in order to conserve energy while malicious ones drop packets to disrupt the network operations by executing Denial of Service (DOS) attacks. Several energy-based mechanisms are proposed in literature to consider this problem [14, 20, 21, 22] but, to the best of our knowledge, none of them integrates the reputation factor.

The rest of this paper is organized as follows. Section 2 describes various power-aware routing protocols proposed in the literature. In Section 3, we present our approach and give an illustrative example to explain how it works. In Section 4, we present the performance evaluations and analysis of our approach. Finally, we present our conclusions and draw some future directions.

2. Related Work
To maximize the MANET life time and minimize the total transmission power required in packet routing, several power-aware routing algorithms are proposed in the literature [6, 8, 9]. We detail some of them below.

2.1 Minimum Total Transmission Power Routing (MTPR)
In [13], the authors propose an interesting algorithm called Minimum Total Transmission Power Routing (MTPR) making use of a simple energy metric representing the total energy consumed along a route. Formally, consider a generic route $R_g = N_0, N_1, \ldots, N_d$, where $N_0$ is the source node, $N_d$ is the destination node, and $T(N_i, N_j)$ denotes the energy consumed when transmitting over the hop $(N_i, N_j)$, the total transmission power $P$ of the route $R_g$ is calculated as:

$$ P(R_g) = \sum_{i=0}^{d-1} T(N_i, N_{i+1}) $$

(1)

The optimal route $R_o$ must have the minimum total transmission power as follows:

$$ P(R_o) = \min_{R_j \in R^*} P(R_j) $$

(2)

Where $R^*$ is the set of all possible routes. Although MTPR can reduce the total transmission power consumed per packet, it however does not reflect directly the lifetime of each node.

2.2 Min-Max Battery Cost Routing (MMBCR)
In [16], the authors consider the remaining battery capacity of each node as a metric to describe the lifetime of each node. The authors assume that when a node has less capacity, it is better to avoid forwarding packets to. Thus, they propose the following battery cost function of a node $N_i$:

$$ F_i(t) = \frac{1}{C_i(t)} $$

(3)

Where $C_i(t)$ is the battery capacity of a node $N_i$ at time $t$. If only the sum of battery costs is considered, a route containing nodes with little remaining battery capacity may still be selected. The Min-Max Battery Cost Routing (MMBCR) defines the route cost as:

$$ C(R) = \max_{\forall N_i \in R_j} F_i(t) $$

(4)

The optimal route $R_o$ is obtained as:

$$ C(R_j) = \min_{R_j \in R^*} C(R_j) $$

(5)

Where $R^*$ is the set of all possible routes.

Since MMBCR [8] considers the weakest and crucial node over the path, a route with the best condition among the path impacted by each crucial node over each path is selected.

2.3 Conditional Max-Min Battery Capacity Routing (CMMBCR)
Prolonging the lifetime of each node, while minimizing the total transmission power consumed per packet is not a trivial method. The MMBCR mechanism, for example, does not guarantee that the total transmission power consumed per packet over a chosen path is minimized. An extended version called Conditional Max-Min Battery Capacity Routing (CMMBCR) is provided in [12]. It attempts to perform a hybrid approach between MTPR and MMBCR. It considers both the total transmission energy consumption of routes and the remaining power of nodes. When all nodes in some possible routes have sufficient remaining battery capacity (i.e., above a threshold $\vartheta \in [0, 100]$), a route with minimum total transmission power is chosen among possible ones. The relaying load for most nodes must be reduced, because less total power is required to forward packets for each connection, and their lifetime is extended. However, if all routes have nodes with low battery capacity, a route including nodes with the lowest battery capacity must be avoided to extend the lifetime of these nodes. The battery capacity $B$ of a route $R_i$ at time $t$ is defined as follows:

$$ B_i(t) = \min_{\forall N_j \in R_i} C_j(t) $$

(6)

Unfortunately, it is not possible to efficiently determine the battery capacity threshold $\vartheta$. CMMBCR either needs a centralized server to keep track of the energy status of all the mobile nodes, or nodes must inform one another about the remaining power of each node, which is energy-consuming. Moreover, if $\vartheta$ is taken as an absolute value, there is no easy way to define the threshold.
value without considering the current network status, e.g., the network traffic.

2.4 Minimum Energy Drain Rate algorithm (MDR)
In [11,12], a new metric called the Minimum Energy Drain Rate (MDR) is used. It consists of using the rate at which the energy gets dissipated at a given node in conjunction with residual battery capacity in order to predict the lifetime of nodes according to the current traffic conditions. Each node monitors its energy consumption and maintains its battery power drain rate value DRi by averaging the amount of energy consumption and estimating the energy dissipation per second during the given past interval. DRi indicates how much the average energy is consumed by a node Ni per second during the interval. The node battery power drain rate actual value is calculated, using the well known exponential weighted moving average method applied to the drain rate values DR old and DR sample representing the previous and the newly calculated values, as follows.

\[
DR_i = \alpha \times DR_{old} + (1-\alpha) \times DR_{sample} \quad (7)
\]

The ratio \( RBP_i/DR_i \), where RBP denotes the Residual Battery Power of a node Ni, indicates when the remaining battery of Ni is exhausted (i.e. how long Ni can keep up routing operations with current traffic conditions). The maximum lifetime \( L \) of a given route \( R_p \) is determined by the minimum value over the path as:

\[
L_p = \min_{N \in R_p} \frac{RBP_i}{DR_i} \quad (8)
\]

This is based on selecting the route \( R_m \) contained in the set of all possible routes \( R^* \), between the source and the destination nodes that presents the highest maximum lifetime value as follows:

\[
R_m = \max_{N \in R^*} L_i \quad (9)
\]

However, MDR does not guarantee that the total transmission power is minimized over a chosen route, as in MMBCR.

3. Proposed Approach
We propose a new approach called Secure Extended optimal Energy Drain Rate (SEOEDR) able to choose the best path with minimum total transmission power among all the possible paths while avoiding the selfishness of the nodes.

3.1 Best Path Selection
The main idea, which consists of choosing a path with minimum total transmission power among all the possible paths constituted by nodes with a lifetime higher than a given threshold \( \varphi \). This threshold \( \varphi \) is computed on the basis of how long each node Ni can sustain its current traffic with its remaining battery power (RBPi) and drain rate (DRi) without power breakage. Formally, \( \varphi \geq \min_{N_i \in R} \frac{RBP_i}{DR_i} \). If no route verifies this condition, EOEDR switches to the basic MDR mechanism. Let Q be the set of all available routes between the source S and destination D, and A be a subset of Q where the routes of A satisfying the appropriate threshold value. If \( Q \cap A = \emptyset \) then we apply the MTPR algorithm in order to select the best route. Otherwise, we apply the MDR algorithm to select a route with nodes having maximum cost function among the minimum cost function. When no route verifies the required threshold value, it is better to predict the lifetime of the node based on the residual battery capacity to increase the current traffic condition. Moreover, we employ a sleep mode required in MANET routing. This is of great importance because total power consumption significantly depends upon the overhearing process (Figure 1). This means that any protocol that leaves a mobile receiver unnecessarily listening is wasting power. This will be detailed later in the simulation study.

In Figure 2, we give a simple example of a MANET with 10 nodes with a source node \( N_s \) and a destination node \( N_d \). Each node \( N_i \) is characterized by the couple (RBPi, DRi). Below, we explain the different steps allowing to route packets from \( N_s \) to \( N_d \):

1. Identify the set Q which contains all available routes between \( N_s \) and \( N_d \): \( Q = \{ R_1, R_2, R_3, R_4 \} \).
2. Find the node with minimum cost function in all possible routes (i.e. minimum value of \( RBP_i/DR_i \) in the corresponding route) in order to compute the lifetime threshold. In our example, the node \( N_i \) has \( 6/3=2 \) cost value indicating that it will dissipate its energy in 2 seconds.
3. Construct the set A containing routes having nodes lifetime values greater than the threshold (assuming the threshold value \( \varphi \) is 3): \( A = \{ R_3, R_4 \} \).
4. Test the intersection set between Q and A: \( Q \cap A = \{ R_3, R_4 \} \).
5. As \( Q \cap A \neq \emptyset \), then apply MTPR on \( R_3 \) and \( R_4 \) in order to select the best route (since both satisfy the required residual battery power, either \( R_3 \) or \( R_4 \) can be chosen). This simple energy metric is enough when there are routes satisfying available required battery power.
6. If \( Q \cap A = \emptyset \), then apply MDR on \( R_3 \) and \( R_4 \) to select the node having the minimum cost function. If no route verifies the required threshold value, it is better to predict the lifetime of the node based on the residual battery capacity to increase the current traffic condition. As per the metric of MDR, the maximum lifetime of a given path is determined by selecting a route with the maximum cost function value among the minimum cost function nodes. Hence, for this example, it selects \( R_3 \) as the optimal route.

\[1\] If the threshold is 8 for instance.
The value of the range \[0, 1\]. This parameter intuitively reflects how willing a user will have to spend part of its energy resource for a connection request will not be blocked. This however, implies that its request for a traffic connection is declined by the network because of the non-cooperation of the relay nodes. If we assume that network nodes may have different behavior, reflecting the level of selfishness/altruism of the nodes. We describe S-EOEDR by taking DSR into consideration. Each node \(N_i\) in the network has to periodically compute its available battery capacity \(BC\). If \(BC > 0\), then \(N_i\) can participate in the network. When \(BC\) falls below the current energy requirements to relay packets, \(N_i\) will not rebroadcast route request \(RREQ\), thereby making it inactive. \(N_i\) computes its cost as described previously and delays the rebroadcast of the \(RREQ\) message being directly proportional to the cost of forwarding. Thus, nodes with lower cost (having higher residual battery energies) will forward \(RREQ\) quickly. When the destination gets the first \(RREQ\) packet, it must have traveled along the least cost path. Upon receiving the least cost path (via \(RREP\)), the source node will use this path for its traffic. Thus, effectively, the nodes with heavy energy constraints are avoided. Finally, our algorithm makes use of the route cache in a similar fashion as that in Dynamic Source Routing (DSR) [15], making sure that routes involving the least energy drain rate are selected.

### 3.2 Network Model of Node

Moreover, to overcome the selfishness of nodes, we consider the network model such that, each node is assumed to be aware of its own geographic location. Also the nodes communicate with each other without a centralized base station and using a multi-hop architecture. In addition, we assume that each node has a unique identifier and is mounted with an omni-directional radio receiver. The following formula represents the power and energy relationship of the network model [23]. The power consumed in receiving a packet is given by the \((d/do)^n\) model.

\[
P_r (dB) = P_o (dB) - 10n \log \left( \frac{d}{d_o} \right)
\]

Where \(n\) is the path loss exponent and \(d_o\) is the close-in reference distance. In the following, we see how our algorithm is able to select the routes that are both energy efficient and reliable by considering nodes selfishness based on reputation factor.

#### 3.2.1 Node Selfishness Overcoming

To overcome the node selfishness we set the application in the following scenario. In this a parameter, called sympathy, which reflects the level of selfishness/altruism of the nodes. We assume that network nodes may have different behavior, because of their application needs or their physical constraints. The first aspect that we investigate is the tradeoff that exists between the energy expenditure of a node, and the probability that its request for a traffic connection is declined by the network due to the non-cooperation of the relay nodes. If we assume that each user wishes to maximize his throughput, it may be in his best interest to be cooperative and relay traffic for another user. He may do this in the hope that when he attempts to transmit at a later time, the favor will be returned and his connection request will not be blocked. This however, implies that the user will have to spend part of its energy resource for relaying traffic to other users. The second aspect that we explore is the ability of the network to guarantee low energy consumption levels for users who want or need to be ‘selfish’.

In order to model users with different behavior, we associate with each node a parameter called sympathy, taking values in the range \([0, 1]\). This parameter intuitively reflects how willing a node is to relay traffic for other nodes: a value of 0 reflects extreme selfishness, while a value of 1 reflects extreme altruism. The value of sympathy may depend on the energy constraints of the wireless node, on its location in the network area, or on the particular user’s needs.

#### 3.2.2 Algorithm Description

Our algorithm aims at maximizing the MANET lifetime by balancing the load evenly among the nodes in proportion to their energy reserves. Our approach can be applied to any MANET routing protocol when performing route discovery. We will describe S-EOEDR by taking DSR into consideration. Each node \(N_i\) in the network has to periodically compute its available battery capacity \(BC\). If \(BC > 0\), then \(N_i\) can participate in the network. When \(BC\) falls below the current energy requirements to relay packets, \(N_i\) will not rebroadcast route request \(RREQ\), thereby making it inactive. \(N_i\) computes its cost as described previously and delays the rebroadcast of the \(RREQ\) message being directly proportional to the cost of forwarding. Thus, nodes with lower cost (having higher residual battery energies) will forward \(RREQ\) quickly. When the destination gets the first \(RREQ\) packet, it must have traveled along the least cost path. Upon receiving the least cost path (via \(RREP\)), the source node will use this path for its traffic. Thus, effectively, the nodes with heavy energy constraints are avoided. Finally, our algorithm makes use of the route cache in a similar fashion as that in Dynamic Source Routing (DSR) [15], making sure that routes involving the least energy drain rate are selected.

#### 3.2.3 Node Reputation

Each node \(N_i\) in the network maintains a reputation value \(R_i\) for itself and for each of its 1-hop neighbors. During route discovery or data forwarding, \(N_i\) will execute a function to check whether its 1-hop neighbor is correctly executing the function or not. If it finds that the 1-hop neighbor has forwarded the data/request, it will increment the neighborhood reputation value. Similarly, \(N_i\) will increment its reputation value for every correct execution. \(R_i\) is defined as follows:

\[
R_i = \frac{F_i}{T_i} \quad 0 \leq R_i \leq 1
\]

Where \(F_i\) is the number of packets forwarded by \(N_i\) and \(T_i\) is the total number of requests for forwarding packets.

The use of \(R_i\) allows to control the usage of MANET resources. A higher value of \(R_i\) indicates that \(N_i\) is genuine and can be included on the least cost path from source to destination. If \(R_i\) goes below a threshold \(\delta\), then it implies that \(N_i\) has not contributed to the network operations and must be gradually excluded from the network. Thus, in order to incorporate nodes reputation into our approach, the cost metric will be modified as follows:

\[
C'((i,j)) = \frac{C((i,j))}{R_i}
\]

As \(R_i \in [0,1]\), \(C'((i,j)) \geq C((i,j))\). Consequently, our approach will select paths along nodes that have a higher value of reputation number and higher residual battery capacities.

#### 3.2.4 Protocol

During route discovery, each node appends a \(RREQ\) packet, its cost of transmission and reputation number, and rebroadcast the \(RREQ\) packet. On receiving \(RREQ\), the immediate 1-hop neighbor will check to see whether the reputation value it sees in the packet is correct or not. If yes, then it rebroadcasts the packet. Otherwise, it simply drops the packet. Using this method, the nodes attempting to artificially increase their reputation value will be identified by the other nodes. When any intermediate node which is not the target, receives \(RREQ\) packet, it checks in its local cache about the recent existence of such a Request. If yes, the node discards the packet so as to avoid forwarding to
4. Performance Evaluation

In this section, we study the performance of our approach and compare it with current mechanisms. We present below the simulation context and the energy consumption model, before explaining the performance results in static and dynamic environments, as well as the throughput, and the energy variance.

4.1 Context

In the set of our experiments, we used the Dynamic Source Routing (DSR) [15] as the underlying route discovery and maintenance protocol. DSR was chosen because it is simple and commonly admitted to be one of the most efficient routing protocols, especially in the bounded regions. However, as mentioned earlier, our approach does not privilege any specific routing protocol. Please recall that DSR consists of two mechanisms:

- **Route Discovery**: is the mechanism by which a node A wishing to send a packet to a destination B obtains a source route to B. To perform a Route Discovery, the source node A broadcasts a ROUTE REQUEST packet through the network in a controlled manner and is answered by a ROUTE REPLY packet from either the destination node or another node that knows a route to the destination. To reduce the cost of Route Discovery, each node maintains a cache of source routes it has learned or overheard, which it aggressively used to limit the frequency and propagation of ROUTE REQUESTs.

- **Route Maintenance**: is the mechanism by which a packet sender A detects if the network topology has changed such that it can no longer use a known route to the destination B, because two nodes listed in the route have moved out of range of each other. When Route Maintenance indicates a source route is broken, A is notified with a ROUTE ERROR packet. The sender A can then attempt to use any other route to B already in its cache or can invoke Route Discovery again to find a new route.

For our experiments, the DSR has been modified to force the source node to periodically refresh its cache and to trigger a new route discovery every 10 seconds for better reflecting the power condition of all nodes. Figure 3 shows the data packet format of the modified DSR Protocol Data Unit (PDU) composed of the IP header, the DST Header, and the IP Payload. The IP header is the datagram packet made up of two parts:

1. The fixed part is made up of fields (up to 20 bytes)
2. The variable part is made up of options (40 bytes).

The IP payload is a 2 byte field. This value indicates the amount of data being carried out in the communication.

The DSR makes use of special header, called DSR header, carrying control information that can be included in any existing IP packet. This header contains a small fixed-size of 4-octet, followed by a sequence of DSR options. The end of the sequence of DSR options in the DSR header is implied by its total length. To be added to a packet, the DSR header is inserted after the packet's IP header, before any following header such as a traditional (e.g., TCP or UDP) transport layer header. Specifically, the protocol field in the IP header is used to indicate that a DSR header follows the IP header, and the next header field in the DSR header is used to indicate the type of protocol header. The Route Request option in a DSR header is encoded as follows: Source and Target addresses, Hop limit (varied from 1 to 255), Optional data type, Optional Data length, Identification. The source address must be set to the address of the node originating this packet. Intermediate nodes that retransmit the packet to propagate the Route Request must not change this field. Whereas, the destination address must be set to the IP limited broadcast address. The energy information (C_r, RBP/DR) is sent encapsulated into the header of the chosen routing packet.

We used a fixed transmission range of 250 meters assuming that only a few wireless cards can be configured to use multiple power levels. Hence, MTPR behaves exactly like the protocol using min-hop paths, because the shortest path minimizes the total transmission power consumed per packet. In theory, MTPR can reduce the total transmission power consumed per packet, only when all nodes are capable of adjusting their transmission ranges according to the distance between nodes. All mobile nodes were assumed to be equipped with 2 Mbps IEEE 802.11 network interface cards.

We also used the random waypoint mobility model [16] to simulate nodes movement. The energy consumption values were obtained by comparing commercial products with the experimental data reported in [17]. The activity is characterized by two factors: the maximum speed and the pause time. Each node starts moving from its initial position to a random target position selected inside the simulation area. As in [17], the node speed is uniformly distributed between 0 and the maximum speed. When a node reaches its target position, it waits for the pause time, then selects another random target location and moves again. All nodes have assigned equal initial energy values except the source and the destination nodes in the network, so that it is possible to transmit the packet throughout the simulation. Hence, more initial energy is assigned to the source and the destination nodes.

4.2 Energy Consumption Model

According to the specification of the NIC model [17], the energy consumption varies from \( E_{Rx} = 230 \text{mA} \) (in receiving mode) to \( E_{Tx} = 330 \text{mA} \) (in transmitting mode), using \( V = 5.0 \text{V} \) energy supply. The bandwidth is assumed to be \( \beta = 106\text{Hz} \). These values correspond to a 2,400MHz Wave LAN [16] implementation of IEEE 802.11. When a node sends or receives a packet, the network interface cards. The energy consumption values were obtained by comparing commercial products with the experimental data reported in [17]. The activity is characterized by two factors: the maximum speed and the pause time. Each node starts moving from its initial position to a random target position selected inside the simulation area. As in [17], the node speed is uniformly distributed between 0 and the maximum speed. When a node reaches its target position, it waits for the pause time, then selects another random target location and moves again. All nodes have assigned equal initial energy values except the source and the destination nodes in the network, so that it is possible to transmit the packet throughout the simulation. Hence, more initial energy is assigned to the source and the destination nodes.

We also used the random waypoint mobility model [16] to simulate nodes movement. The energy consumption values were obtained by comparing commercial products with the experimental data reported in [17]. The activity is characterized by two factors: the maximum speed and the pause time. Each node starts moving from its initial position to a random target position selected inside the simulation area. As in [17], the node speed is uniformly distributed between 0 and the maximum speed. When a node reaches its target position, it waits for the pause time, then selects another random target location and moves again. All nodes have assigned equal initial energy values except the source and the destination nodes in the network, so that it is possible to transmit the packet throughout the simulation. Hence, more initial energy is assigned to the source and the destination nodes.

4.2 Energy Consumption Model

According to the specification of the NIC model [17], the energy consumption varies from \( E_{Rx} = 230 \text{mA} \) (in receiving mode) to \( E_{Tx} = 330 \text{mA} \) (in transmitting mode), using \( V = 5.0 \text{V} \) energy supply. The bandwidth is assumed to be \( \beta = 106\text{Hz} \). These values correspond to a 2,400MHz Wave LAN [16] implementation of IEEE 802.11. When a node sends or receives a packet, the network interface cards. The energy consumption values were obtained by comparing commercial products with the experimental data reported in [17]. The activity is characterized by two factors: the maximum speed and the pause time. Each node starts moving from its initial position to a random target position selected inside the simulation area. As in [17], the node speed is uniformly distributed between 0 and the maximum speed. When a node reaches its target position, it waits for the pause time, then selects another random target location and moves again. All nodes have assigned equal initial energy values except the source and the destination nodes in the network, so that it is possible to transmit the packet throughout the simulation. Hence, more initial energy is assigned to the source and the destination nodes.
\[ E = (\alpha \times \text{Packet Size}) \times \beta \]

Where \( \alpha \) represents the packet size in terms of bits.

Conventional ad hoc routing protocols, as mentioned earlier, require that all nodes keep listening even if there is no traffic or neighbor nodes are totally redundant for each other. Obviously, this induces energy wasting and significantly reduces the lifetime of the nodes as well as the network. To reduce the energy consumption in mobile devices, there have been some efforts in physical and data link layers as well as in the network layer related to the routing protocol. The physical layer can save energy by adapting transmission power according to the distance between nodes [18]. At the data link layer, the energy conservation can be achieved by adopting the efficient retransmission schemes and making the node turn off (sleep state) when not receiving or transmitting a packet. If an error occurs in a network during transmission in this layer, the data is retransmitted by the Automatic Repeat reQuest (ARQ) technique. Hence, the listen operation becomes energy free, since all the evaluated ad hoc routing protocols will have similar energy consumption due to the node idle time. Finally, note that when a packet is transmitted, a percentage of the consumed energy represents the Radio Frequency (RF) energy. This energy is used for the propagation model in NS-2 (Network Simulator version 2.26) [17] to determine the energy with which the neighbor’s interface nodes will receive the packet, and consequently determine the successful or wrong packet reception. In this simulation, RF values are maintained as 281.8 mw, which corresponds to the RF energy required to model a radio range of 250 meters.

4.3 Simulation Results

This section outlines the results obtained during our simulations conducted using a network consisting of 50 nodes placed in an area of \( 1 \times 1 \) km and starting from a random initial position. We investigate the performance using ns-2 simulator with the CMU wireless extension. The performance of the algorithms has been evaluated in both static and dynamic environments.

4.3.1 Static Environment

In static environment, the behaviors of the MTPR and MDR mechanisms are evaluated when all nodes maintain their initial position throughout the duration of the simulations. We used DSR as our underlying route discovery and maintenance protocol. However, we modified DSR to force the source node to periodically refresh its cache and to trigger a new route recovery for every 10 seconds to better reflecting power condition of all nodes. As the MTPR approach attempts to minimize the total transmission power consumed per packet, regardless of the lifetime of each node, it exhibits longer lifetime of connections despite shorter lifetime of nodes, because, it is able to easily acquire many other alternative routes with enough battery. Whereas the other mechanisms force more nodes to consume energy by using much longer routes. The MDR approach can properly extend the lifetime of nodes and connections by evenly distributing the energy expenditure among nodes by using their residual battery capacity. However, since it allows nodes to accept all connection requests if they temporarily have enough battery regardless of current traffic condition, nodes eventually experience lack of battery and halt. The absence of some particular nodes due to the traffic overload, forces the current connection to attempt to establish a new route. Therefore it suffers from the short lifetime of connections.

When looking at Figure 4, we conclude that, using our approach the number of nodes alive is maximum compared to other protocols. It avoids the over-dissipation of specific nodes by taking into account the current traffic condition and by utilizing the drain rate of the residual battery capacity. As we can see, after 125 seconds, our approach does not provide optimal results due to the cache refresh time of 10 seconds that we fixed in our tests for cache optimization. This is because each node running DSR protocol is equipped with a route cache which maintains the routes that a node is aware about. DSR uses the cache intensively in order to reduce the overhead caused by the route discovery. This cached route may be a suboptimal from the algorithm’s point of view, since it may not reflect actual traffic load and battery rate depletion conditions. But when it is refreshed each source node initiates periodically a new route discovery in order to obtain routes reflecting more accurately the power condition of the nodes. This contributes to a fair distribution of the network load. As the frequency of route refreshing is decreased (i.e. with larger route refresh intervals) the routing scheme losses its sensitivity to the available battery power of the nodes and starts behaving like other scheme. This shows the tradeoff between the network lifetime and simulation time. Hence, depending on the application requirements, the route refresh interval has to be appropriately selected. If we select the more time interval we can get optimum result even after 125 seconds.

4.3.2 Dynamic Environment

Here, the behaviors of the MTPR and MDR mechanisms are evaluated when all nodes are mobile (based on the Random Waypoint Mobility model) throughout the duration of the simulations. When introducing node mobility, the MTPR mechanism allows some particular nodes to halt earlier than in the other protocols, because MTPR agrees to use the shortest paths. On the other hand, MDR distributes the energy spending by alternating the usage of existing paths, if any. MDR seems to use longer routes among a few paths even in the sparse network to balance energy consumption among nodes. As some nodes die over time, the total number of possible routes between the source and destination nodes decreases. Moreover, the nodes movement allows new routes to appear [19]. In MTPR, it is more likely that the nodes over a given path have enough remaining capacity of battery than in the other algorithms. In Figure 5, we can see that the performance of our approach in dynamic environment is higher than with current ones and totally depends on the node mobility. All the protocols show similar performance, particularly because of the limitation of routes available.
As per static environment, the performance of the dynamic approach decreases after 110 seconds due to the cache refresh time of 10 seconds. Here again, if we select more time interval, we can get optimum result even after 110 seconds.

4.4.3 Throughput

The throughput is measured as the ratio of data packets delivered to the destination and the number of data packets sent by the sender. In Figure 6, the throughput of S-EOEDR is better than other algorithms' with respect to varying pause time. We can also see that unity throughput is obtained for the proposed approach when compared to other ones. This slight increase, though difficult is attained due to lower partitions and lower network overheads in our algorithm.

In addition, we carried out the test under various pause time and uniformly distributed speed. We carried out our simulation with different movement pattern generated for different pause times. That is nodes in the simulation move according to a model that we call the random way point model. The movement scenario files used for each simulation are characterized by a pause time. Each node begins the simulation by remaining stationary for pause time seconds. Upon reaching the destination, the node pauses again for pause time seconds, selects another destination, and proceeds. Our simulation run with movement pattern generated for different pause times.

The throughput of all protocols for random waypoint mobility with uniformly distributed speed is shown in Figure 7. In order to explore how the protocols scale as the rate of topology change varies, we changed the maximum node speed from 1m/s to 10 m/s. This shows that all protocols deliver more than 99% of the packets at different end speeds. Hence, the performance of our algorithm does not degrade the performance of the routing mechanism by the introduction of our changes in DSR algorithm.

4.3.4 Energy Variance

The Energy Variance is defined as the variance of the remaining energy levels of the entire network. It is inversely proportional to the uniform energy distribution in a network. It is used to identify the distribution of energy in the network. Figure 8 shows the marginal increase in the energy variance while increasing the number of source-destination pairs. The energy variance of our algorithm is better than that of the MTPR and MDR algorithms.

In Figure 9, we vary the number of nodes from 5 to 40 with respect to mobility. We can easily observe here that the energy variance of our approach is really reduced when we vary the speed.

In Figure 10, we vary the number of nodes from 5 to 40 with respect to the energy variance. All the above simulation results show that there is a uniform drain of energy in the entire network. Hence, the probability of a particular link alone being drained completely is really weak, which leads to the minimization of link failure. Thus, the lifetime of the network is increased and our algorithm improves the energy efficiency of ad hoc networks. From all the above-mentioned results it can be concluded that our approach can properly extend the lifetime of nodes and connections by evenly distributing the energy expenditure among nodes. That is, it avoids the over dissipation of packets through specific nodes by taking into account the current traffic profiles and drain rate of the participating nodes.
5. Conclusion

In this paper, we proposed an extended version of optimal energy drain rate algorithm in order to overcome the shortcoming of the existing power-aware routing algorithms for Mobile Ad-hoc Networks. This approach takes both the available metrics such as remaining battery capacity and the drain rate into consideration to find out whether a node can be a part of an active route or not. Using this algorithm, the lifetime of each node can be extended, while prolonging the duration of each connection. Our approach is also able to obtain a secure and energy-efficient routing protocol by selecting paths along nodes with a higher reputation number and higher residual battery capacities. This ensures that nodes with lower residual energies are not selected on the communication paths and nodes with low reputation values are eventually eliminated from network operations. This leads to increase any MANET lifetime and its paths reliability. It is important to point out that our approach does not necessarily need to be used for routing all the time. Rather, when the network is new (when all nodes are replete with energy resources), shortest-hop routing can be used. However, after a period of time and particularly when energy resources have fallen below a given threshold, nodes routing can be switched to use our algorithm. The set of conducted simulation tests shows, the relevance of our approach in comparison with other routing protocols. In the future, we are planning to study load balancing algorithms, and how to include one of them in our approach. We will also study how the route cache refresh time can be optimized and test it in several real application scenarios.

References


S. Sankara Gomathi is currently an Assistant professor in the Computer Science and Engineering department at Sri Venkateswara College of Engineering, Chennai, India. Also she is pursuing her Ph.D Degree in wireless communications from Anna university, Chennai, India. She received her Master degree in Communication systems in 1997 from Thiagaraja college of Engineering, Madurai Kamaraj University. GOMATHI served as a review committee member for various international Journals and conferences. Her key research areas are Mobile Computing, Mobile security, Wireless sensor networks, Pervasive computing and networking. Currently she is working on energy conservation issues for mobile adhoc networks.

Richard Chbeir received his PhD in Computer Science from the University of INSA FRANCE in 2000. He is the member of IEEE/ACM since 1999.He is currently an Associate Professor at the Computer Science Department of the Bourgogne University, Dijon-France. His research interests include distributed multimedia database management, object-oriented information systems, spatio-temporal applications, indexing methods, security and watermarking, Bioinformatics,and the development and the integration of information systems. Dr. CHBEIR has published in international journals (including IEEE Transactions on SMC, Journal of Methods of Information in medicine, JDIM, etc.), conferences (ACM SAC, Visual, IEEE CIT, FLAIRS, PDCS, etc.) and contributed papers in edited volumes. He has served in the program committees of several international conferences (IEEE SITIS, ACM SAC, IEEE ISSPIT, EuroPar, BBD, etc.). He is currently the Chair of the French Chapter ACM SIGAPP.

S. Krishnamurthi received the DMIT degree from Madras Institute of Technology in 1976, the Master of Engineering from Madras University in 1978 and the Doctorate degree from the Indian Institute of Technology, Madras in 1989. He started the teaching in 1978 and is currently the Professor of Electronics and Communication Engineering at College of Engineering, Guindy, and Anna University. He is a fellow of IETE and a senior member of ISTE, CSI and BMESI. He has organized several short term courses including “Computers in medicine”. He has several publications to his credit in national and international journals. He took keen interest in developing a medical electronics laboratory at Madras Institute of Technology. His research interests include multi-rate and multi-dimensional signal processing, medical electronics, communication and networking.