Research Article
A Hybrid Energy Efficient Protocol for Mobile Ad Hoc Networks

Niranjan Kumar Ray¹ and Ashok Kumar Turuk²

¹Department of Computer Science & Engineering, Silicon Institute of Technology, Bhubaneswar 751024, India
²Department of Computer Science & Engineering, National Institute of Technology, Rourkela 769008, India

Correspondence should be addressed to Niranjan Kumar Ray; niranjan_ray2000@yahoo.co.in

Received 30 November 2015; Accepted 7 April 2016

1. Introduction

Wireless ad hoc networks do not require any preestablished network infrastructure. They are more suitable for deployment in areas where a fixed backbone network infrastructure is infeasible and/or economically nonviable. Mobile ad hoc networks (MANETs) have variety of applications such as emergency communication services, military communication, and environmental monitoring. However, they suffer from many challenges such as unpredictable mobility, restricted battery power, limited bandwidth, multihop routing, and security. Efforts are being made to address the above issues.

As nodes in MANETs are battery operated, energy efficiency is important design criteria for the longevity of the network. If a node runs out of battery, its ability to route the network traffic gets affected which in turn adversely affects the network lifetime. Network lifetime of MANETs can be enhanced by either maximizing the battery power of nodes or minimizing the total power consumption in the network. Though a considerable progress has been made in the battery technologies in recent years, it is incomparable with the progress made in semiconductor technology yet. This difference has created a gap between the amount of energy needed to operate in a wireless environment and the battery capacity that powered the nodes. Hence, it necessitates the requirement of power conservation techniques to enhance the network lifetime. Such techniques can be applied at different layers of protocol stack.

Nodes in MANET usually transmit packets with maximum power. A packet transmitted with maximum power may reach the destination with lesser number of hops but can decrease the channel utilization and the remaining energy of the node to a greater extent. Energy saving can be done at the node level by adjusting the transmission power to a lower level [1–4]. In recent years, many techniques have been proposed to conserve energy in MANETs. Topology control approach is one among them. The primary objective of topology control algorithms is to adjust the network topology by reducing the transmission power at node level and at the same time maintaining the network connectivity. In other words the objective of topology control approach is to remove the energy inefficient links at the node level by reducing the transmission power. The major design goal in most of the topology control protocols is to minimize the maximum power used by a node. Other design goals are to improve the
network performance such as throughput, network lifetime by alleviating contention, and interference in the network. In this paper we have proposed an energy efficient technique, where nodes are put into sleep state depending on their connectivity information, and transmission power is determined from the location of next-hop node. The proposed scheme inherits the merit of both topology control approach and power management approach. A node in this scheme goes to sleep state only when its absence does not create local partition in its neighborhood.

The rest of the paper is organized as follows. In Section 2, few of the power control techniques are briefly discussed. Proposed technique is described in Section 3. In Section 4 simulation results are presented and conclusion is drawn in Section 5.

2. Related Work

Energy conservation strategies for MANET can be broadly classified into two types: (i) power management approach and (ii) topology control approach. In power management approach a node remains in one of the three possible states: (a) active: participating in the network traffic by sending and/or receiving packets, (b) idle: waiting for the traffic, and (c) sleep: switching off its radio transceiver for a particular period and waking up at the end of the period. Among the above three states, sleep state consumes lesser amount of energy. Therefore, power management based protocols put as many nodes as possible into sleep state to save energy. However, they are more prone to network disruption. This is because, as the nodes goes to sleep state, the connectivity may be lost. Few representatives of power management approach are discussed in [5–9].

Topology control approach conserves energy by adaptively adjusting the transmission power while maintaining the network connectivity. Most of the topology control protocols require information like location, direction, neighbor, and so forth, to construct the topology. Location information can be obtained by means of global positioning systems (GPS) or other positioning methods. The directional information can be obtained by using angle-of-arrival (AOA) technique. Location based techniques produce more accurate topology but are associated with higher hardware cost. In order to reduce the hardware cost some of the techniques assume that a subset of nodes are equipped with GPS while other nodes get their location information by exchanging message with GPS enabled nodes. Few topology control protocols are discussed below.

Topology optimization for different topologies and communication pattern is addressed by Santi [10]. Protocols like Geographical Adaptive Fidelity (GAF) [11] and SPAN [12] save energy by maintaining a connected dominating set (CDS). GAF identifies the routing prospective nodes and redundant nodes. Redundant nodes are switched off to save energy. In SPAN, few nodes are selected as coordinator, which stay/awake continuously and perform multihop packet routing. Nodes not selected as coordinator remain in power save mode to conserve energy. SPAN saves energy by putting few nodes into sleep state, while communication takes place through coordinator. Bao and Garcia-Luna-Aceves [13] proposed a topology management scheme called Topology Management by Priority Ordering (TMPO) [13], which is based on CDS and minimum determining set (MDS). TMPO uses two-hop neighborhood information to determine CDS and MDS.

Sahoo et al. [14] proposed a distributed power control protocol that constructs a power saving tree without taking the nodes local information into account. In their proposed approach, network topology is maintained by controlling the transmission power. Topology based on neighborhood information is constructed in Adaptive Neighbor-based Topology Control (ANTC) proposed by Mir and Ko [15], where the transmission range is determined adaptively in order to maintain connectivity. Based on the local connectivity, each node selects a backbone that guarantees a hierarchical topology structure. In the scheme proposed by Sheu et al. [16] transmission power is calculated considering sender maximum power, receiver minimum power, and the signal strength of the received packet. Both LFTC and ANTC do not put the node into sleep state to save energy. Gui and Liu [17] proposed a technique to optimize the delay and energy. They modeled link delays as a function of signal to interference noise ratio of the receiving node and its packet forwarding time in the link. The sum of delay and energy consumption is considered as the weight of an edge, and their objective was to find the minimum weight sum of any edge.

A prediction based data reduction strategy for sensor network is proposed in [18]. In this approach a hierarchical least mean square (HLMS) adaptive filter mechanism is used to maximize the lifetime of sensor nodes. Energy saving is achieved by reducing the actual data transmission between a source-sink pair. Both source and sink predict the value of sensed data; if the error is acceptable then sink can use predicted value. Otherwise the source will transmit the sensed data to sink. HLMC achieves better energy saving. However, in dynamic environment when link failure occurs due to node mobility its energy saving is a major concern. A tree based broadcast algorithm for wireless ad hoc network is proposed in [19]. The protocol uses two-hop neighborhood information through the exchange of Hello message to construct broadcast tree. The node in the broadcast tree assigned channels in a time-oriented way such that the message forwarded in the broadcast tree is collision free. Simulation results suggest that the protocol achieves lower broadcast redundancy and higher reachability.

A distributed energy efficient multicast routing algorithm is proposed in [20]. The protocol maximizes the lifetime of a source based multicast tree by considering both hop count and energy efficient metric. A weight is assigned to the cost metrics to assign the priority. This helps the protocol to use the cost metrics more efficiently.

A topology control technique CoopSink for ad hoc network is proposed in [21]. CoopSink considers cooperative communication (CC) employing decode-and-forward approach. CC is used to reduce energy consumption in the network and also provides efficient route to the destination node. The authors claim that CoopSink improves the network connectivity and has lower energy consumptions compared
to other protocols. X-LMST, an extension of local minimum spanning tree (LMST), is proposed in [22]. Objective of X-MST is to build an energy efficient spanning tree to improve network lifetime. A cost function is proposed to measure the energy efficiency of a node. Network lifetime is enhanced by avoiding the transmission through energy critical nodes.

Two topology control techniques, node-to-cluster topology control (NCTC) and cluster-to-cluster topology control (CCTC), are proposed in [23]. NCTC and CCTC define two different types of links. In both schemes, cooperative links are employed to connect the clusters obtained by the initial topology. Higher energy saving and connectivity are maintained by NCTC and CCTC considering transmitter and receiver cooperation. However, both are based on centralized approach.

The protocols discussed in this section do not put the node to sleep state to save energy.

3. Proposed Work

In this section, we propose a power conservation technique for ad hoc network, called Location Based Topology Control with Sleep Scheduling (LBTC). The proposed technique takes into the account the merits of both topology control and power management approach. It uses variable transmission power, like the topology control approach. A sleep scheduling mechanism is introduced where nodes are put to sleep state like the power management approach. However, a node in the proposed scheme goes to sleep state only when it is satisfied that its absence will not create a local partition in its neighborhood [24]. Section 3.1 describes the network model and notation used. The proposed scheme is presented in Section 3.2 and analyzed in Section 3.3.

3.1. Network Model and Notation. Wireless ad hoc network is modeled as a graph \( G = (V, E) \), where \( V \) represents the set of nodes and \( E \) represents the set of edges. Network consists of \( n \) number of heterogeneous nodes randomly deployed in the network. Each node has a unique identity (ID) and is equipped with omnidirectional antenna. It is assumed that nodes are location aware and can calculate their relative distance to their neighboring nodes. Let \( P_{\text{max}}(u) \) be the maximum transmission power, let \( P_{\text{min}}(u) \) be the minimum transmission power, and let \( P_u \) be the transmission power of a node \( u \in V \). Initially nodes transmit with maximum power \( P_{\text{max}} \). We assumed that transmission power \( P(u) \) can be adjusted between the maximum and minimum value; that is, \( P_{\text{min}}(u) \leq P(u) \leq P_{\text{max}}(u) \). Let \( P_{uv} \) be the minimum transmission power required to communicate between nodes \( u \) and \( v \), which can be computed as \( P_{uv} = D^\beta + C \), where \( D \) is the Euclidean distance between \( u \) and \( v \), \( \beta \) is the path loss exponent, where \( 2 \leq \beta \leq 4 \), and \( C \) is a constant. Let \( G = (V, E) \) be the initial topology of the network and let \( G' = (V, E') \) be the topology, obtained when the transmission power control mechanism is applied at nodes. We have further assumed that links are symmetric and the power control technique is applied within the neighborhood of a node.

3.1.1. Neighbor Set. One-hop neighbor set of node \( u \) denoted as \( N_u \) is defined as the set of nodes that have a direct link to \( u \), when it transmits with power \( P(u) \), \( P_{\text{min}}(u) \leq P(u) \leq P_{\text{max}}(u) \).

Two-hop neighbor set of node \( u \) denoted as \( N_u^2 \) is defined as the set of nodes that have a direct link to the nodes in the set \( N_u \). This is represented as

\[
N_u^2 = \bigcup_{x \in N_u} N_x^1, \tag{1}
\]

The common node between \( u \) and \( v \) is denoted as \( \text{ComNode}_{uv} \). A node is said to be a common node between \( u \) and \( v \) when it lies in the intersection area of nodes \( u \) and \( v \). This is represented as

\[
\text{ComNode}_{uv} = \{i | i \in \{N_u^1 \cap N_v^1\}\}, \tag{2}
\]

where \( N_u^1 \) and \( N_v^1 \) are the one-hop neighbor set of nodes \( u \) and \( v \), respectively.

3.2. Location Based Topology Control with Sleep Scheduling (LBTC). The proposed scheme operates in two phase. First phase is the link selection phase, while the second is the sleep scheduling phase. In the link selection phase a node determines its transmission power. In sleep scheduling phase a node decides whether to go to sleep state or not depending on the present traffic condition and neighborhood connectivity. A node goes to sleep state only when it satisfies that its neighbors are reachable from each other without its active participation.

3.2.1. Link Selection Phase. In this phase a node \( u \) determines its transmission power that is enough to reach any of the nodes in the set \( N_u \). Nodes periodically broadcast a Hello message with \( P_{\text{max}} \) which contains sender ID and location information of the sender. On receiving Hello message a node computes the transmission power required to transmit to the sender of the Hello message and updates its vicinity table which is maintained at each node. The structure of the vicinity table is as follows:

- \( \text{SenID} \).
- \( \text{LocInfo} \).
- \( \text{MinCost} \).
- \( \text{ComNode} \).
- \( \text{DirCost} \).
- \( \text{LinkType} \).

The meaning of each field in the vicinity table is explained as follows.
SenID. It is identity of the node (Node ID), from which Hello message is received.

LocInfo. It is location information of the sender of Hello message.

DirCost. It is link cost from the node to the sender of the Hello message.

MinCost. It is minimum link cost between the current node and the sender. Current node is the node that is updating its vicinity table. Initially the value in this field is set to that of DirCost. This value is updated when there exists common node i between the current node and SenID such that transmission cost from the current node to SenID node through node i has lower total transmission cost than the DirCost.

ComNode. This field records the common node i through which there exists an energy efficient path between the current node and SenID. It is updated when there exists a common node i between u and v such that $P_{ui} + P_{iv} < P_{uv}$.

LinkType. It indicates whether the link between the current node and SenID is direct (one-hop) or indirect (more than one-hop). For direct the entry is zero, else it is one.

Initially, the vicinity table is empty and is updated when a node receives a Hello message from its neighbors. To illustrate the updation of vicinity table we consider Figure 1. Let node X with its current location (91, 61) receive a Hello message (Z, (101, 61)) from node Z, where coordinate (101, 61) is the current location of node Z. Then, the vicinity table of node X is updated as shown in Table 1. Node X computes the DirCost between itself and Z, which is computed to be 101, assuming $\beta = 2$ and $C = 1$. DirCost and MinCost field is set to 101 and the LinkType to zero. ComNode is set to null as no common node between X and Z exists at this stage.

Vicinity table of node X after receiving Hello message from neighbors is shown in Table 2. After gathering information about its neighbor, node X determines whether there exists any ComNode between itself and its neighbor. Nodes Y, N, and O are the ComNode between X and Z, X

<table>
<thead>
<tr>
<th>SenID</th>
<th>LocInfo</th>
<th>MinCost</th>
<th>ComNode</th>
<th>DirCost</th>
<th>LinkType</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z</td>
<td>(101, 61)</td>
<td>101</td>
<td>—</td>
<td>101</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SenID</th>
<th>LocInfo</th>
<th>MinCost</th>
<th>ComNode</th>
<th>DirCost</th>
<th>LinkType</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z</td>
<td>(101, 61)</td>
<td>101</td>
<td>—</td>
<td>101</td>
<td>0</td>
</tr>
<tr>
<td>Y</td>
<td>(94, 58)</td>
<td>19</td>
<td>—</td>
<td>19</td>
<td>0</td>
</tr>
<tr>
<td>M</td>
<td>(86, 70)</td>
<td>98</td>
<td>—</td>
<td>98</td>
<td>0</td>
</tr>
<tr>
<td>N</td>
<td>(89, 66)</td>
<td>30</td>
<td>—</td>
<td>30</td>
<td>0</td>
</tr>
<tr>
<td>P</td>
<td>(85, 51)</td>
<td>137</td>
<td>—</td>
<td>137</td>
<td>0</td>
</tr>
<tr>
<td>O</td>
<td>(90, 55)</td>
<td>38</td>
<td>—</td>
<td>38</td>
<td>0</td>
</tr>
<tr>
<td>Q</td>
<td>(86, 65)</td>
<td>42</td>
<td>—</td>
<td>42</td>
<td>0</td>
</tr>
</tbody>
</table>
and $M$, and $X$ and $P$, respectively. There can be more than one node within the transmission range of a pair of nodes. Only that node is selected as the ComNode through which the cost between the pair of nodes is minimum. For example, nodes $N$, $O$, $P$, and $Q$ are within the transmission range of nodes $X$ and $M$. But node $O$ is selected as the ComNode between $X$ and $M$. After obtaining the ComNode, node $X$ updates the LinkType, ComNode, and MinCost field in its vicinity table. The updated vicinity table is shown in Table 3. From the table it is observed that the cost of transmitting from $X$ to $Z$ through node $Y$ is 78, whereas the direct cost is 101. Similarly, the cost of transmitting from $X$ to $P$ through $O$ is 80, whereas the direct cost is 137.

3.2.2. Sleep Scheduling Phase. In this phase a node decides whether to enter into sleep state or not, depending on the traffic pattern and neighborhood connectivity information. A node remains in one of the following three states: (i) active, (ii) watch, and (iii) sleep. In active state nodes participate in data communication and periodically broadcast a Hello message. After the expiry of Hello message period, $T_H$, nodes enter into a watch state. In watch state, a node decides whether to go to active state or to sleep state. In active and watch state the radio transceiver of a node is turned on, whereas it is turned off in sleep state. A node in sleep state wakes up at the end of sleep time, $T_S$, and enters into watch state. The state transition diagram between states is shown in Figure 3. In

![Figure 2: Resulting topology of node X after link selection phase.](image)

![Figure 3: State transition in LBTC.](image)
the watch state, a node decides whether to go to active state or sleep state. If the node has some pending traffic, it returns to active state; otherwise it executes a sleep eligibility algorithm, to check whether it can enter into the sleep state.

When a node goes to sleep state it may create a local partition, within its neighborhood. For example, a critical node on a path, when goes to sleep state, creates a local partition, within its neighborhood. For example, a critical node on a path, when goes to sleep state, creates a local partition, within its neighborhood. For example, a critical node on a path, when goes to sleep state, creates a local partition, within its neighborhood. For example, a critical node on a path, when goes to sleep state, creates a local partition, within its neighborhood. For example, a critical node on a path, when goes to sleep state, creates a local partition, within its neighborhood. For example, a critical node on a path, when goes to sleep state, creates a local partition, within its neighborhood.

Assuming that $\Delta$ is the maximum degree of a node, the running time of steps (7)–(16) is $O(\Delta^2)$. In steps (17)–(22), node $u$ does the following for each of its $\mathcal{N}(u)$ neighbors:

1. Set $CM[u][v] = CM[v][u] = TRUE$ if $u$ doesn’t have a path of length $\geq k$ to $v$ with a single link.
2. Else, go to active state.
3. End if.

So the running time of steps (17)–(22) is $O(\Delta^2)$. Running time of steps (24)–(26) and (27)–(31) are constant. So total running time of Algorithm 1 is $O(\Delta^3)$.

### 3.3 Analysis

In this subsection, we analyze the network connectivity in the proposed scheme.

**Theorem 1.** If two nodes are connected before the link selection phase, then they are also connected after the link selection phase.

**Proof.** Let $G = (V, E)$ be the network topology before the link selection phase and let $G' = (V, E')$ be the network topology after the link selection phase. We have to show that, for any $u, v \in V$, if there exists a path in $G$, then there also exists a path from $u$ to $v$ in $G'$.

Since $G$ is connected, for any $u, v \in V$, there exists a $k$-path from $u$ to $v$ in $G$. Let the path be $u \rightarrow v$, where $k$ is the number of links in the path from $u$ to $v$. We show by means of induction that there also exists a path from $u$ to $v$; that is, $u \rightarrow v$, with $k' \geq k$.

Let $k = 1$, which means that node $v$ is within the transmission range of node $u$ in $G$. In $G'$, $v$ will be in either one-hop or two-hop neighbor of $v$. Therefore, a path from $u$ to $v$ exists in $G'$.

Let us assume that it is true for $k = n$; that is, there exists a path in $G$ from $u$ to $v$ with $n$ number of links. Therefore, there also exists a path from $u$ to $v$ in $G'$, and the number of links in the path is greater than or equal to $n$.

Let $k = n + 1$; then the path from $u$ to $v$ with $n + 1$ number of links in $G$ is shown in Figure 4.

The path in Figure 4 can be interpreted as two subpaths one from $u$ to $v'$ with $n$ number of links and the other from $v'$ to $v$ with single link in $G$. From the induction hypothesis, we know that if there exists a path from $u$ to $v'$ in $G$ with $n$ links then there exists a path from $u$ to $v' \in G'$. From the basis, we know that if there exists a path from $v'$ to $v$ with single link in $G$ then there exists a path from $v'$ to $v$ in $G'$. Hence there
exists a path from $u$ to $v$ in $G'$. Therefore if $G$ is connected then $G'$ is also connected.

In the sleep scheduling phase each node checks the local connectivity within its neighborhood. Partition in a network means global partition, which is not the same as that of local partition. In the process of maintaining local connectivity there always exists a node redundancy due to false alarm of network partition. Therefore, connectivity is always maintained in the proposed scheme.

4. Simulation and Comparison

We simulated the proposed technique using QualNet 4.5 [25] simulator to evaluate its performance. The parameters considered for simulation are shown in Table 4.
For the comparison, we consider two topology control protocols, LFTC [16] and ANTC [15]. Lifetime of Hello message is set to two sec and Hello message interval to one sec. The following metrics were considered for performance evaluation.

4.1. Throughput. We plot the throughput versus pause time, number of nodes, CBR connection, and packet size in Figures 5(a), 5(b), 5(c), and 5(d), respectively. From Figure 5, it is observed that a higher throughput is attainable in the proposed scheme. Higher throughput is attributed to the increase in longevity of the network due to proper adjustment of transmission power in the link selection phase and energy saving in the sleep scheduling phase.

4.2. Energy Consumption. Energy consumption determines the effectiveness of energy saving scheme. We consider the energy model in [26] for measuring the energy consumption. The plot for energy consumption versus pause time, number of nodes, CBR connection, and packet size is shown in Figures 6(a), 6(b), 6(c), and 6(d), respectively. It is observed from Figure 6 that the proposed LBTC scheme has lower energy consumption. This is because LBTC selects energy efficient links and nodes are put to
4.3. Network Lifetime. The plot for network lifetime versus pause time, number of nodes, CBR connection, and packet size is shown in Figures 7(a), 7(b), 7(c), and 7(d), respectively. It is observed that LBTC gives a superior network lifetime because it is energy efficient.

4.4. Delay. The plot for delay versus pause time, number of nodes, CBR connection, and packet size is shown in Figures 8(a), 8(b), 8(c), and 8(d), respectively. From Figure 8, it is observed that LBTC has higher end-to-end delay as compared to LFTC and ANTC. This is due to increase in hop count as nodes transmit with lower power. This observation is consistent with the published report that the average end-to-end delay increases when the transmission power control mechanism is applied [27].

5. Conclusion

In this paper, we proposed a hybrid energy efficient protocol called LBTC for ad hoc network. The proposed protocol considers the merits of both topology control scheme and

sleep state, if they do not actively participate in ongoing transmission.
power management scheme. It selects energy efficient links, and nodes which do not actively participate in communication are put to sleep state. LBTC is compared with two existing schemes and observed that it is more energy efficient and delivers higher throughput. However, the end-to-end delay is higher. This is because as the transmission power is controlled, packets take more number of hops to travel the destination. In future, we intend to reduce end-to-end delay with enhancing the network lifetime.

**Competing Interests**

The authors declare that they have no competing interests.

**References**


Submit your manuscripts at http://www.hindawi.com