

Energy Efficient Routing With Unreliable Links in Wireless Networks

Xiang-Yang Li	Haiming Chen	Yantai Shu	Xiaowen Chu	Yan-Wei Wu
Dept. of Compt. Sci. Illinois Inst. of Tech. Chicago, IL, USA xli@cs.iit.edu	Inst. of Compt. Tech. Chinese Acad. of Sci. BeiJing, China chenhaiming@ict.ac.cn	Dept. of Compt. Sci. TianJin University TianJin, China ytshu@tju.edu.cn	Dept. of Compt. Sci. Baptist University Hong Kong, China chxw@comp.hkbu.edu.hk	Dept. of Compt. Sci. Illinois Inst. of Tech. Chicago, IL, USA yw24@iit.edu

Abstract—Energy efficient routings and power control techniques in wireless networks have drawn considerable research interests recently. In this paper, we address the problem of energy efficient reliable routing in wireless networks in the presence of unreliable communication links or devices or lossy wireless link layers by integrating the power control techniques into the energy efficient routing. We study both the case when the link layer implements a perfect reliability and the case when the reliability is implemented through the transport layer, *e.g.*, TCP. We study the energy efficient unicast when the links are unreliable. Subsequently, we study how to perform power control (thus, controlling the reliability of each communication link) such that the unicast routings use the least power when the communication links are unreliable. We presented both centralized algorithms and distributed algorithms for all the questions we studied. We conducted extensive simulations to study the power consumption, the end-to-end delay, and the network throughput of our protocols compared with existing protocols.

I. INTRODUCTION

Many routing protocols have been proposed for wireless ad hoc networks recently. In many scenarios, design of wireless protocols are guided by two essential requirements: energy efficiency and resilience to packet losses. Efficiently handling losses in wireless environments, therefore, has significant importance. Even under benign conditions, various factors, like fading, interference, multi-path effects, and collisions, lead to heavy loss rates on wireless links [1]. Due to the end-to-end reliability requirement of many applications, it is necessary to study how such reliability can be guaranteed in an energy efficient way in wireless environments. In this paper, we study how to achieve reliable and energy efficient routing in multi-hop wireless networks where each wireless link and device could be unreliable. We will propose several novel methods (both centralized and distributed) that appropriately handle packet losses by systematically integrating the energy efficient routing, reliability, and power control techniques.

A number of energy efficient routing protocols [2]–[9] have been proposed recently using a variety techniques (dynamic transmission power adjustment, adaptive sleeping, topology control, multi-path routing, directional antennas, etc). The conventional power aware routing protocols did not take into account the reliability of the wireless links. It is often assumed that the wireless links of a wireless network are reliable by

these traditional protocols with certain theoretically proven performance guarantees [7], [9], [10]. This is clearly too optimistic since in practice, the wireless communications are unreliable and often unpredictable. A number of protocols have been proposed recently to remedy the unreliability of the wireless channels such as using multi-path routing [11], [12], building reliable backbone [10], [13], and using energy efficient reliable routing structure [8], [14]. Obviously, one can increase the transmission power to improve the link reliability and consequently reducing the retransmission times potentially. However, this is not free: we consume more power for single transmission. In this paper, we seek the balance of the smaller transmission power and lower link error rate.

The main contributions of this paper are follows. We integrate the energy efficient routing and power assignment into one scheme by considering the link error rate as a certain function of the transmission power. Notice that when the power used to support the communication of every link is given, the expected link error rate could be derived. Thus, the path with the minimum expected power consumption connecting any two nodes can then be found [8]. When the transmission power changes, the found shortest path will likely change also. Given a fixed source node s (or destination node t) we propose algorithms to find the optimal power assignment for every link such that the expected¹ power consumption of the unicast from the source node s to every other node in the network is the minimum among all possible power assignment. The expected energy consumption depends on the power assignment to all links; on the other hand, the optimal power assignment needs the algorithm to find the path with the minimum expected power consumption. It is then sort of *chicken-and-egg* problem. We consider two different scenarios: either the link layer reliability or the transport layer reliability is implemented. Notice that, in practice, a certain link layer reliability is already implemented in the MAC layer. Our second contribution is the study of integrated power assignment and energy efficient routing using multi-path routing techniques. We conduct extensive simulations to study the performance of our protocols. Our simulations

¹Since the links are unreliable (could be broken with certain probability), the energy consumption of a unicast is a random variable.

show that our protocols significantly reduce the expected energy consumption of routing. The main differences of our result with the result recently presented at [8] are as follows (1) we integrate the power assignment and energy efficient routing; (2) we perform a more realistic simulation to study the performance of our protocols and the simulations show a significant improvement over previous method in terms both expected energy consumption, and network throughput.

The rest of the paper is organized as follows. In Section II, we present our network model, present the problems to be studied, show how to compute the expected energy consumption of a path under unreliable link model, and review the related works. In Section III we present our centralized methods and distributed methods that integrate the power assignment, energy efficient routing, and reliability. In Section IV, we study the minimum energy reliable routing using multi-paths by presenting an efficient method that finds the optimal solution. We report our simulation results that compare the performance of our methods with existing routing methods in Section V. We conclude our paper in Section VI.

II. PRELIMINARIES AND NETWORK MODEL

A. Network Model

We assume that there is a set V of $n = |V|$ of wireless devices (called nodes hereafter) distributed in a region. Each node is assigned a unique ID $i \in [1, n]$. Additionally, each node i has a maximum transmission power $\mathbb{P}(i)$. The multi-hop wireless network is then modelled by a directed communication graph $G = (V, E)$, where E is the set of $m = |E|$ directed links and a directed link (u, v) belongs to E if and only if node v can receive the signal sent from u directly when u transmits at a power $\mathbb{P}(u)$.

For a specific task, we need to assign the power to each wireless node (or link) such that the induced networks can meet the requirement of this task. For example of unicast from source node s to a target node t , we assign a power to all wireless links. Let $\mathbf{p}(u, v)$ denote the power assigned to node u to transmit signal from u to v . We always assume that this power can maintain a reasonably good communication link quality² from node u to node v . This power $\mathbf{p}(u, v)$ could be fixed throughout the network operations if no power control techniques are employed, or it could be changed dynamically when it is needed by the power control techniques or to ensure energy efficient routing. It is well-known that the wireless propagation suffers severe attenuation. Let $\|uv\|$ denote the Euclidean distance between two wireless nodes u and v . If node u transmits at a power $P_t(u)$, the power of the signal received at a node v is assumed to be $P_r(v) = \frac{P_t(u)}{g(u, v)}$, where $g(u, v)$ is the wireless gain between node u and v . It is commonly assumed in the literature that we can always correctly decode the signal when the received power $P_r(v)$ satisfies that $P_r(v) \geq \beta_0 \cdot N_0$, where β_0 is the required minimum *signal-to-interference-noise ratio* (SINR) and N_0

²In practice, it often means that the link error probability is not larger than a certain threshold.

is the strength of the ambient noise. Here the constant β_0 is technology dependent. Thus, by assuming that the node u transmits at power $P_r(u) \geq \beta_0 \cdot N_0 \cdot g(u, v)$, it is assumed in the literature that we can guarantee that node v will receive the signal correctly. In practice, this is not the case though. When a node u transmits at a power p to another node v , the link (u, v) has a packet error probability $\mathcal{E}_{u, v}(p)$ dependent on the transmission power p . Notice that the packet error probability also depends on other factors, such as the environment, the digital modulation techniques and so on. Since the power is the only factor we will control, we assume that this link error probability $\mathcal{E}_{u, v}(p)$ (which is derived from the bit-error-rate BER) only depends on the transmission power p for a specific pair of nodes by assuming all other factors are fixed. For convenience, we use $\mathcal{E}_{u, v}(\mathbf{p})$ to denote the link error probability $\mathcal{E}_{u, v}(\mathbf{p}(u, v))$ of a link (u, v) when the link power is assigned by a method \mathbf{p} .

We also assume that for each node u , there is a node error probability \mathcal{E}_u such that when node u is asked to relay a certain message, it may make a mistake (such as dropping the packets) with probability \mathcal{E}_u . This could happen due to many reasons such as the congestion, queue-buffer overflow, nodes' movement, nodes' sleep, or a sheer failure. Notice that the node error probability can be integrated into the link error probability as follows. For every link (u, v) , we define a new link error probability as $\tilde{\mathcal{E}}_{u, v}(\mathbf{p}) = 1 - (1 - \mathcal{E}_{u, v}(\mathbf{p})) \cdot (1 - \mathcal{E}_v) = \mathcal{E}_{u, v}(\mathbf{p}) + \mathcal{E}_v - \mathcal{E}_{u, v}(\mathbf{p}) \cdot \mathcal{E}_v$. In other words, when the receiving node v makes an error (thus it cannot forward the data further), it is equivalent to say that node v did not get the data at all due to the error by link (u, v) . Consequently, for the remaining of the paper, we always assume that the node will not have error by integrating its error to the in-coming links.

Obviously, as long as there is some link in the multi-hop path that cannot guarantee reliable packet delivery, we will have to rely on TCP-like transport protocols to initiate end-to-end retransmissions starting from the source if end-to-end reliability is required. Assume that we want to implement a reliable communication from the source node s to a target node t . We further assume that a simple path $v_{i_1} v_{i_2} \cdots v_{i_h}$ is used for routing where $s = v_{i_1}$, $t = v_{i_h}$ and direct links $v_{i_j} v_{i_{j+1}}$, $1 \leq j \leq h - 1$, belong to G . There are two possible approaches to implement a reliable communication in practice:

- 1) **Reliable link layer** if the transmission from a node v_{i_j} to node $v_{i_{j+1}}$ is not successful, node v_{i_j} will resend the data till node $v_{i_{j+1}}$ successfully receives the data.
- 2) **Reliable transport layer** if the transmission from a node v_{i_j} to node $v_{i_{j+1}}$ is not successful, node v_{i_j} will discard the data and thus the source node s will start the retransmission due to the time-out signal.

The reliable minimum energy efficient unicast routing problem (abbreviated as **MEER**) is, given the power $\mathbf{p}(u, v)$ assigned to each link (u, v) and the corresponding link error probability $\mathcal{E}_{u, v}(\mathbf{p})$, to find a route from the source node to the receiver such that the *expected* total energy used by all wireless nodes is minimized when either reliable link layer or reliable transport layer is implemented. This has been studied

recently in [14] for reliable link layer and in [8] for reliable transport layer.

In this paper, we will study the following problems.

a) *Power Assignment for Unicast*: Obviously, the final path found depends on the power $\mathbf{p}(u, v)$ used by link (u, v) . Then the problem of power control for reliable energy efficient unicast is to find a power assignment $\mathbf{p}(u, v)$ for each link (u, v) such that the minimum energy efficient reliable route from the source node to the receiver consumes the least *expected* energy among all possible power assignments. In this paper, we will first study the problem of finding a power assignment for every link and the corresponding path between the source and the target node with the minimum expected energy consumption, when either a reliable link layer is implemented³ or a reliable transport layer is implemented. Formally speaking, we consider the following problem.

Instance: A directed graph $G = (V, E)$ with link error probability $\mathcal{E}_{u,v}(\mathbf{p}) \in [0, 1)$ that is function of transmitting power $\mathbf{p}(u, v)$. A value $\kappa(u, v)$ specifies the maximum number of retransmissions implemented at the MAC layer by node u for every link (u, v) . Typically this value is 7 for 802.11. It is set to ∞ if no such bound is set at the MAC layer. We are also given a pair of fixed source node s and target node t .

Question: Find a power $\mathbf{p}^*(u, v)$ for each link (u, v) such that the minimum expected energy path connecting s and t consumes the least power among all possible power assignments. There are two scenarios here: either link layer reliability or transport layer reliability is implemented.

b) *Power Assignment for Single Sink Unicasts*: A power assignment that will produce the most energy efficient routing for a specific pair of source and target nodes does not mean that it will also produce the most energy efficient routing for all pairs of nodes. It is easy to show that no a single power assignment will consistently produce the most energy efficient unicast for *all* pairs of source and target nodes when the reliable transport layer is to be implemented. On the other hand, when a set of unicasts have the same target node (or equivalently have the same source node), we will show that we can find a *unique* power assignment such that it will produce the most energy efficient routing for all such unicasts. Formally, we consider the following problem.

Instance: A directed graph $G = (V, E)$ with link error probability $\mathcal{E}_{u,v}(\mathbf{p}) \in [0, 1)$ that is function of transmitting power $\mathbf{p}(u, v)$. A value $\kappa(u, v)$ specifies the maximum number of retransmissions for every link (u, v) . Fixed source node s .

Question: Find $\mathbf{p}^*(u, v)$ for each link (u, v) such that the minimum expected energy path connecting s and any node t consumes the least power among all possible assignments.

c) *Energy Efficient Multi-path Unicast*: Multi-path routing has been proposed to improve the reliability or the network throughput [6], [12], [15]. However, none of these specifically studied the minimum energy multi-path routing in unreliable environment. Simple heuristics were given in [8] for minimum

energy unicast using multi-paths. In this paper, given source node s and target node t and a parameter k , we will present a polynomial time method to find disjoint k -paths connecting s and t such that the expected total energy is minimized. Specifically, we will consider the following two problems (routing and power assignment):

Instance: A directed graph $G = (V, E)$ with link error probability $\mathcal{E}_{u,v}(\mathbf{p}) \in [0, 1)$ that is function of transmitting power $\mathbf{p}(u, v)$. A given power assignment $\mathbf{p}(u, v)$ for every link (u, v) in the network. A value $\kappa(u, v)$ specifies the maximum number of retransmissions for every link (u, v) . Specified source node s and target node t . An integer k specifies the number of disjoint paths required from s to t .

Question: Find k node disjoint paths connecting s and t such that the total expected energy consumption is minimized.

d) *Power Assignment for Multi-path Unicast*: We then seek the optimum power assignment that results in the minimum power consumption for multi-path unicast routing.

Instance: A directed graph $G = (V, E)$ with link error probability $\mathcal{E}_{u,v}(\mathbf{p}) \in [0, 1)$ that is function of transmitting power $\mathbf{p}(u, v)$. A value $\kappa(u, v)$ specifies the maximum number of retransmissions for every link (u, v) . Specified source node s and target node t . An integer k specifies the number of disjoint paths from s to t .

Question: Find a power $\mathbf{p}^*(u, v)$ for each link (u, v) such that the minimum expected energy k -node disjoint paths connecting s and target node t consumes the least power among all possible power assignments.

B. Compute the Expected Energy Consumption of a Path

Given a simple path $\Pi = v_{i_1} v_{i_2} \cdots v_{i_h}$ connecting s and t , where $s = v_{i_1}$, $t = v_{i_h}$, we briefly show how to compute the *expected* energy consumption of this path under both models.

When a link-layer reliability is implemented, obviously, the *expected* power consumption of path Π with link-layer reliability is $\mathcal{P}_l(\Pi) = \sum_{j=1}^{h-1} \frac{1}{1 - \mathcal{E}_{v_{i_j}, v_{i_{j+1}}}(\mathbf{p})} \cdot \mathbf{p}(v_{i_j}, v_{i_{j+1}})$. Here $\frac{1}{1 - \mathcal{E}_{v_{i_j}, v_{i_{j+1}}}(\mathbf{p})}$ is the *expected* number of total retransmissions of link $(v_{i_j}, v_{i_{j+1}})$ including the initial transmission.

When a transport-layer reliability is implemented, let $\Pi|_{i_j}$ be the subpath of Π from node $s = v_{i_1}$ to node v_{i_j} . The *expected* power consumption of path Π under transport-layer reliability model is then $\mathcal{P}_t(\Pi) = \frac{\mathcal{P}_t(\Pi|_{i_{h-1}}) + \mathbf{p}(v_{i_{h-1}}, v_{i_h})}{1 - \mathcal{E}_{v_{i_{h-1}}, v_{i_h}}(\mathbf{p})}$
 $= \sum_{j=2}^h \frac{\mathbf{p}(v_{i_{j-1}}, v_{i_j})}{\prod_{t=j}^h (1 - \mathcal{E}_{v_{i_{t-1}}, v_{i_t}}(\mathbf{p}))}$.

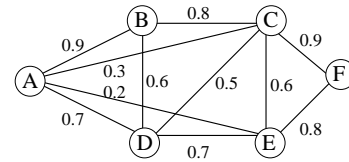


Fig. 1. Example of 6 nodes network with link successful delivery probabilities shown along the edges of the graph.

Let us see an example of computing the expected energy consumption of a path. Figure 1 illustrates a network of 6

³The corresponding problems are then called PAMEEL and PAMEET respectively.

nodes where the link successful delivery probabilities are shown along the edges. Assume that the node power by all nodes are equal, denoted as 1 unit here. When link layer reliability is implemented, the energy efficient path from A to F is then $ABCF$ and total cost is $\frac{1}{0.9} + \frac{1}{0.8} + \frac{1}{0.9} = 3.47$. When the reliable transport-layer is implemented, for the same path $ABCF$, its expected energy cost becomes $\frac{1}{0.9 \cdot 0.8 \cdot 0.9} + \frac{1}{0.8 \cdot 0.9} + \frac{1}{0.9} = 4.04$.

C. Related Work

Routing In Reliable Link Layer Implementation: For single unicast problem, we assume that there is a source node s and a target node t . The approach of implementing reliable link layer has been studied in [8]. For a link (u, v) , it is easy to see that the expected total power needed until there is one successful transmission from u to v is $\mathcal{P}(u, v) = \frac{\mathbf{p}(u, v)}{1 - \mathcal{E}_{u, v}(\mathbf{p})}$. Thus, to find an energy efficient reliable path from s to t is equivalent to find the lowest cost path from s to t in a link weighted network $G = (V, E, \mathcal{P})$ where the weight for each link (u, v) is the expected power $\mathcal{P}(u, v)$ needed for one successful transmission. This clearly can be directly solved by Dijkstra's algorithm [16] in a centralized manner and Bellman-Ford algorithm [16] in a distributed manner.

Routing In Reliable End-To-End Implementation: When the reliable transport layer is used instead, authors of [8] mainly studied the energy efficient reliable routing when the power used by each link is already *fixed*. For completeness of presentation, we briefly reviewed their method here using our own word (illustrated by Algorithm 1). Assume that the simple path $v_{i_1} v_{i_2} \cdots v_{i_{h-1}} v_{i_h}$ is the least cost path where $s = v_{i_1}$, $t = v_{i_h}$. A key observation is that the path $v_{i_1} v_{i_2} \cdots v_{i_{h-1}}$ also consumes the least expected total energy from $s = v_{i_1}$ to node $v_{i_{h-1}}$. Then an algorithm similar to Dijkstra's shortest path algorithm can be used to find the path with the least expected total energy [14]. Let $\mathcal{P}(u)$ be the expected minimum power needed from the source node s to a node u in the network. Obviously, $\mathcal{P}(s) = 0$ and the following algorithm to find the shortest path tree is straightforward. Here $F(u)$ denotes the parent node of u in the shortest path tree rooted at the source node s . It is easy to prove that whenever a node u is added to the set \mathcal{S} , the path defined by the transversal of nodes $u \rightarrow F(u) \rightarrow F(F(u)) \rightarrow \cdots \rightarrow s$ indeed has the minimum expected energy.

III. RELIABLE UNICAST: POWER ASSIGNMENT AND ROUTING

A. Reliable Link Layer Implementation

For convenience, let $\mathcal{P}_{\mathbf{p}}(s, t)$ denote the minimum expected power from node s to node t when the power of each link (u, v) is assigned by \mathbf{p} . We first study how to dynamically adjust the transmission power of each link (u, v) such that the expected power $\mathcal{P}_{\mathbf{p}}(s, t)$ is minimized among all possible power assignment method \mathbf{p} . Assume that the power assignment \mathbf{p}^* produces the optimum answer and the simple path $v_{i_1} v_{i_2} \cdots v_{i_h}$ is the least cost path where $s = v_{i_1}$, $t = v_{i_h}$.

Algorithm 1 Centralized Minimum Expected Energy Reliable Transport-layer Routing cMEET($G, s, \mathbf{p}, \mathcal{E}, F()$, \mathcal{P})

```

1: for every node  $v \in V$  do
2:    $F(v) \leftarrow \emptyset$ , and  $\mathcal{P}(v) = \infty$ .
3:  $\mathcal{P}(s) \leftarrow 0$ ,  $\mathcal{S} \leftarrow \{s\}$ , and  $u \leftarrow s$ .
4: while  $\mathcal{S} \neq V$  do
5:    $temp \leftarrow \infty$ ;
6:   for each node  $v \notin \mathcal{S}$  do
7:     if  $\frac{\mathcal{P}(u) + \mathbf{p}(u, v)}{1 - \mathcal{E}_{u, v}(\mathbf{p})} < \mathcal{P}(v)$  then
8:        $F(v) \leftarrow u$ , and  $\mathcal{P}(v) \leftarrow \frac{\mathcal{P}(u) + \mathbf{p}(u, v)}{1 - \mathcal{E}_{u, v}(\mathbf{p})}$ ;
9:     if  $\mathcal{P}(v) < temp$  then
10:       $temp \leftarrow \mathcal{P}(v)$ , and  $u' \leftarrow v$ ;
11:    $u \leftarrow u'$ , and  $\mathcal{S} \leftarrow \mathcal{S} \cup \{u\}$ ;

```

Obviously, $\mathcal{P}_{\mathbf{p}^*}(s, t) = \sum_{j=1}^{h-1} \frac{\mathbf{p}^*(v_{i_j}, v_{i_{j+1}})}{1 - \mathcal{E}_{u, v}(\mathbf{p}^*)}$. Consequently, to find the optimum power assignment \mathbf{p}^* , it is equivalent to find a power assignment for each link (u, v) such that $\frac{\mathbf{p}(u, v)}{1 - \mathcal{E}_{u, v}(\mathbf{p})}$ is minimized by intelligently choosing \mathbf{p} . This can clearly be solved optimally for each link based on Algorithm 2.

Algorithm 2 Centralized Power Assignment & Min Energy Reliable Link-layer Routing cPAMEEL($G, s, \mathcal{E}, \mathbf{p}^*, F()$, \mathcal{P})

```

1: for every node  $v \in V$  do
2:    $F(v) \leftarrow \emptyset$ , and  $\mathcal{P}(v) = \infty$ .
3:  $\mathcal{P}(s) \leftarrow 0$ ,  $\mathcal{S} \leftarrow \{s\}$ , and  $u \leftarrow s$ .
4: while  $\mathcal{S} \neq V$  do
5:    $temp \leftarrow \infty$ ;
6:   for each node  $v \notin \mathcal{S}$  do
7:     Find the power  $\mathbf{p}^*(u, v)$  minimizing  $\frac{\mathbf{p}(u, v)}{1 - \mathcal{E}_{u, v}(\mathbf{p})}$  among
      all power assignments  $\mathbf{p}(u, v)$  for link  $(u, v)$ .
8:     if  $\mathcal{P}(u) + \frac{\mathbf{p}^*(u, v)}{1 - \mathcal{E}_{u, v}(\mathbf{p}^*)} < \mathcal{P}(v)$  then
9:        $F(v) \leftarrow u$ , and  $\mathcal{P}(v) \leftarrow \mathcal{P}(u) + \frac{\mathbf{p}^*(u, v)}{1 - \mathcal{E}_{u, v}(\mathbf{p}^*)}$ ;
10:    if  $\mathcal{P}(v) < temp$  then
11:      $temp \leftarrow \mathcal{P}(v)$ , and  $u' \leftarrow v$ ;
12:    $u \leftarrow u'$ , and  $\mathcal{S} \leftarrow \mathcal{S} \cup \{u\}$ ;

```

Similarly, we can design a distributed method that is similar to Bellman-Ford [16] to find the optimum power assignment. The detail is omitted here.

B. Reliable End-To-End Implementation

We are now ready to study how to assign an optimum power $\mathbf{p}^*(u, v)$ to every link (u, v) such that the expected energy consumption is minimized among all possible power assignments for all links. Observe that the least cost path of a routing depends on the power \mathbf{p} assigned to each link (u, v) , while on the other hand, to find the optimum power assignment \mathbf{p}^* , we need to compute the least cost path from the source to the target under the optimum power assignment. In the following, we will present a novel approach to break this dependence cycle. Assume for the moment that we already have an optimum power assignment \mathbf{p}^* . Consider the path $v_{i_1} v_{i_2} \cdots v_{i_{h-1}} v_{i_h}$ from s to t with the minimum expected

total energy, where $s = v_{i_1}$, $t = v_{i_n}$. Notice that $\mathcal{P}(s, v_{i_h}) = \frac{\mathcal{P}(s, v_{i_{h-1}}) + \mathbf{p}^*(v_{i_{h-1}}, v_{i_h})}{1 - \mathcal{E}_{v_{i_{h-1}}, v_{i_h}}(\mathbf{p}^*)}$. Then we clearly need to select a

power level $\mathbf{p}^*(v_{i_{h-1}}, v_{i_h})$ such that $\frac{\mathcal{P}(s, v_{i_{h-1}}) + \mathbf{p}^*(v_{i_{h-1}}, v_{i_h})}{1 - \mathcal{E}_{v_{i_{h-1}}, v_{i_h}}(\mathbf{p}^*)}$ is minimized when $\mathcal{P}(s, v_{i_{h-1}})$ is known. We thus have the following power assignment algorithm for minimizing the expected energy consumption from a source node s to any given node v . We assume that the link error probability function $\mathcal{E}_{u,v}(p)$ (i.e., its dependence on the transmission power p) is already known for each link (u, v) in the network.

Algorithm 3 Centralize Power Assignment & Min Energy Reliable Transport-layer Routing cPAMEET($G, s, \mathcal{E}, \mathbf{p}^*, F(), \mathcal{P}$)

- 1: **for** every node $v \in V$ **do**
 - 2: $F(v) \leftarrow \emptyset$, and $\mathcal{P}(v) = \infty$.
 - 3: $\mathcal{P}(s) \leftarrow 0$, $\mathcal{S} \leftarrow \{s\}$, and $u \leftarrow s$.
 - 4: **while** $\mathcal{S} \neq V$ **do**
 - 5: $temp \leftarrow \infty$;
 - 6: **for** each node $v \notin \mathcal{S}$ **do**
 - 7: Find the power $\mathbf{p}^*(u, v)$ minimizing $\frac{\mathcal{P}(u) + \mathbf{p}(u, v)}{1 - \mathcal{E}_{u,v}(\mathbf{p})}$ among all power assignments $\mathbf{p}(u, v)$ for link (u, v) .
 - 8: **if** $\frac{\mathcal{P}(u) + \mathbf{p}^*(u, v)}{1 - \mathcal{E}_{u,v}(\mathbf{p}^*)} < \mathcal{P}(v)$ **then**
 - 9: $F(v) \leftarrow u$, and $\mathcal{P}(v) \leftarrow \frac{\mathcal{P}(u) + \mathbf{p}^*(u, v)}{1 - \mathcal{E}_{u,v}(\mathbf{p}^*)}$;
 - 10: **if** $\mathcal{P}(v) < temp$ **then**
 - 11: $temp \leftarrow \mathcal{P}(v)$, and $u' \leftarrow v$;
 - 12: $u \leftarrow u'$, and $\mathcal{S} \leftarrow \mathcal{S} \cup \{u\}$;
-

For Algorithm 3, we then prove the following theorem.

Theorem 1: The power assignment \mathbf{p}^* computed by Algorithm 3 is indeed optimum, and the path tree traversed based on $F()$ indeed gives the shortest path tree rooted at the source node s .

Proof: We prove this by using induction on all nodes in V . Without loss of generality, assume that we add nodes $v_1 = s, v_2, \dots, v_{n-1}, v_n$ to \mathcal{S} in this order. It is easy to show that the link (s, v_2) consumes the least expected energy among all paths connecting s and v_2 . Assume that the statement is true for all nodes v_1, v_2, \dots, v_k , i.e., the path found by Algorithm 3 using the corresponding power assignment consumes the least expected energy among all power assignments. For all other nodes, let u be the node such that its precedent node in the path, which consumes the least expected energy, is some v_i with $1 \leq i \leq k$. Then clearly, the path from s to v_i must consume the least expected power, i.e., $\mathcal{P}(v_i)$. Since the expected power from s to u along the optimal path is $\frac{\mathcal{P}(v_i) + \mathbf{p}(v_i, u)}{1 - \mathcal{E}_{v_i, u}(\mathbf{p})}$, Algorithm 3 indeed finds the correct node u and the correct power assignment \mathbf{p} . ■

Distributed Implementation: We then show how to implement it in a distributed manner. Assume that each node v stores a variable $\mathcal{P}(s, v)$ that denotes the expected power from source node s to node v of the best known path so far. Algorithm 4 then illustrates our distributed method of finding the optimum power assignment and also the route from s to any node v in the network. It is not difficult to prove that this

distributed method will terminate after at most n rounds and it will produce a correct answer.

Algorithm 4 Distributed Power Assignment And Minimum Expected Energy Reliable Transport-layer Routing at node v : dPAMEET($G, s, \mathcal{E}, \mathbf{p}^*, F(), \mathcal{P}$)

- 1: $F(v) \leftarrow \emptyset$, $\mathcal{P}(s, v) = \infty$ and $temp(u, v) \leftarrow \infty$;
 - 2: If $v = s$, then $\mathcal{P}(s, s) \leftarrow 0$ and sends a message to all its out-neighbors informing a new $\mathcal{P}(s, v) = 0$;
 - 3: **while** received a message from incoming neighbor u updating $\mathcal{P}(s, u)$ **do**
 - 4: Find the power $\mathbf{p}^*(u, v)$ minimizing $\frac{\mathcal{P}(s, u) + \mathbf{p}(u, v)}{1 - \mathcal{E}_{u,v}(\mathbf{p})}$ among all power assignments $\mathbf{p}(u, v)$ for link (u, v) .
 - 5: **if** $\frac{\mathcal{P}(s, u) + \mathbf{p}^*(u, v)}{1 - \mathcal{E}_{u,v}(\mathbf{p}^*)} < \mathcal{P}(s, v)$ **then**
 - 6: $F(v) \leftarrow u$, and $\mathcal{P}(s, v) \leftarrow \frac{\mathcal{P}(s, u) + \mathbf{p}^*(u, v)}{1 - \mathcal{E}_{u,v}(\mathbf{p}^*)}$;
 - 7: Node v records $temp(u, v) \leftarrow \mathbf{p}^*(u, v)$;
 - 8: Node v sends a message to its out-neighbors informing its new $\mathcal{P}(s, v)$;
 - 9: Node $F(v)$ is the parent node of v in the minimum energy path tree rooted at s , $temp(u, v)$ is the final optimum power assignment $\mathbf{p}^*(u, v)$;
-

C. Mixed Reliability Implementations

When some links in the wireless networks implement a link layer reliability, the power assignment algorithm should be modified to accommodate this accordingly. The previous algorithms are motivated and designed for the pure end-to-end retransmission model, i.e., assuming the the MAC layer does not provide any retransmission mechanism. Notice that in practice, some links may already provide the link reliability to some extent. A simple modification of the above algorithm will enable it to solve the mixed retransmission model. When a link (u, v) already provides the link layer reliability, we modify the link power and the link error probability as follows: $\tilde{\mathbf{p}}(u, v) \leftarrow \frac{\mathbf{p}(u, v)}{1 - \mathcal{E}_{u,v}(\mathbf{p})}$, and $\tilde{\mathcal{E}}_{u,v}(\tilde{\mathbf{p}}) \leftarrow 0$. When a link (u, v) does not provide the link layer reliability, we simply let $\tilde{\mathbf{p}}(u, v) \leftarrow \mathbf{p}(u, v)$ and $\tilde{\mathcal{E}}_{u,v}(\tilde{\mathbf{p}}) \leftarrow \mathcal{E}_{u,v}(\mathbf{p})$. We can then call cMEET($G, s, \tilde{\mathbf{p}}, \tilde{\mathcal{E}}, F(), \mathcal{P}$) to find the minimum expected energy path from the source node s to all other nodes in V , call cPAMEET($G, s, \tilde{\mathcal{E}}, \tilde{\mathbf{p}}^*, F(), \mathcal{P}$) (or dPAMEET($G, s, \tilde{\mathcal{E}}, \tilde{\mathbf{p}}^*, F(), \mathcal{P}$)) to find the best power assignment. Notice that we will replace \mathbf{p} in all algorithms with $\tilde{\mathbf{p}}$ whenever it is used.

D. Bounded Retransmission Times

In previous discussions of implementing link layer reliability, we assume that a node u will retransmit the frame until it is received by the other end node v regardless the number of existing retransmissions of the frame. In practice, link layer technologies such as the 802.11 MAC protocol typically make a bounded number of retransmission attempts for a lost or corrupted frame. Further losses can be recovered through end-to-end retransmissions. Thus, we generally assume that for each link (u, v) , there is an integer $\kappa(u, v)$ specifies the

maximum number of retransmissions (including the initial transmission) for a lost or corrupted frame. When a link (u, v) does not pose such limit, we simply set $\kappa(u, v) = \infty$. If a link (u, v) does not implement link layer reliability, we can simply set $\kappa(u, v) = 1$. Obviously, we need to design transport-layer retransmission to guarantee the end-to-end reliability. We then modify the link power and the link error probability as follows $\tilde{\mathbf{p}}(u, v) \leftarrow \mathbf{p}(u, v) \cdot \min\{\frac{1}{1-\mathcal{E}_{u,v}(\mathbf{p})}, \kappa(u, v)\}$, and $\tilde{\mathcal{E}}_{u,v}(\tilde{\mathbf{p}}) \leftarrow \mathcal{E}_{u,v}(\mathbf{p})^{\min\{\frac{1}{1-\mathcal{E}_{u,v}(\mathbf{p})}, \kappa(u, v)\}}$. We can then call algorithm cMEET($G, s, \tilde{\mathbf{p}}, \tilde{\mathcal{E}}, F(), \mathcal{P}$) to find the minimum expected energy path from the source node s to all other nodes in V , and algorithm cPAMEET($G, s, \tilde{\mathcal{E}}, \mathbf{p}^*, F(), \mathcal{P}$) to find the optimum power assignment for minimum expected energy routing.

E. Single Sink Multiple Unicasts

It is easy to show that there is no a single power assignment that will consistently produce the most energy efficient unicast for all pairs of source and target nodes. Fortunately, in many application scenarios, the communications often have a common source node or a common target node, *e.g.*, there is a common sink node in the data collection communications in wireless sensor networks. Thus, we study how to set the transmission power for each individual link that is globally applicable for every unicast communication when there are many simultaneous unicasts with the same sink or source. In other words, the single power assignment will produce the unicast paths with the least expected energy consumptions. Our algorithm is exactly same as cPAMEET($G, s, \tilde{\mathcal{E}}, \mathbf{p}^*, F(), \mathcal{P}$) (or dPAMEET($G, s, \tilde{\mathcal{E}}, \mathbf{p}^*, F(), \mathcal{P}$)), where s is the common source node. The proof of the correctness is straightforward and thus is omitted here. Notice that when only the link layer reliability is implemented, Algorithm 2 also gives the optimal power assignment for any set of unicasts. However, when the transport layer reliability is implemented, Algorithm 3 does not necessarily produce the optimal power assignment for an arbitrary set of unicasts.

IV. MULTI-PATHS ROUTING

We study how to find k node-disjoint paths between the source node and the target node with the minimum expected energy. We will present centralized method to solve it optimally. Notice that since the paths are node-disjoint (except the source node and the target node), except the power used by the source node s , the power used by any other node on a path, say u , is used to reach exactly one next-hop node. Thus, if we fix the power level of the source node s as p , then the problem becomes finding k node disjoint paths with minimum total expected link energy consumption when we set the cost of every link (s, v_i) as 0 for link (s, u_i) with $\mathbf{p}(s, u_i) \leq p$. By checking all possible power levels for the source node⁴, we will find the optimum k -node disjoint paths for routing.

⁴There are at most $d_s - k + 1$ power levels to check where d_s is the total out-neighbors of node s .

Algorithm 5 Minimum Expected k -Disjoint Multi-path Reliable Link layer Routing MEEMPL($G, s, tk, \mathbf{p}, \mathcal{E}$)

- 1: Assume that the power levels of source node s to its d neighbors v_1, v_2, \dots, v_d are $p_1 \leq p_2 \leq \dots \leq p_d$, where $d \geq k$. Let $P = \infty$.
 - 2: **for** $i = k$ to d **do**
 - 3: Assume source node s uses power p_i . Node s can communicate with all nodes v_1, v_2, \dots, v_i using power p_i . Let $p'_i = \frac{\mathbf{p}(s, v_i)}{1-\mathcal{E}_{s, v_i}(\mathbf{p})}$.
 - 4: Assume the link cost of each link (s, v_j) , for $1 \leq j \leq i$, is 0. The cost of each other link (u, v) is $\frac{\mathbf{p}(u, v)}{1-\mathcal{E}_{u, v}(\mathbf{p})}$. Find k internal-node disjoint paths Π_i from s to t with the minimum total link costs δ_i .
 - 5: If $\delta_i + p'_i < P$ then $P = \delta_i + p'_i$, $\ell = i$ and the current best k -internal-node disjoint paths Π is Π_i .
 - 6: Source node s transmits at power p_ℓ and the optimum k -disjoint path is Π .
-

Traditionally, when we need send a data from a source node to the target node, often a path is used for routing. As discussed in [8], we could use a more general directed subgraph, say H , rooted at the source node s , has the target node t as its only leaf node. If a node u relays the data from the source, potentially, all its downstream children in H could receive the data. Since the links are unreliable, some (or none) of its downstream children receive the data correctly. Then these nodes receive the data correctly continue to relay the data to their downstream children nodes. When the target node gets the data, it sends an acknowledge message back to the source node. As always, we assume that the ACK is not lost here for the simplicity of analysis. The source node will restart the transmission if no ACK is received. The objective is to find a directed graph H such that the expected power consumption of unicast over H is minimum among all directed graphs rooted at s and having t as its leave node. We leave this as future work to find such structure and its corresponding optimum power assignment.

V. PERFORMANCE EVALUATION

A. Simulation Settings

We conducted extensive simulations to study the performances of the proposed protocols. We use Qualnet 3.7 in RH Linux 9.0 to run our simulations. We adopts TWO-RAY path loss model and Additive White Gaussian Noise (AWGN) model, and the noise factor is 10. The interference is calculated as the sum of all signals on the channel. The physical layer model we adopted in the simulation is PHY802.11b with 2Mbps data rate. Signal reception model is BER based other than SINR threshold.

We implement the distributed power assignment and minimum expected energy routing (dPAMEER) protocol (including dPAMEEL and dPAMEET) and add the implemented routing protocol to Qualnet. Both dPAMEEL and dPAMEET are based on a modified Bellman-Ford, but they do not take

counts of hop as distance from source to destination. They first assign optimal transmission power for each link, and then take the expected power needed for one successful transmission as distance from source to destination. We then evaluate the performance of dPAMEER in a typical scenario (whose performance is omitted due to space limit) and in several random networks. We compare the performance of dPAMEER with existing distributed routing protocols, which include the Bellman-Ford method that does not specifically take the energy efficiency into consideration, and the protocol GAMER described in [8] that considers the unreliability of the wireless links but not the dynamic power assignment. We choose Bellman-Ford as the base protocol to compare. The reason that we do not choose AODV or DSR as the compared protocol is that they are not table-driven, but on-demand routing protocols. Here we implement a modified Bellman-Ford protocol because the traditional Bellman-Ford protocol will not adjust transmission power and it is incomparable with DPAMEER. After Bellman-Ford algorithm is used to find a path with the minimum number of hops from the source node to the target node, we adjust the transmission power of every link (u, v) on the path to the optimum power p that minimizes the expected power consumption over this link (u, v) .

First, we modify the packet structure of Qualnet so that we can store more information in the routing table for DPAMEER to enable dynamic assigning power and using the assigned power for packet transmission in the physical layer. Secondly, we should get the signal propagation information of a link in the period of establishing routing table. It includes transmission power (TxPower) of the source node who sends the routing message, and receiving power (RxPower) of current node who receives the routing message, and SINR (Signal Interference and Noise Ratio). We then attach these information to the received message in the PHY layer and deliver it to dPAMEER which is in the NETWORK layer. Using these information, DPAMEER can compute the optimal transmission power for the link between source node and current node. Notice that all simulation scenarios using 802.11b as MAC protocol provide retransmission mechanism in link layer. We need to use the adjusted link error probability and link transmission power.

Thirdly, we should adjust the transmission power of the data packets (not control packets) to ensure energy efficiency. Because we cannot set the transmission power of the data packet to the optimal in NETWORK layer, we attach the optimal transmission power, which is retrieved from the routing table of DPAMEER, to the data packet to be delivered to PHY layer. PHY layer checks whether the data packet carries the optimal transmission power. If yes, PHY layer then transmits the this packet using the optimal transmission power; otherwise, transmits the data packet with default power. All the broadcast messages are transmitted with default power.

We use CBR to evaluate the performance of DPAMEER and BFL (Bellman-Ford for link layer), while FTP to evaluation performance of DPAMEER and BFT (Bellman-Ford for transport layer), because CBR adopts unreliable UDP as its transport layer while FTP adopts reliable TCP as its transport

layer. The packet sizes of both CBR and FTP are 512 bytes. The start time of them is 10 seconds and all traffics last for 1000 seconds. The interval of CBR is 1 second. The maximum transmission power of all nodes is set as $15.0dBm$. The receiver sensitivity is set as $-89dBm$. The retransmission times for short packets is at most 4 and is at most 7 for long packets in the link layer.

We study the performances of various protocols using the following three metrics.

- 1) **End-to-End Delay**: Time to send a packet from source to destination.
- 2) **Throughput**: Bytes successfully transmitted from source to destination per second.
- 3) **Average path energy consumed per packet**: Average energy consumed to transmit a packet to destination along the path established by different routing protocols.

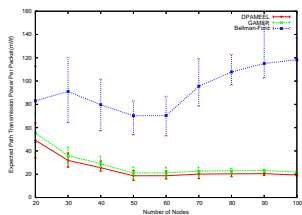
The first two metrics represent the quality of service provided by routing methods, while the third metric represents energy efficiency of routing methods.

For random networks, we randomly generate n nodes, where $n \in [20, 100]$. The coordinates of the wireless nodes are uniformly and randomly (with SEED specified in our configure file) distributed in a square region of 1000 meters by 1000 meters. So the layouts of the nodes can be different with different SEEDs. We repeated 10 simulations with different seeds for each scenario with n nodes placed.

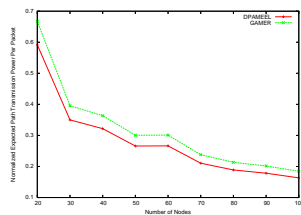
B. Random Networks With Single Traffic

In our first simulation, we study the performance of several routing protocols in random networks when there is only single traffic in the network. Given a network deployment, we first randomly generate the source node and the target node for a traffic. We then run three different routing protocols (modified Bellman-Ford, protocol GAMER proposed in [8], and our dPAMEER protocol) to test their respective performances. To study the performances of various protocols for random networks or different sizes, we always normalize the performance of each protocol by using the performance of modified Bellman-Ford protocol as the denominator. Thus, the performance of modified Bellman-Ford protocol is always treated as one. In our implementations of GAMER protocol, we use a more realistic model: the power of a link (u, v) is set proportional to $\|uv\|^\alpha$ as in [8], but the link error probability is based on the BER table provided in Qualnet instead of being randomly selected in [8]. Here we assume that we know the node's position in implementing GAMER protocol. Notice that we do not use such information in our dPAMEER protocol.

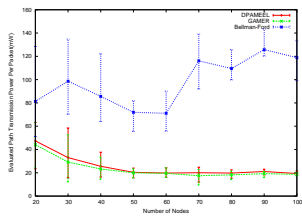
Figure 2 illustrates the energy consumption differences by different routing schemes when only the reliable link layer is implemented and the network only has a single CBR traffic. For a network of n nodes, we run 10 simulations. In each simulation, we randomly generate a flow request. We compute the expected energy consumption of a routing path used by a certain routing method (Bellman-Ford, GAMER, or DPAMEER). These numbers are plotted in Figure 2 (a). The average of the three different flows for different networks of



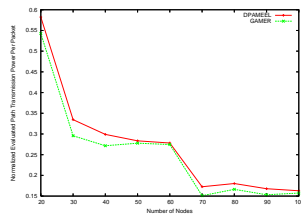
(a) expected energy



(b) normalized expected energy

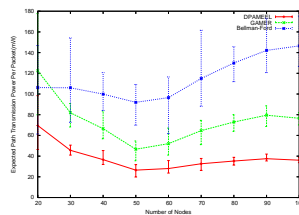


(c) experimental energy

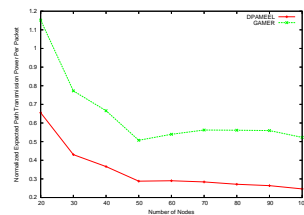


(d) normalized experimental energy

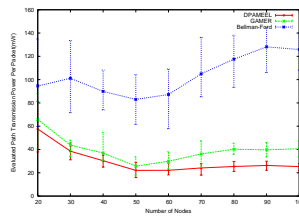
Fig. 2. Comparison of Energy Efficiency with Reliable Link Layer



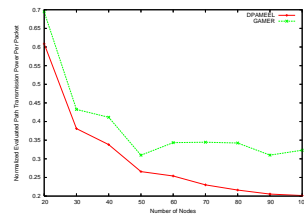
(a) expected energy



(b) normalized expected energy



(c) experimental energy



(d) normalized experimental energy

Fig. 3. Comparison of Energy Efficiency with Reliable Transport Layer.

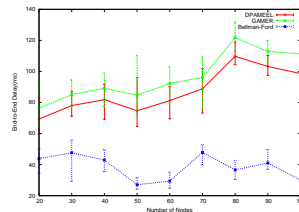
n nodes is plotted in Figure 2 (b). We also actually run the routing based on the path found by various routing protocols and measured the actual power consumption used by routing. The measured data are reported in Figure 2 (c) and the average of the measured data from 10 different simulations is reported in Figure 2 (d).

Clearly, both GAMER protocol [8] and our dPAMEER protocol consume much less energy than the modified Bellman-Ford method. The reducing of energy is more significant when the network becomes dense. This is because both GAMER and dPAMEER protocol tends to use short links, which results in smaller energy consumption, while Bellman-Ford protocol tends to use longer links, which results in large energy consumption due to more retransmissions caused by fragile long links and each transmission uses more power. Performances of DPAMEEL and GAMEER are more stable than Bellman-Ford with different node layouts. Expected performance of DPAMEEL in energy efficiency is obviously better than GAMER. Our protocol saves about 10% power consumption when the network are sparse.

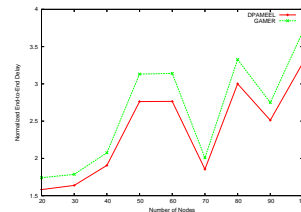
We also conducted simulations to study the performances of different protocols when there is a single FTP traffic. Figure 3 shows a clear advantage of DPAMEEL over GAMER in energy consumption in both simulations and computed theoretical expectation values.

Figure 4 (a) and (b) illustrate the evaluated (and normalized) end-to-end delay of CBR traffics by simulations when only reliable link layer is implemented. As expected, both GAMER and dPAMEER have larger delay than the modified Bellman-Ford protocol since they tend to use short links. The delay degradation becomes more significant when the network density increases. The delay of proposed dPAMEER protocol is about 10% to 20% smaller than that of the GMAER protocol. Figure 4 (c) and (d) illustrate the evaluated (and normalized) network throughput of FTP traffics when reliable transport layer is implemented. Since both GAMER protocol and our dPAMEER protocol use short links, the network

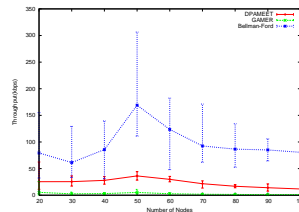
throughputs achieved by these two protocols are smaller than that achieved by the modified Bellman-Ford method. In the worst case, the throughput achieved by the GAMER protocol is only about 5% of that of modified Bellman-Ford method. In this scenario, our protocol achieves a throughput at least twice of the throughput achieved by the GAMER in most networks. The improvement of dPAMEER over GAMER is more significant when the network becomes dense. In summary, DPAMEER has better performance on the end-to-end delay and throughput than GAMER.



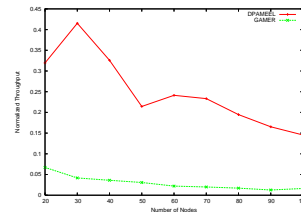
(a) delay



(b) normalized delay



(c) throughput



(d) normalized throughput

Fig. 4. Comparison of End-to-End Delay and Throughput.

C. Random Networks With Multi-Traffics

In our second set of simulations, we study the performance of several routing protocols in random networks when there are several simultaneous traffics in the network. In the results reported later, we run three traffics (CBR and FTP). We run three different routing protocols (modified Bellman-Ford, protocol GAMER proposed in [8], and our dPAMEER protocol) to test

their respective performances. Again, we always normalize the performance of each protocol by using the performance of modified Bellman-Ford protocol as the denominator.

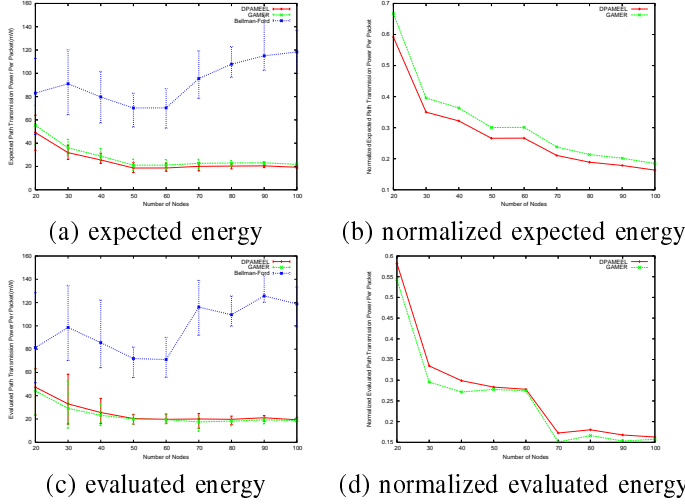


Fig. 5. Comparison of Energy Efficiency with Reliable Link Layer.

Figure 5 illustrates the expected energy consumed for sending one packet from the source node to the target node when only reliable link layer is implemented. Similar to the single traffic case, both GAMER protocol [8] and our dPAMEER protocol consume much less energy than the modified Bellman-Ford method. The reducing of energy is more significant when the network becomes dense. The saving of our protocol compared with the GAMER protocol is not as significant as the single traffic case. The protocol proposed in this paper could save about 10% power consumption when the networks are sparse. Figure 6 shows that both expected and evaluated transmission power of DPAMEET are less than GAMER when only the reliable transport layer is implemented.

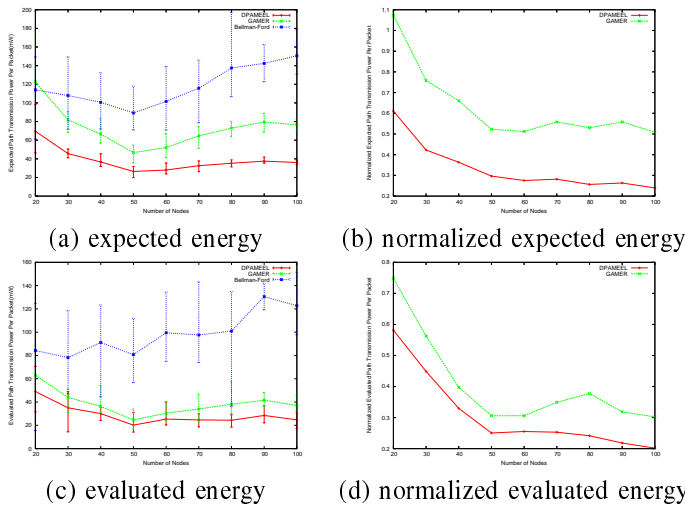


Fig. 6. Comparison of Energy Efficiency with Reliable Transport Layer.

Figure 7 (a) and (b) illustrate the expected end-to-end delay for sending one packet from the source node to the target node

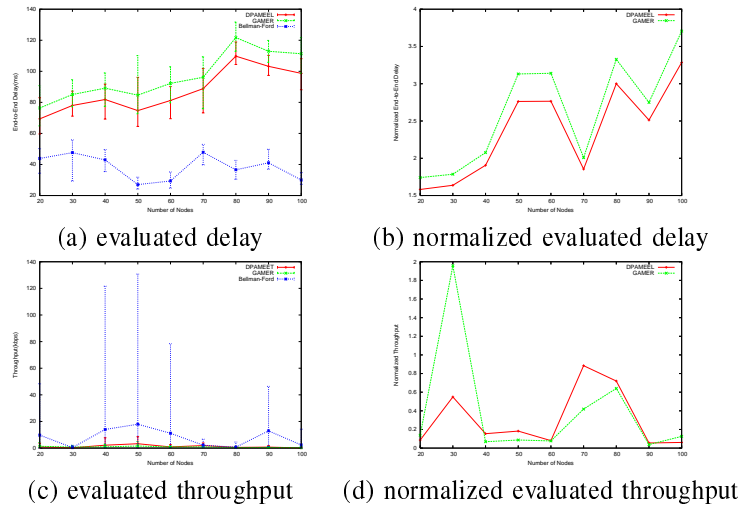


Fig. 7. Comparison of End-to-end Delay and Throughput.

using CBR when only reliable link layer is implemented. As expected, both GAMER and dPAMEER have larger delay than the modified Bellman-Ford protocol since they tend to use short links. The delay degradation becomes more significant when the network density increases. The proposed protocol dPAMEER has smaller delay than the GAMER protocol: its delay is about 15% smaller than that of GAMER protocol. Figure 7 (c) and (d) illustrate the expected network throughput from the source node to the target node using FTP when reliable transport layer is implemented. Since both GAMER protocol and our dPAMEER protocol use short links, the network throughputs achieved by these two protocols are smaller than that achieved by the modified Bellman-Ford method. In the worst case, the throughput achieved by the GAMER protocol is only about 5% of that of modified Bellman-Ford method. Observe that our protocol always achieves a throughput much larger than the previous GAMER protocol. On average, the proposed dPAMEER protocol achieves a throughput about 3 to 4 times of the throughput achieved by GAMER protocol. In one example, the throughput achieved by dPAMEER protocol is about 5 times of the throughput achieved by GAMER protocol.

D. Practical Improvement

When we implement the minimum energy routing, we can do further improvement as follows. When a node u is sending a message to next-hop node v on the minimum expected energy path, the following scenario may happen: node v did not receive it due to link error, but another node w (here node w could be not on the path from s to t) gets the data correctly. Then a question to ask is: “should we stick to resend to node v , or we switch to node w by letting w forward the message instead”. We give a criterion when we should switch, *i.e.*, the node w could start to forward the data now, *i.e.*, the node w could start to forward the data now. Assume that the link layer reliability is implemented. Then node u lets node w to do so when $\mathcal{P}(w, t) < \mathcal{P}(u, t)$. This simple modification will decrease the expected energy consumption

of the path. This is because the retransmission times from node u to node v , which is a geometry distribution, is memoryless: for node v to get the data, the expected number of “new” retransmissions does not depend on the existed retransmissions from u to v . In other words, we still need on average $\frac{1}{1-\varepsilon_{u,v}(\mathbf{p})}$ transmissions to send the message from u to v , although at the moment we know that a number of transmissions already occurred from u to v . If there are multiple such nodes w that got the data from node u , we choose the one with the smallest expected path power consumption to the destination. The detailed implementation will be similar to the ExOR routing in [17] with the following differences. In the approach taken by ExOR protocol [17] a node w will forward the data packet if it has the smallest **ETX** value (expected transmission count) to the destination. In our approach we use the expected total power consumption as the metric instead of ETX to order the neighboring nodes of a sender u . Furthermore, in our approach, we will choose the sender u to resend the data, instead of letting a neighboring node w that received packets from u to relay the data packets for u when the expected path cost from w to destination is higher than that the expected path cost from u to the destination. Let’s illustrate this by an example shown in Figure 8. When node B sends some data with destination F . Assume that in some scenario, *only* node D got the data. Then node D will *not* forward the data for node B since it has a higher expected cost to the destination F . Notice that node D will forward the data if protocol ExOR [17] is used. On the other hand, when node D wants to send data to destination node F . Assume that only node B and A got the data (nodes E and C did not receive it correctly). Then node B will forward the data for node D although it is not on the most energy efficient path from D to F . By adopting this strategy, we can prove that it will save energy compared with sticking to the pre-computed path, *i.e.*, DEF .

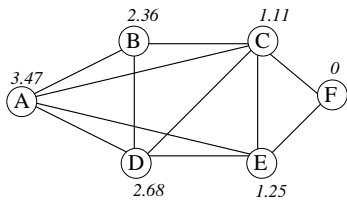


Fig. 8. Expected energy consumption to the destination node F from each node in the sample network from Figure 1. Here we assume that the power used at each node is uniform (thus treated as 1 unit here).

When the reliable transport layer is implemented, it is little bit trickier than the case when the reliable link layer is implemented. We can show that actually in this case we do need to stick to the path computed by Algorithm 1.

VI. CONCLUSION

We proposed several power assignment and routing protocols and performed extensive simulations to study the performance of our unicast routing protocols. When there is only one common source node, we show that our power assignment and routing are optimal. We also presented a multicast routing

protocol whose energy consumption is no more than 2 times of the minimum in a one-to-one communication model.

There are several challenging questions left for further study. First of all, it is an open problem whether we can find a uniform power assignment that is approximately good for all unicasts using reliable transport layer. Secondly, how to find a general structure that supports the power efficient routing by relaxing the disjointness requirement of the disjoint multi-path routing? Thirdly, how to design a power assignment and multicast routing protocol that is efficient?. Last but not the least important, we would like to take the mobility into the consideration of the power assignment: design a power assignment strategy such that it is efficient for a mobile networks, *i.e.*, the power assignment needs to take the possible future movement of neighbors into account.

REFERENCES

- [1] T.S. Rappaport, *Wireless Communications: Principles and Practices*, Prentice Hall, 1996.
- [2] Robin Kravets and P. Krishnan, “Power management techniques for mobile communication,” in *MobileCom*, 1998.
- [3] Volkan Rodoplu and Teresa H. Meng, “Minimum energy mobile wireless networks,” in *IEEE ICC’98*, 1998, vol. 3.
- [4] Jae-Hwan Chang and Leandros Tassiulas, “Energy conserving routing in wireless ad-hoc networks,” in *Infocom*, 2000.
- [5] Benjie Chen, Kyle Jamieson, Hari Balakrishnan, and Robert Morris, “Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks,” in *Mobile Computing and Networking*, 2001, pp. 85–96.
- [6] Anand Srinivas and Eytan Modiano, “Minimum energy disjoint path routing in wireless ad-hoc networks,” in *ACM MobiCom*, 2003, pp. 122–133.
- [7] Peng-Jun Wan, G. Calinescu, Xiang-Yang Li, and Ophir Frieder, “Minimum-energy broadcast routing in static ad hoc wireless networks,” in *IEEE Infocom*, 2001.
- [8] Qunfeng Dong, Suman Banerjee, Micah Adler, and Archan Misra, “Minimum energy reliable paths using unreliable wireless links,” in *ACM MobiHoc*, 2005.
- [9] Xiang-Yang Li, Wen-Zhan Song, and WeiZhao Wang, “A unified energy efficient topology for unicast and broadcast,” .
- [10] Yu Wang, WeiZhao Wang, and Xiang-Yang Li, “Distributed low-cost weighted backbone formation for wireless ad hoc networks,” in *ACM MobiHoc*, 2005.
- [11] J. Raju and J. Garcia-Luna-Aceves, “A new approach to on-demand loop-free multipath routing,” in *Proceedings of the Int’l Conf. on Computer Communications and Networks (IC3N)*, 1999, pp. 522–527.
- [12] A. Nasipuri, R. Castaneda, and S. R. Das, “Performance of multipath routing for on-demand protocols in ad hoc networks,” *ACM/Kluwer MONET*, vol. 6, no. 4, pp. 339–349, 2001.
- [13] Manki Min, Feng Wang, Ding-Zhu Du, and Panos M. Pardalos, “A reliable virtual backbone scheme in mobile ad-hoc networks,” in *1st IEEE MASS*, 2004.
- [14] S. Banerjee and A. Misra, “Minimum energy paths for reliable communication in multi-hop wireless networks,” in *the 3rd ACM MobiHoc’02*, 2002, pp. 146–156.
- [15] S. J. Lee and M. Gerla, “Split multipath routing with maximally disjoint paths in ad hoc networks,” in *IEEE ICC*, 2001, pp. 3201–3205.
- [16] T. J. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, MIT Press and McGraw-Hill, 1990.
- [17] Sanjit Biswas and Robert Morris, “Exor: opportunistic multi-hop routing for wireless networks,” in *SIGCOMM ’05*, 2005, pp. 133–144.