

Efficient Opportunistic Routing in Utility-Based Ad Hoc Networks

Mingming Lu, Feng Li, IEEE Student Member, and Jie Wu, IEEE Senior Member *)

Abstract

Due to resource scarcity, a paramount concern in ad hoc networks is utilizing limited resources efficiently. The self-organized nature of ad hoc networks makes the network utility based approach an efficient way to allocate limited resources. However, the effect of link instability has not yet been adequately addressed in literature. To efficiently address the routing problem in ad hoc networks, we integrate the cost and stability into a network utility metric, and adopt the metric to evaluate the routing optimality in a unified, opportunistic routing model. Based on this model, an efficient algorithm is designed, both centralized and distributed implementations are presented, and extensive simulations on NS-2 are conducted to verify our results.

Index Terms: *Ad hoc networks, distributed algorithms, opportunistic routing, network utility, stability.*

*Mingming Lu, Feng Li, and Jie Wu are with Dept. of Computer Science and Engineering, Florida Atlantic Univ., Boca Raton, FL 33431 USA (email: {mlu2@, jie@cse.}fau.edu)

NOTATION

s	source node
d	destination node
\mathcal{N}	the set of nodes in the network, $\mathcal{N} = \{1, 2, \dots, N\}$
E	the set of links that connect nodes
(i, j)	a link in E
$c_{i,j}$	the link cost of link (i, j)
$p_{i,j}$	the link stability of link (i, j)
v	the benefit value of a packet
U	the expected utility
u_i	the expected utility of node i
δ_i	the maximum expected utility of node i
opu_i	the expected utility under opportunistic routing (OpEU) of node i
Q	the set of nodes whose expected utilities and OpEU have been maximized
N_i	node i 's neighbor set in which nodes are pairwise connected
S_i	node i 's relay candidate set
c_i	the selected transmission cost of node i
P_i	node i 's relay set in a OpEU-based priority list
$P_i(c)$	the relay candidate set of node i that corresponds to cost level c of node i
$opu_i(c)$	node i 's OpEU corresponding to cost c and relay candidate set $P_i(c)$
\vec{c}_i	the set of transmission costs of node i to the nodes in its relay candidate set
γ	power attenuation coefficient in the wireless medium
K	ambient noise
R_j	receiver power at receiver j
p^{ber}	bit-error-rate

1 Introduction

Power scarcity and unstable links are two prominent problems in ad hoc networks. Existing routing protocols [5, 8, 12, 24] address them either separately or simply combinatorially without considering their correlation. However, they are correlated, and they both have a direct influence on the efficient allocation of energy resources over networks. Intuitively, the lowest-cost routing path is the most energy-efficient route, but it may not be the best choice in the event of packet delivery failure because packet loss causes energy depletion rather than energy conservation. Energy conservation is meaningful only if packets are delivered successfully. On the other hand, the most stable route is also not always a good choice because the most stable path can be costly. It is not hard to see that the routing path with the lowest cost and the highest stability is the best choice to allocate power resources, but the lowest cost route and the most stable route are often not the same. A trade-off exists between low cost and high stability.

We adopt utility to reflect this trade-off. We assign each packet a benefit value which reflects the importance of the packet. As in previous works, it is natural to maximize the (expected) utility to be equal to the (expected) benefit minus the (expected) cost. Due to unstable links, the source's utility depends not only on its benefit and the link cost of the selected path, but also on the link stability of the selected path. Therefore, we design an *expected utility*, which integrates the benefit, link cost, and link stability to efficiently allocate energy costs in the network.

The utility model helps the system achieve a good trade-off between lower cost routes and higher stability routes because sources with high benefits prefer higher stability, while low-benefit sources choose low-cost routes. It is like gambling: the high benefit source is less likely to take a risk while the lower benefit source tends to take risks to maximize utility.

To further reduce the effect of link instability, we adopt the opportunistic routing framework introduced in [4]. Opportunistic routing utilizes the redundancy in omnidirectional transmissions to make up for packet loss over lossy links. Intuitively, the wireless channels between a pair of nodes change over time, but if there are multiple receivers for a sender, then the probability of successful delivery is likely to increase.

The opportunistic routing scheme consists of two components: a routing component and a MAC component. The routing component is used to select candidate receivers for each node and determine their priorities, while the MAC component is responsible for identifying one receiver from the candidate receivers based on their priority and actual reception of packets. However, the above cross-layer routing scheme has three shortcomings. First, it does not propose an explicit optimization goal. Second, its MAC component cannot guarantee that only one receiver will forward packets. Third, the transmission range is fixed, which is unreasonable when selecting candidate

receivers.

In this paper, we simplify the above cross-layer routing scheme into a single-layer opportunistic routing scheme, present an explicit optimization goal, design an efficient algorithm that allows for an adjustable transmission range, prove the optimality of our algorithm, and implement the algorithm in both centralized and distributed ways. To simplify our model, we only consider a source-destination pair. Our distributed implementation provides a framework to implement routing components for all on-demand and opportunistic-based routing protocols. In our distributed implementation, only the summarized routing information (the expected network utility) is propagated from the destination to the source. Our scheme is easy to implement - it is based on existing reactive routing protocols and does not introduce additional cost.

The main contributions of this paper can be summarized as follows:

1. We devise a new metric which incorporates link cost, link stability, and benefit, and can evaluate the performance of a single route and the opportunistic routing scheme.
2. We enhance the opportunistic routing model, design an efficient heuristic based on our simplified opportunistic routing model, provide an experimental comparison to the optimal solution, and implement our algorithm in both a centralized and distributed way.
3. Our routing protocol can provide different routings for different values of benefit, i.e., different services for different routing requirements.
4. Our distributed implementation provides a framework for on-demand and opportunistic-based routing protocols.
5. We verify our results through extensive simulations on NS-2.

The following assumptions are used in this paper: 1) Each node has a priori knowledge about its associated link cost and link stability. Numerous works [23, 27] address the issue of how to collect this information. 2) In our model, the transmission range is adjustable but retransmission is not considered.

The remainder of this paper is organized as follows. Section 2 discusses related work. Some preliminaries are presented in Section 3. Section 4 introduces the model and algorithm, proves the optimality of our algorithm, and implements our algorithm in both centralized and distributed approaches. Section 5 empirically evaluates performance. Finally, Section 6 concludes this work and outlines future work.

2 Related Work

Various existing routing protocols [5, 27] pursue the minimum hop count or minimum cost as their optimization goal. As discussed in [8] and [24], these metrics are not necessarily ideal because they do not take link stability into account. Another routing algorithm [22] models the link stability as the longevity of the link and identifies the best route as the route with the longest lifetime. One similar scheme [25] adopts the “best worst link” concept in which the least stable link (the bottleneck link) over any route is identified and the route with the best “bottleneck” is selected. Although these schemes consider the link stability issue, they fail to differentiate between two routes with the same bottleneck link but different link characteristics. *Expected transmission count* [8] and MintRoute [24] adopt path delivery ratio by measuring hop-by-hop link delivery ratio along the path. Expected transmission count weighs links with a metric called *minimum expected transmission number*, which is equal to the link cost divided by the link delivery ratio. However, this metric requires an unrealistic assumption that retransmission is unlimited.

Opportunistic routing (OR) [5, 18, 19] is mainly proposed to address the unreliable communication in multi-hop wireless networks by utilizing the broadcast advantages of wireless communication. Relay selection and prioritization are two key issues of OR protocol design. Some works [11, 18, 31] apply the OR scheme to geographic routing, and select nodes that are geographically closer to the destination to form a relay set. Other OR protocols [5, 6, 10] select any node with smaller cost to the destination as a relay. In the model proposed in this paper, relays do not need to be “closer” to the destination.

Some previous works [7, 13, 21, 26, 29] focus on link layer coordination schemes to ensure that multiple receivers of a packet agree on a relay in a distributed fashion. These schemes select the relay set based on some heuristic methods. Although one recent work [6] demonstrates the performance benefit by integrating network coding, this result is based on a strong assumption: unlimited retransmissions. Before several recent works [6, 28] that proposed metrics that are capable of capturing the expected number of any-path transmissions needed to successfully deliver a packet between two nodes under the OR framework, most existing works [3, 5] prioritized the relays based on the expected number of transmissions along the best path alone.

In our model, we reduce the two-phase framework of ExOR into a single-phase model by replacing the sequential ACK in the MAC layer with a simple neighbor selection strategy. Instead of adopting the metric with little correlation to link stability, we adopt utility [16] to integrate link stability, link cost, and benefit into a single metric to assess the routing optimality.

Recently, there has been a trend in integrating network coding techniques into opportunistic routing [6, 9]. In [6], the authors adopt the random linear coding scheme to allow intermediate nodes

to code the received packets so that no MAC layer acknowledge scheduling is needed. However, controlling of outgoing packets is a big challenge, and the authors [6] fail to provide a convincing solution with an in-depth analysis.

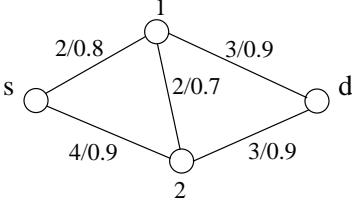
Numerous works adopt utility as the optimization objective. Li, Xue, and Nahrstedt [14] present a price-based scheme to effectively allocate resources among multiple multi-hop flows. Their approach maximizes the aggregated utility of flows (the network utility), while maintaining basic fairness among multiple flows. Qiu and Marbach [17] propose a market-based approach to efficiently allocate bandwidth in wireless ad hoc networks. Although there are numerous existing works applying the network utility related approaches, as far as we know, none of them combine stability and link cost to design an optimization model that will maximize the expected network utility. In our previous work [15], we proposed the basic expected network utility based model. In this paper, we incorporate an opportunistic routing scheme and design an efficient routing scheme. Zhou et al. [30] propose a utility-based routing model in wireless ad hoc networks. Similar to our work, their work also uses utility to reflect the QoS requirement, but they use delay as the QoS measurement, which is estimated through the number of hops. In their work, a trade-off exists between energy cost and transmission delay. The objective of their routing model is to maximize the surplus equal to utility minus cost.

3 Preliminary

3.1 Cost and Stability in a Wireless Medium

Any signal transmitted over a wireless medium experiences two different effects: attenuation due to the medium and interference with ambient noise on the channel. Due to the characteristics of the wireless medium, the transmitted signal suffers an attenuation proportional to $dist_{ij}^\gamma$, where $dist_{i,j}$ is the distance between transmitter i and receiver j and $\gamma = 2$ or 4 . The ambient noise at the receiver is independent of the distance between the transmitter and the receiver, and depends only on the operating conditions at the receiver. In order for the receiver to receive a signal with strength R_j , the sender should transmit the packet at the power $R_j \times dist_{ij}^\gamma$. Since the transmission power dominates the received signal strength at the receiver, we define the transmission power (per bit) as the bit-cost associated with link (i, j) , and the product of the bit-cost and packet size as the corresponding packet-cost. The bit-error-rate associated with link (i, j) is essentially a function of the ratio of R_j to the ambient noise. The packet-error-rate associated with a link is a non-decreasing function of the corresponding bit-error-rate and packet size.

For any directional link (i, j) , in order to correctly receive a packet, the ratio of the received sig-



	u_d	u_1	u_2	U
r_1	20/30	15/24		10*/17.2
r_2	20/30		15/24	9.5/17.6*
r_3	20/30	8.5/14.8	15/24	4.8/9.8
r_4	20/30	15/24	8.5/14.8	3.65/9.3

Figure 1. The figure on the left shows the topology of a simple network where the attributes of each link are labeled in the form of ‘cost/stability’. The expected utilities of the nodes on each route ($r_1 : < s, 1, d >$, $r_2 : < s, 2, d >$, $r_3 : < s, 1, 2, d >$, and $r_4 : < s, 2, 1, d >$) are listed in the table on the right where, in each cell, two values that are separated by ‘/’ represent the expected utilities under benefits of 20 and 30, respectively.

nal power to the ambient noise at receiver j has to be larger than or equal to a predefined threshold value. In order to conserve energy, sender i can set its transmission power so that the above ratio at receiver j is equal to the threshold value. For simplicity, we assume that the energy cost for transmitting a packet from sender i to receiver j consists of the transmission power (consumed at sender i) alone. This assumption is valid because the energy consumed at the receiver is relatively small compared with the energy consumed at the sender.

For simplicity, we do not consider ambient noise in our model, but leave the evaluation of the ambient noise to our experimental study. Therefore, a directional link (i, j) can be regarded as a unidirectional link. Without loss of generality, we assume that all packets are of a constant size. Hence, we can define the stability of a directional link (i, j) (denoted as $p_{i,j}$) as the packet-error-rate associated with the link, and the cost of link (i, j) (denoted as $c_{i,j}$) as a function of transmission power consumed at transmitter i for a single transmission and the packet size.

3.2 Utility-Based Routing

We consider that a source s intends to send packets to a destination d through other nodes in an ad hoc network. For each successfully delivered packet, s will get a benefit v . The network is modeled as a graph (\mathcal{N}, E) , where $\mathcal{N} = \{1, 2, \dots, N\}$ is the set of nodes and E is the set of links.

To illustrate the basic idea of the expected network utility metric, we first consider a single-link route from s to d with link stability $p_{s,d}$ and link cost $c_{s,d}$. If a transmission is successful, s will obtain benefit v , incur cost $c_{s,d}$, and have a utility value of $v - c_{s,d}$. Otherwise, its utility is $0 - c_{s,d}$. Since the success probability is $p_{s,d}$, and the failure probability is $1 - p_{s,d}$, the expected utility is:

$$U = p_{s,d} \cdot (v - c_{s,d}) + (1 - p_{s,d}) \cdot (0 - c_{s,d}) = p_{s,d} \cdot v - c_{s,d}. \quad (1)$$

In general, for a multi-hop path $R, \langle s = 0, 1, 2, \dots, m - 1, d = m \rangle$. The corresponding expected utility is as follows:

$$U = \left(\prod_{j=0}^{m-1} p_{j,j+1} \right) \cdot v - \sum_{i=0}^{m-1} c_{i,i+1} \prod_{j=0}^{i-1} p_{j,j+1} = P_R \cdot v - C_R \quad (2)$$

where P_R is path stability and C_R is path cost. Comparing Formula (2) with Formula (1), we find that the expected utility of a multi-hop path and a single-hop path have a similar form: stability times benefit minus cost. The stability of a multi-hop path is simply the multiplication of stability of each single hop on the path. The cost of a multi-hop path is the sum of the expected costs of each single hop, where the expected cost of each single hop is the cost of that hop times the stability of the subpath from the source to the hop right before the concerned hop. The expected cost is introduced so that intermediate nodes will not consume energy unless they really receive packets from previous hops.

Formula (2) can be derived through Formula (1) in a backward-fashion. For example, in Figure 1, four paths exist: $r_1 : \langle s, 1, d \rangle$, $r_2 : \langle s, 2, d \rangle$, $r_3 : \langle s, 1, 2, d \rangle$, and $r_4 : \langle s, 2, 1, d \rangle$. Each link is labeled with its associated cost/stability. Considering path r_1 , by applying Formula (2), we have $U = 0.8 \cdot 0.9 \cdot 20 - 2 - 3 \cdot 0.8 = 10$. We can also view node 1 as the virtual source and apply Formula (1) to link $(1, d)$: $u_1 = 0.9 \cdot 20 - 3 = 15$. Then, we can apply Formula (1) to link $(s, 1)$ by viewing 1 as the virtual destination: $U = 0.8 \cdot 15 - 2 = 10$.

A desirable property of our metric is that the selection of the optimal route not only depends on the network properties, such as link stability and link cost, but also on the value of the benefit (v). Consider the example in Figure 1. If the benefit $v = 20$, the optimal route is $\langle s, 1, d \rangle$, but if $v = 30$, the optimal route is $\langle s, 2, d \rangle$.

3.3 Opportunistic Routing Schemes

Most existing routing protocols select a fixed route, but in ad hoc networks, links between pairs of nodes tend to change over time. This affects the stability of the predetermined route. The opportunistic routing scheme increases link stability by exploiting the idea that if each node has multiple neighbors waiting to send, it is likely that there will be a stable route at any given time.

The opportunistic routing scheme (ExOR) [5] regulates that each node selects a *relay set* (the set of relay nodes) to forward packets. Because of unstable links, some relay nodes may fail to receive a packet, but if at least one relay node receives the packet, the packet will not be lost. Since it is likely that more than one relay node will receive the packet, ExOR adopts the following scheme to ensure that only one relay node forwards the packet: 1) A node not only selects a relay set but also assigns a priority to each relay node according to some metric. 2) The node puts the IDs of

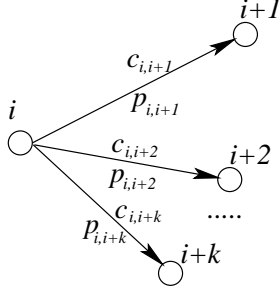


Figure 2. An example of a node in the new model.

the relay nodes into the packet header in the order of their priorities and forwards the packet. 3) Upon receiving the packet, a relay node schedules the timing to forward the packet according to its priority. The higher the priority, the shorter the timeout. 4) Upon timeout, a relay node decides to forward the packet if and only if no higher-priority node has forwarded the packet.

However, ExOR cannot guarantee that no redundant copies of the same packet will be transmitted in the network because not all next-hop nodes of a node can receive the acknowledgement from the next-hop nodes with higher priority. A simple technique can enforce perfect cooperation where only one next-hop node can forward the received packet. We can set the transmission range of the acknowledgement packets to be twice that of the transmission range of the data packets. By doing so, all next-hop nodes of a sender and the sender itself can receive the acknowledgement packets.

4 The Proposed Model and Solutions

4.1 The Model

The metric for opportunistic routing. The *expected utility under opportunistic routing* (OpEU) is different from the expected utility for a single multi-hop route. The calculation of OpEU is based on all nodes in its relay set rather than a single downstream node like the calculation of expected utility. We use opu_i to denote node i 's OpEU. Consider a node i with cost c (determined by its transmission range) and the corresponding relay set in its priority list $P_i = \langle i+1, i+2, \dots, i+k \rangle$, in Figure 2. With probability $p_{i,i+1}$, node $i+1$ receives, and hence forwards the packet. In this event, the OpEU provided by node $i+1$ is $p_{i,i+1} \cdot opu_{i+1}$. If node $i+1$ does not receive the packet, with probability $p_{i,i+2} \cdot (1 - p_{i,i+1})$ node $i+2$ receives, and hence forwards the packet. In this case, the OpEU provided by node $i+2$ is $p_{i,i+2}(1 - p_{i,i+1}) \cdot opu_{i+2}$. By this reasoning, node i 's OpEU

benefit	opu_d	opu_1	opu_2	opu_s
$v = 20$	20	16.1	15	14.7
$v = 30$	30	25.7	24	19.7

Table 1. The OpEUs of nodes in Figure 1 under $v = 20$ and $v = 30$.

can be calculated based on the OpEUs of nodes in its relay set:

$$opu_i = \sum_{j=i+1}^{i+k} (opu_j \cdot p_{i,j} \cdot \prod_{l=i+1}^{j-1} (1 - p_{i,l})) - c \quad (3)$$

To illustrate Formula (3), we consider the example shown in Figure 1. The OpEUs of nodes are shown in Table 1, where we assume that node 2 is node 1's next hop. Comparing Table 1 to the table in Figure 1, we can see that the utility of transmitting a packet with benefit value 20 (30) is increased from 10 (17.6) to 14.7 (19.7) in the sample network.

The relay set. Similar to ExOR [4, 5], in which a node's distance to the destination is calculated in terms of hop count or ETX [8], we adopt expected utility as the metric to evaluate the distance. The higher the expected utility of a node, the closer it is to the destination. To avoid a loop, we include only those neighbors who are "closer" than the node itself to the destination in a node's relay set. To represent a node's neighbor nodes which are closer to the destination than the node itself, we use the term *relay candidate set*.

The relay set of a node is selected from the node's relay candidate set based on the cost c in Formula (3). In our model, the cost is directly proportional to the node's transmission range. As each node has an adjustable transmission range, its transmission cost is also adjustable. In Figure 2, $c_{i,j}$ reflects the minimal power level to connect node i and j . If node i chooses cost c to be its transmission cost for a node j whose $c_{i,j} > c$, then node j cannot receive the packet. We define \vec{c}_i as the possible *cost level set* for node i , which includes all $c_{i,j}$, where j is in node i 's relay candidate set. For each $c \in \vec{c}_i$, we can find a subset of the relay candidate set. Each node j in this subset satisfies $c_{i,j} \leq c$. By arranging nodes in this subset according to their priorities, we can construct the relay set associated with the transmission cost c .

4.2 The Solutions

In previous opportunistic routing models [5, 6, 19], the optimal solution is related to a total order among intermediate nodes according to metrics such as expected transmission count or geometric distance. This is due to the fact that their models allow unbounded retransmissions in cases of transmission failure. Hence, it is always better to select a node that is 'closer' to the destination

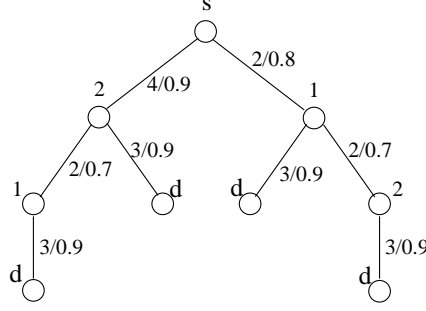


Figure 3. The optimal-solution tree for the example in Figure 1.

as forwarding nodes. However, in our utility-based opportunistic routing model, such a total order does not necessarily imply optimality because a node can increase its OpEU by selecting a node farther away from the destination. Compared with previous opportunistic routing models, a node can reduce its expected transmission count by choosing a neighbor farther away from the destination, but it is better for the node to choose itself as its forwarding node, which is equal to retransmission. Since we separate the effect of retransmission from opportunistic routing by forbidding retransmission in our utility-based opportunistic routing model, a total order is not related to the optimal solution.

We observe that two intermediate nodes can select each other as their forwarding nodes. For example, in Figure 1, both node 1 and node 2 can select each other as their forwarding nodes. However, if node 1 (2) has already received a packet, node 2 (1) cannot select node 1 (2) as its forwarding node. Otherwise, a loop occurs. Hence, the optimal solution should include all loop-free paths from the source to the destination. Each path represents a possible node sequence that a packet will traverse. In one path, node i may use node j as its forwarding node, and node j may use node i as its forwarding node in the other path. Therefore, for each node, the relay set is dynamic, depending on the node sequence that a packet has traversed. We give an optimal solution for the example in Figure 1 to illustrate the optimal solution as shown in Figure 3. In the optimal solution, source s can select both node 1 and node 2 as its forwarding nodes, and node 1 can include both node 2 and destination d in its relay set if node 1 receives a packet directly from s . Otherwise, node 1 only includes d in its relay set.

Despite its optimality, the complexity of the optimal solution is not desirable. Suppose that we have n nodes in the network; the maximum number of paths is the sum of permutations of combinations of different number of intermediate nodes in the path: $\sum_{k=1}^{n-2} (C_{n-2}^k \cdot P_k^k) = \sum_{k=1}^{n-2} \frac{(n-2)!}{(n-2-k)!} = \omega(2^n)$, where ω stands for lower bound. It is not an efficient solution for real applications. Therefore, we consider a heuristic solution that uses the expected utility of each node as a guideline to

determine a total order among all nodes and calculate their corresponding OpEUs.

In previous opportunistic routing models, intermediate nodes and the source node select the neighbor nodes that are ‘closer’ to the destination as their potential forwarding nodes. Therefore, an order (in terms of some distance metric) among all nodes should be determined so that the relays of a node can only be selected from those neighbor nodes ‘closer’ to the destination. The relative order between two forwarding nodes should reflect their relative contributions to increase utility. The best metric to reflect this is the OpEU. Since the OpEU is initially unknown, we initially prioritize nodes according to their expected utilities. The higher the expected utility of a node, the higher the priority of this node will be.

Our algorithm first sorts nodes according to their expected utility so that the relays of a node can only be selected from those neighbor nodes with higher expected utilities. Then, our algorithm computes the opportunistic routing scheme, where each node determines its transmission range, and prioritizes neighbor nodes within the determined transmission range. The benefit of this ordering is as follows: 1) the expected utility reflects the relative contributions of a node; 2) with this initial ordering among all nodes, the destination is reachable. Note that a random ordering cannot guarantee the reachability of the destination. For example, in Figure 1, if the ordering is $1 < d < s < 2$, s cannot find a routing scheme to the destination. Based on this ordering, we can calculate the OpEU of all nodes according to Formula (3).

In our algorithm, we use a Dijkstra-based algorithm to calculate the initial ordering of the expected utility of each node and calculate the OpEU at the same time. The calculation of the expected network utility starts from the destination with the initial expected network utility equal to the benefit of a packet. The expected utility and OpEU will be reduced at each intermediate node backward from the destination to the source according to the cost and stability of the links, where the node is an endpoint. Algorithm 1 (MaxUtility) iteratively finds the node that will reduce the expected network utility the least. When such a node has been found, Algorithm 2 (OpUtility) completes the transmission cost selection, relay set construction, and OpEU calculation. The priority of a node in the relay set is determined by its OpEU. The larger the OpEU, the higher the priority.

Our algorithm is presented in Algorithm 1 (MaxUtility) and Algorithm 2 (OpUtility), in which the following notations are used for presentation purposes.

- Q , the set of nodes whose expected utilities and OpEUs have been maximized.
- N_i , node i 's neighbor set.
- S_i , node i 's relay candidate set.

Algorithm 1 MaxUtility(\mathcal{N}, s, d, v)

Functionality: Compute the initial order of all nodes according to their expected utility, and the opportunistic routing scheme based on the initial ordering.

Output: The transmission cost and the corresponding prioritized relay set of each node.

- 1: Initialize, $Q \leftarrow \emptyset$;
- 2: **while** $s \notin Q$ **do**
- 3: Find node i with the largest u_i in \mathcal{N} , delete i from \mathcal{N} ;
- 4: Terminate if $u_i \leq 0$;
- 5: $Q \leftarrow Q \cup \{i\}$;
- 6: If $i \neq d$, then OpUtility(S_i);
- 7: For each node $j \in N_i$ and $j \in \mathcal{N}$, Relax(i, j);
- 8: **end while**

Relax(i, j)

- 1: Calculate the node j 's expected utility based on node i 's, $u_i \cdot p_{i,j} - c_{i,j}$;
 - 2: If $u_j < u_i \cdot p_{i,j} - c_{i,j}$, then update u_j ;
 - 3: Add node i into S_j ;
-

- c_i , the selected transmission cost of node i .
- P_i , node i 's relay set in a OpEU-based priority list.

The input to Algorithm 1 (MaxUtility) is the node set \mathcal{N} , source s , destination d , and source's benefit v . Each node $i \in \mathcal{N}$ has its neighbor set N_i . The link cost $c_{i,j}$ and link stability $p_{i,j}$ for each link (i, j) are also given. The input to Algorithm 2 (OpUtility) is node i 's relay candidate set S_i .

Initially, the expected utilities of all nodes except for d are set to 0. d 's expected utility and OpEU are set to v . In the beginning, d 's expected utility is the highest, thus, d is fetched from \mathcal{N} . d will relax its neighbors, send them its expected utility and OpEU, be added into their relay candidate set, and be removed from \mathcal{N} . The relaxation consists of three steps: first, node i calculates the expected utility of each neighbor in \mathcal{N} based on its own expected utility according to Formula (1); second, node i compares the neighbor's calculated expected utility with a neighbor's original expected utility and saves the larger value as the neighbor's new expected utility; third, node i adds itself into the neighbor j 's relay candidate set S_j .

Algorithm 1 (MaxUtility) repeatedly removes the node with the highest expected utility from \mathcal{N} and relaxes its neighbors until node s is removed from \mathcal{N} . When a node i is extracted from \mathcal{N} , Algorithm 2 (OpUtility) will be called. Algorithm 2 (OpUtility) adds each $c_{i,j}$ into \vec{c}_i if j is in the

Algorithm 2 OpUtility(S_i)

Functionality: Determine the transmission cost and corresponding relay set of node i .

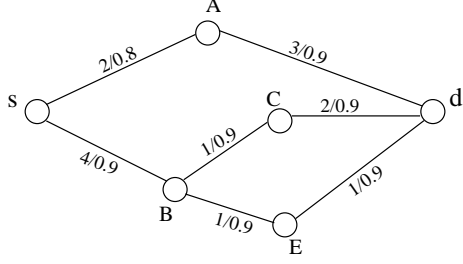
Output: i 's cost and corresponding prioritized relays of node i .

- 1: Initialize;
 - 2: Construct \vec{c}_i based on S_i ;
 - 3: **for** each $c \in \vec{c}_i$ **do**
 - 4: Initialize $P_i(c)$ to empty;
 - 5: Insert each node $j: j \in S_i$ and $c_{i,j} \leq c$, to relay set $P_i(c)$;
 - 6: Sort $P_i(c)$ to the decreasing order of $opu_j, j \in P_i(c)$;
 - 7: Compute $opu_i(c)$ according to Formula (3);
 - 8: **end for**
 - 9: $opu_i \leftarrow \max_{c \in \vec{c}_i} opu_i(c)$;
 - 10: Set c_i, P_i to corresponding $c, P_i(c)$;
 - 11: Return opu_i, c_i, P_i ;
-

S_i , and removes any duplicates. For each c in \vec{c}_i , it decides a corresponding relay set $P_i(c)$ which is a subset of S_i , and sorts the $P_i(c)$ based on the OpEU of each node in that set. It then computes node i 's OpEU $opu_i(c)$ based on the cost level c and relay set $P_i(c)$. Finally, it finds the largest $opu_i(c)$, selects a corresponding value of c as its transmission cost and sets a corresponding value of $P_i(c)$ as its relay set P_i .

If node i has a packet to send, it includes the relay set P_i in the header of the packet, and sends the packet at the selected cost level c_i . Each node that receives the packet looks for its ID in the relay set in the header. Each recipient delays an amount of time determined by its priority in the relay set and examines whether another node in the relay set is forwarding the packet. If it detects no such transmission before its delay timeout, it should forward the packet. The forwarding node replaces the relay set in the packet header with its own relay set and then forwards the packet. This process is repeated until the packet reaches the destination.

Algorithm 1 (MaxUtility) guarantees that nodes are sequentially extracted from \mathcal{N} based on their expected utilities. A node's relay candidates should be nodes with higher expected utilities (i.e. closer to destination). Expected utility reflects the distance, but we choose OpEU instead of expected utility to determine the priority of the nodes in the relay set in Algorithm 2 (OpUtility). There are two reasons that both these metrics are needed in our algorithms. First, prioritizing node i 's relay set by OpEU instead of expected utility leads to larger OpEU of a node i , which means a better routing scheme for node i . Second, using expected utility instead of OpEU to reflect the distance will avoid loops. Consider Figure 1. Node 2's OpEU exceeds node 1's because node 2



benefit	d	A	B	C	E	s
u	20	15	14.3	16	17	10
opu	20	15	15.74	16	17	11.4

Figure 4. An example illustrating the MaxUtility algorithm. The expected utility and OpEUs of each node under $v = 20$ are listed in the right table.

adds node 1 in its relay set, although node 2 still has lower value of expected utility. Looping occurs if we allow node 1 to add node 2 to its relay set.

4.3 Example

To illustrate Algorithm 4.2, we give an example, as shown in Figure 4, where the benefit v is set to 20. Initially, node d is selected. Its three neighbors A , C , and E are relaxed, with their expected utilities updated from 0 to 15, 16, and 17, respectively. At present, nodes A , C , and E have only one candidate relay node. Thus, their expected utilities are equal to their OpEUs. d itself is added into the relay candidate set of A , C , and E . Then, node E (with the second largest expected utility) is added into Q . Node E 's OpEU is 17 (equal to its expected utility), its transmission cost is 1, and its relay candidate set is d . Sequentially, nodes C , A , B and S are selected. When it is time for B to be selected, B has two relay candidates (C and E). B has only one cost level, which is 1. $opu_B = opu_E \times 0.9 + opu_C \times 0.9 \times (1 - 0.9) - 1 = 15.74$, which is larger than opu_A , which is 15. Finally, when s is selected, s has two relay candidates (A and B). s 's cost level set $\vec{c}_s = \{2, 4\}$. s sets its transmission cost to 4 and determines its relay candidate set $\{A, B\}$ because the expected network utility calculated based on the relay set associated with cost level 4 is larger than that of cost level 2. The OpEU and expected utility of each node are listed in the right-hand table of Figure 4.

4.4 Implementation

We consider two implementations: the centralized one, which is relatively costly for collecting global link state information, and the decentralized one, which can be gracefully integrated into reactive routing protocols such as AODV [1] or DSR [2].

Centralized implementation. We adopt link-state-based protocols in the centralized implementation. In traditional link-state-based protocols, information is spread through flooding techniques.

Initially, every node broadcasts its local network view (link cost and link stability associated with the node) to every other node. At the end of this, every node has a global view of the network (consistent, up-to-date routing information). Here we adopt a reactive version of the link state approach, assuming (i, j) exists if and only if (j, i) exists.

1. Source sends out a flooding message.
2. Each intermediate node responds to the first request by replying to the message.
3. A global directed flooding tree is formed rooted at the source. The first requester becomes the parent of the corresponding node.
4. Each node sends out its link state (the cost and stability of each link) to its parent node.
5. The source collects all link state information through the reversed spanning tree and then applies the algorithms.

Our centralized implementation spreads local link state information in a distributed manner, but computes expected utility and OpEU, selects the transmission cost, and decides the relay set of each node at the source in a centralized way. It requires every node to maintain local link state information. The source is in charge of computing. Although the centralized implementation can find the optimal transmission cost and the relay set in the priority list, it requires global link state information, which in turn requires broadcasting and information collection. It is too expensive and thus not a good implementation in practice.

Distributed implementation. The distributed implementation, unlike the above centralized implementation, computes expected utility and OpEU in a distributed manner. Expected utility and OpEU could be treated as the summary of local link state information. It is not required for each node to propagate all available local link state information to its upstream node. Instead, it propagates summarized routing information, expected utility, and OpEU to its upstream nodes. Each node locally determines its relay set and the priorities of nodes in the relay set according to the summarized routing information.

The distributed implementation can be gracefully integrated in a reactive routing protocol, such as AODV and DSR, where two phases are used. In the route discovery phase, the source broadcasts a RREQ (route request) to its neighbors. The RREQ is propagated in the network until it arrives at the destination, which initiates a RREP (route reply) containing relevant information following the reverse link leading to the source.

1. The source sends out a flooding message to inform d of its benefit.

2. Each intermediate node forwards the message upon receiving the first request.
3. The destination broadcasts its expected utility and OpEU to initialize a route discovery phase that will form a global directed flooding tree rooted at the destination.
4. Each node, including the source, sets a timer $w_j = f(v - u_j)$ on receiving the first expected utilities, where $f(v - u_j)$ is a function in which w_j is proportional to $v - u_j$. Before timeout, it improves its expected utility based on received expected utilities of its neighbors, adjusts its timer, and adds the nodes from which it receives expected utility into its relay candidate set.
5. After timeout, each intermediate node computes its OpEU, selects its transmission cost and decides the priorities of the nodes in its relay set, and sends out its expected utility and OpEU to all neighbors.

The initial value of the timer w_j reflects the expected utility of the node. The higher the u_j , the shorter time node j will backoff before it broadcasts its *RREP*. Whenever a node j receives an *RREP* that improves its expected utility, it will reduce the remaining backoff time accordingly. If there is no transmission delay, the node with maximum expected utility in \mathcal{N} will broadcast *RREP* first, which includes its expected utility and OpEU. Each node's relay candidate set will include all of the potential forwarding candidates, so the optimal transmission cost and relay set can be decided locally at each node. However, due to transmission delay, the node with larger expected utility is not necessarily the node that broadcasts *RREP* earlier. If the backoff time for a node is up, but the *RREP* that can increase its relay set and improve its expected utility and OpEU is still on the way, the expected utility and OpEU of the node is not maximized.

The distributed implementation is an approximation, using a timeout mechanism. It may lead to an incomplete relay candidate set, so the OpEU of a node is not always optimal. However, it has two desirable features: first, the calculation is distributed, each node decides its own transmission cost and relay set; second, it greatly reduces the transmission overhead, as only the expected utility and OpEU which summarize the link state information are propagated.

5 Simulation

In this section, we give an evaluation of our metric and algorithm. Without loss of generality, the link cost is modeled as the energy consumption. We compare different metrics for determining the priority of the nodes in the relay set under the framework of opportunistic routing. The metrics

include: (1) minimum hop count, (2) maximum path stability (path delivery ratio), (3) OpENU (expected utility), and (4) minimum cost.

5.1 Simulation Environment

Our simulation is based on NS-2.29. We set up the simulation in a $900m \times 900m$ area, which is the target field. We assume that nodes are homogeneous and can be deployed in this area arbitrarily. We fix the position of the source s and the destination d at locations $(50m, 450m)$ and $(850m, 450m)$ respectively and randomly deploy the intermediate nodes. The cost of any link (i, j) is generated according to the product of node i 's transmission power (per bit) and the packet size. The stability of any link (i, j) is the function of the received signal strength at node j (denoted as R_j). As in [20], the relation between bit-error-rate (p^{ber}) and R_j is a function of the modulation scheme. In general, this relation can be expressed in the following form:

$$p^{ber} \propto \text{erfc}\left(\sqrt{\frac{\text{constant} * R_j}{K}}\right),$$

where K is the ambient noise (noise spectral density) and $\text{erfc}(x)$ is defined as follows:

$$\text{erfc}(x) = 1 - \int_0^x e^{-t^2} dt.$$

As specific examples, the bit-error-rate is given by $p^{ber} = \text{erfc}\left(\sqrt{\frac{R_j}{K}}\right)$ for coherent on-off keying, by $p^{ber} = (M - 1) \times \text{erfc}\left(\sqrt{\frac{R_j \log_2(M)}{K}}\right)$ for M -ary frequency shift keying, and by $p^{ber} \approx 2 \sin\left(\frac{\pi}{M\sqrt{2}}\right) \times \text{erfc}\left(\sqrt{\frac{2R_j}{K}}\right)$. Since the bit-error-rates for different modulation schemes are constant multiples of each other, we adopt the coherent on-off keying in our simulation for simplicity. Since we are interested in the general dependence of the error rate on the received power as opposed to the details of a specific modulation scheme, we assume that the packet-error-rate p is equal to the multiplication of p^{ber} and the packet size. That is, the packet-error-rate is directionally-proportional to the bit-error-rate for a given packet size. As illustrated in Section 3, the received signal power is equal to the transmission power divided by $dist^\gamma$.

For each set of specified parameters, we run each algorithm 100 times and use the average value of the results to evaluate the performance. In the simulation, we consider n , the number of nodes (in our experiments we vary n between 30 and 100) as the tunable parameter.

5.2 Simulation Results

We first compare our heuristic solution with the optimal solution. Because the optimal solution is an exponential algorithm in terms of network size and is very slow in the NS-2 simulator, we

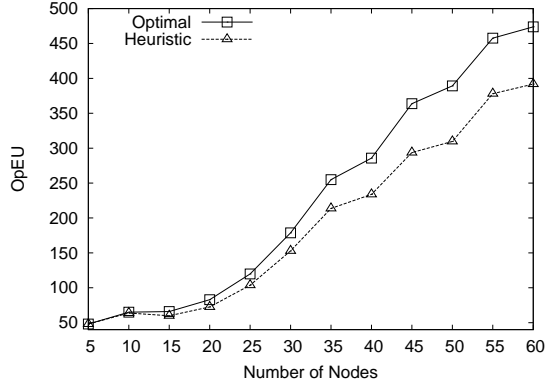
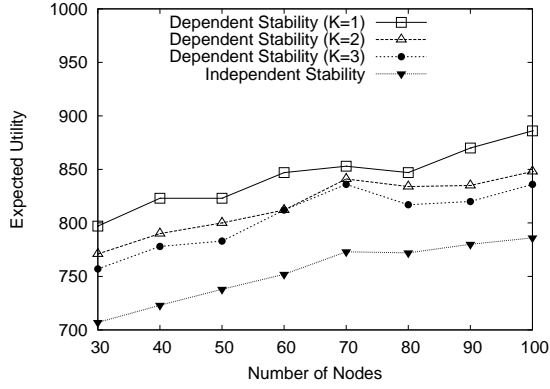


Figure 5. The comparison between the optimal solution and the heuristic solution.

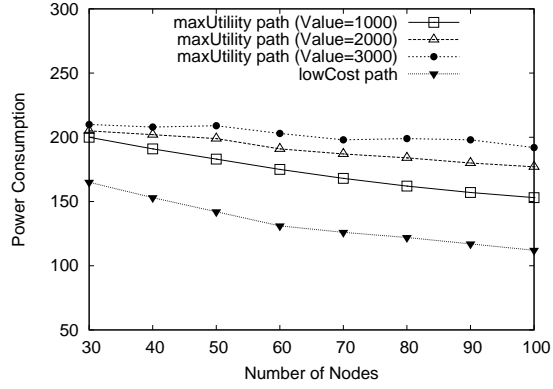
conduct this experiment in our customized simulator. The rest of the simulations are conducted in NS-2 (version 2.29). The result of the first simulation is presented in Figure 5. We set a stability range: $\alpha = 0$ and $\beta = 1$. The benefit value is set to 1000. We vary the number of nodes from 5 to 60 in increments of 5. The simulation results show that our heuristic solution is close to the optimal solution. This illustrates that expected utility provides good guidance for relay selection and prioritization. The reason for this is that expected utility characterizes those nodes that contribute the most to the increment of OpEU.

We then conduct sensitivity analysis, i.e., evaluate the effect of parameters such as ambient noise K and benefit value v . As discussed above, the stability of link (i, j) is generated based on the received power R_j . As a comparison, we also randomly generate the stability from $[0, 1]$. We use Dependent Stability to denote the former and Independent Stability to denote the latter. We consider three values of ambient noise: $K = 1, 2,$ and 3 . The simulation result is illustrated in Figure 6(a). As expected, the ambient noise decreases the stability, and thus reduces the expected utility. Moreover, we observe that the effects of the ambient noise on the expected utility have similar trends for different values of ambient noise. Therefore, we set $K = 1$ thereafter in the remainder of the simulations. We also observe that the maximum expected path, based on the stability generated according to the received power, has a larger expected utility than that based on randomly-generated stability. The reason for this is that the former has a smaller stability range compared with the latter.

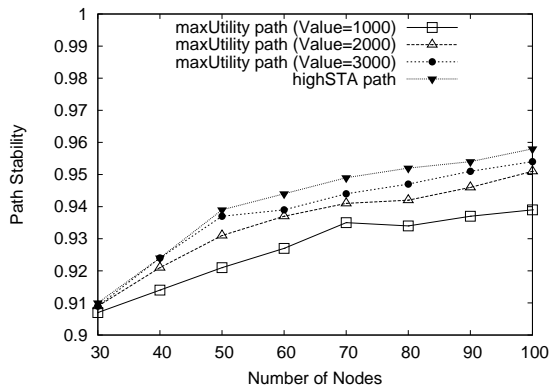
We also evaluate the value of benefit (v) on three metrics: cost, stability, and expected utility. We use maxUtility, lowCost, highSTA to denote the optimal routes under maximum expected utility, lowest cost, and highest stability metrics, respectively. Roughly speaking, a source with higher benefit is more likely to avoid taking risks by selecting a less stable (lower path delivery ratio)



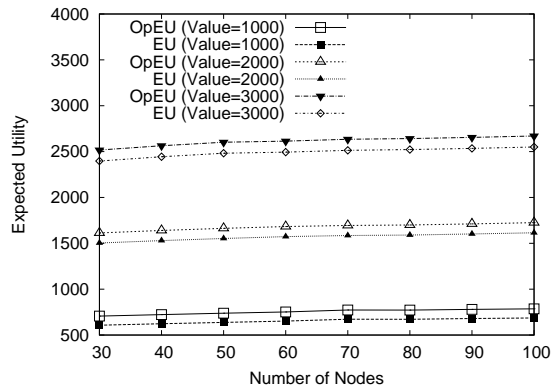
(a) Effect of ambient noise



(b) Effect of benefit on cost



(c) Effect of benefit on stability



(d) Effect of benefit on utility

Figure 6. The effect of ambient noise and benefit on utility-based opportunistic routing.

but low-cost path. If the value of v reflects the priority of a routing task, a higher priority routing task should select the more stable - but probably the most costly - routing path. Figures 6 (b) and (c) verify our analysis. Figure 6 (b) compares the cost of the selected routes under three different values of v (the bottom line is the lowest cost path), while Figure 6 (c) compares the path delivery ratio of the selected routes (the bottom line is the most reliable path).

To explain why the integration of the opportunistic routing scheme can further improve the utilization of network resource compared to the expected network utility based method, we have shown that the integration of the opportunistic routing scheme can increase the expected network utility. Since we have already shown that expected utility is an efficient metric to assess the utilization of network resources, it is valid to evaluate the utilization based on the value of expected utility. Figure 6 (d) shows the experiment results. In each of the selected values of v , the maximum opportunistic routing scheme (OpEU) beats the maximum expected utility route (EU) in terms of the expected utility.

To illustrate that the expected utility can be used to efficiently allocate energy cost over networks, we compare it with three other metrics (minimum hop count, lowest cost, and highest stability). For each metric, we compute the corresponding optimal path. The four optimal paths are compared using different metrics. In Figure 7, the optimal routes under maximum expected utility, minimum hop count, lowest cost, and highest stability are abbreviated as maxUtility, minHop, lowCost, highSTA, respectively. Figure 7 (a)-(d) compare the four optimal paths under expected utility metric, cost metric, path stability metric, and hop count metric, respectively.

Figure 7 (a) shows that the maxUtility path computed by our MaxUtility algorithm has the best performance in terms of expected network utility. From Figures 7 (b) and (c), we can see that the maxUtility path's performance is second best in terms of cost and path stability. The results show that our expected utility metric is a good metric to evaluate routing performance in wireless ad hoc networks. Our MaxUtility algorithm can achieve a good trade-off between cost and stability.

In Figure 7 (a)-(d), the minimal hop count path has similar performance to the most reliable path. The stability of a path is equal to the product of the stability of links on the path. Because the link stability is uniformly distributed, the lower the hop count, the higher the path stability. Figures 7 (c) and (d) verify the relation between the hop count and the path stability.

In Figure 7 (a), in terms of expected utility, the highSTA path has better performance than the lowest cost path. The reason for this is that the stability has a greater effect on expected utility than the cost. In Formula 1, if the benefit v is large enough, the expected utility will decrease by half with the stability $p_{s,d}$ decreasing by half, but the expected utility will not decrease too much with the cost $c_{s,d}$ doubled. Figure 7 (a) also shows that the expected utilities of the max expected utility path, the most reliable path, and the minimal hop count path increase with the increment of the number of nodes, but the expected utility of the lowest cost path decreases instead. With more nodes, more paths are available. With the increment of node number, although the lowest cost path algorithm has more choices, the selected path will have more hop counts and hence have lower stability. The effect of cost decrement cannot make up the effect of the stability decrement.

In Figure 7 (b), both the max expected utility path and the lowest cost path decrease with the increment of the number of nodes, but the most reliable path and the minimal hop count path do not. As we have argued in Figure 7 (a), with the increment of node number, the available paths increase, and hence lower-cost paths will be available. However, the cost of the most reliable path and the minimal hop count path do not necessarily decrease. Our maximum expected utility balances the trade-off between stability and cost well. It has the advantages of both stability metric and cost metric.

In Figure 7 (c), except for the lowest cost path, the path stability of the paths increases with

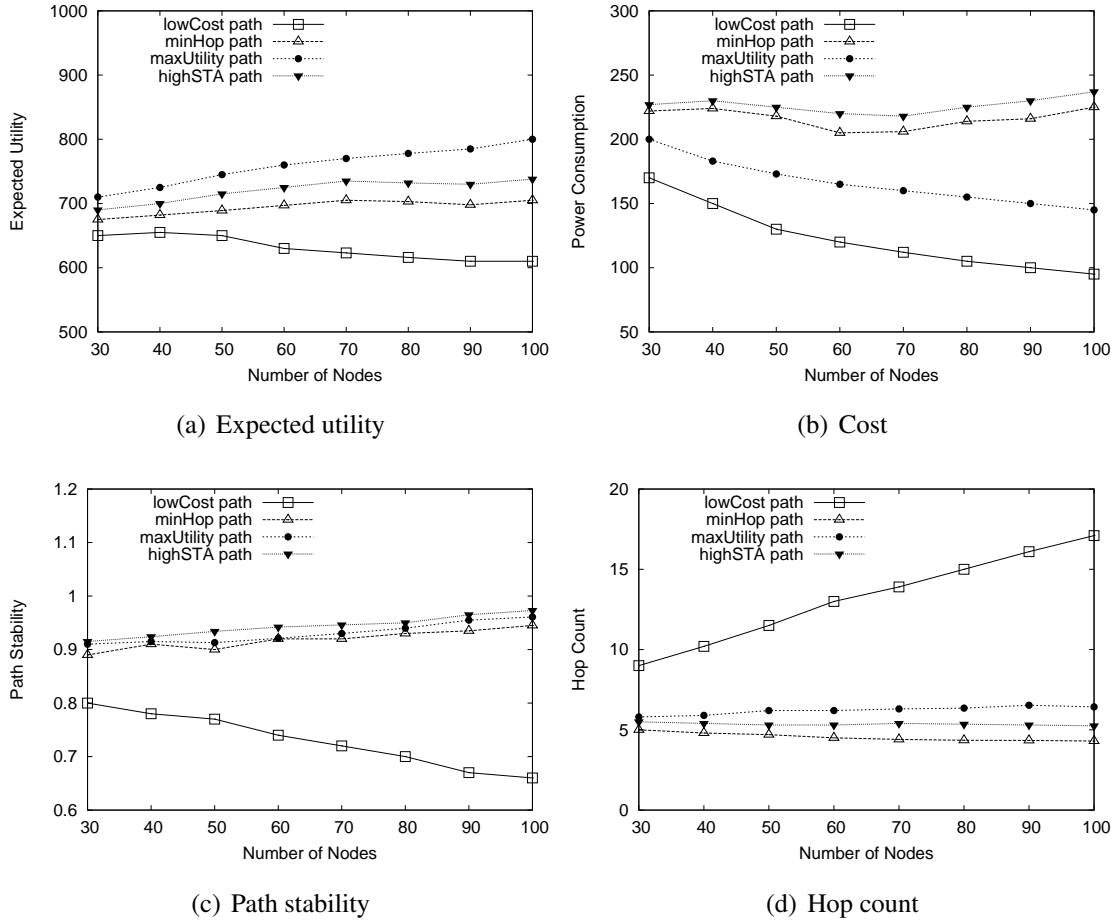


Figure 7. Performance comparison of four optimal routing paths under four different metrics.

the increment of the number of nodes. We have discussed the reason for this in the argument for Figure 7 (a). The desirable result is that the max expected utility path shows good path stability. The lowest-cost path is the worst of the four algorithms in terms of expected utility. The results are not surprising because the path stability is equal to the multiplication of the link stability and hence the link stability has a great influence on expected utility. The above experiment illustrates that expected utility is an efficient metric to assess the utilization of network resource.

6 Conclusion

In this paper, we studied the routing problem in ad hoc networks by considering resource scarcity and the unstable nature of mobile nodes. We used a model that is different from existing resource-efficient routing and adopted a new metric called *expected maximum network utility* to assess the routing optimality. By studying the relationship between link cost and link stability, we success-

fully combined these two different metrics and designed an algorithm which integrates the opportunistic routing scheme to find the efficient routing solution. In the future, we will explore the effect of signal strength on stability and analyze the relationship between retransmission and opportunism.

7 Acknowledgments

The work was supported in part by NSF grants ANI 0073736, CCR 0329741, CNS 0422762, CNS 0434533, EIA 0130806, CNS 0531410, and CNS 0626240.

References

- [1] C. Perkins. Ad Hoc On Demand Distance Vector (AODV) Routing IETF, Internet Draft, draft-ietf-manet-aodv-00.txt, November 1997.
- [2] D. B. Johnson and D. A. Maltz. Dynamic Source Routing in Ad hoc Wireless Networks. Mobile Computing, 1994.
- [3] J. Ai, A. A. Abouzeid, , and Z. Ye. Cross-layer optimal decision policies for spatial diversity forwarding in wireless ad hoc networks. In *Proceedings of the third IEEE Conference on Mobile Ad-hoc and Sensor Systems (MASS'06)*, 2006.
- [4] S. Biswas and R. Morris. Opportunistic routing in multi-hop wireless networks. *ACM SIGCOMM Comput. Commun. Rev.*, 34(1):69–74, 2004.
- [5] S. Biswas and R. Morris. ExOR: Opportunistic multi-hop routing for wireless networks. In *Proceedings of ACM SIGCOMM'05*, pages 133–144, 2005.
- [6] S. Chachulski, M. Jennings, Sachin K., and D. Katabi. Trading structure for randomness in wireless opportunistic routing. In *Proceedings of ACM SIGCOMM'07*, 2007.
- [7] R. R. Choudhury and N. H. Vaidya. MAC-layer anycasting in ad hoc networks. *SIGCOMM Comput. Commun. Rev.*, 34(1), 2004.
- [8] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. In *Proceedings of ACM MOBICOM'03*, 2003.
- [9] T. Cui, L. Chen, T. Ho, and S. Low. Opportunistic source coding for data gathering in wireless sensor networks. In *Proceedings of the 4th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS'07)*, 2007.

- [10] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli. Least-Cost Opportunistic Routing. In *Proceedings of 2007 Allerton Conference on Communication, Control, and Computing*.
- [11] H. Füßler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein. Contention-based forwarding for mobile ad hoc networks. *Elsevier's Ad Hoc Networks*, 1(4):351–369, 2003.
- [12] J. G. Jetcheva and D. B. Johnson. Adaptive demand-driven multicast routing in multi-hop wireless ad hoc networks. In *Proceedings of ACM MOBIHOC'01*, pages 33–44, 2001.
- [13] P. Larsson. Selection diversity forwarding in a multihop packet radio network with fading channel and capture. *SIGMOBILE Mob. Comput. Commun. Rev.*, 5(4), 2001.
- [14] B. Li, Y. Xue, and K. Nahrstedt. Price-based resource allocation in wireless ad hoc networks. Technical report, UIUCDCS-R-2003-2331, Univ. of Illinois at Urbana-Champaign, 2003.
- [15] M. Lu and J. Wu. Social welfare based routing in ad hoc networks. In *Proceedings of the 35th International Conference on Parallel Processing (ICPP'06)*, pages 211–218, 2006.
- [16] A. Mas-Collel, W. Whinston, and J. Green. *Microeconomic Theory*. Oxford University Press, 1995.
- [17] Y. Qiu and P. Marbach. Bandwidth allocation in wireless ad hoc networks: A price-based approach. In *Proceedings of IEEE INFOCOM'03*, 2003.
- [18] R. C. Shah, A. Bonivento, D. Petrovic, E. Lin, J. van Greunen, and J. Rabaey. Joint optimization of a protocol stack for sensor networks. In *Proceedings of IEEE Military Communications Conference, 2004 (MILCOM'04)*, pages 480–486, 2004.
- [19] R. C. Shah, S. Wietholter, A. Wolisz, and J. M. Rabaey. When does opportunistic routing make sense? In *Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'05)*, 2005.
- [20] B. Sklar. *Digital Communications, 2nd ed.* Prentice-Hall, Inc., 2001.
- [21] M. R. Souryal and N. Moayeri. Channel-adaptive relaying in mobile ad hoc networks with fading. In *Proceedings of IEEE SECON 2005*, 2005.
- [22] C. K. Toh. A novel distributed routing protocol to support ad-hoc mobile computing. In *the 1996 IEEE Fifteenth Annual International Phoenix Conference on Computers and Communication*, pages 480–486, 1996.

- [23] Y. Wang, M. Martonosi, and L. Peh. A new scheme on link quality prediction and its applications to metric-based routing. In *Proceedings of ACM SENSYS'05*, pages 288–289, New York, NY, USA, 2005.
- [24] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *Proceedings of ACM SENSYS'03*, pages 14–27, 2003.
- [25] Z. Ye, S. V. Krishnamurthy, and S.K. Tripathi. A framework for reliable routing in mobile ad hoc networks. In *Proceedings of IEEE INFOCOM'03*, pages 270 – 280, 2003.
- [26] B. Zhao and M. C. Valenti. Practical relay networks: A generalization of hybrid-ARQ. *IEEE Journal on Selected Areas in Communications*, 23(1), 2005.
- [27] S. Zhong, L. Li, Y. G. Liu, and Y. Yang. On designing incentive-compatible routing and forwarding protocols in wireless ad-hoc networks: an integrated approach using game theoretical and cryptographic techniques. In *Proceedings of ACM MOBICOM'05*, pages 117–131, 2005.
- [28] Z. Zhong and S. Nelakuditi. On the efficacy of opportunistic routing. In *Proceedings of the IEEE SECON 2007*, 2007.
- [29] Z. Zhong, J. Wang, S. Nelakuditi, and G. Lu. On selection of candidates for opportunistic anypath forwarding. *SIGMOBILE Mob. Comput. Commun. Rev.*, 10(4), 2006.
- [30] C. Zhou, D. Qian, and H. Lee. Utility-based routing in wireless ad hoc networks. In *Proceedings of the 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS'04)*, pages 588–593, 2004.
- [31] M. Zorzi and R. Rao. Geographic random forwarding (GeRaF) for ad hoc and sensor networks: energy and latency performance. *IEEE Transaction on Mobile Computing*, 2(4), 2003.

MINGMING LU — He is a PhD candidate in the Department of Computer Science and Engineering, Florida Atlantic University. He received the BS and MS degrees in computer science in 2000 and 2003, respectively, from Tongji University, Fudan University, Shanghai, China. His current research focuses on the algorithm design in the wireless ad hoc and sensor networks.

Contact information: Address: Department of Computer Science and Engineering Florida Atlantic University 777 Glades Road Boca Raton, FL 33431-0991, Email: mlu2@fau.edu, Phone: 1-561-297-2804, Fax: 1-561-297-2800

Feng Li — He received the B.S. degree (2002) and M.S. degree (2005) in computer science from Southeast University, China. He is currently working toward the PhD degree in the Department of Computer Science and Engineering, Florida Atlantic University, under the supervision of Dr. Jie Wu. His research interests include security and trust issues in Wireless Networks and Mobile Computing, Parallel and Distributed Systems.

Contact information: Address: Department of Computer Science and Engineering Florida Atlantic University 777 Glades Road Boca Raton, FL 33431-0991, Email: fli4@fau.edu, Phone: 1-561-297-2804, Fax: 1-561-297-2800

Dr. Jie Wu — He is a Professor at Department of Computer Science and Engineering, Florida Atlantic University. He has published over 200 papers in various journal and conference proceedings. His research interests are in the area of mobile computing, routing protocols, fault-tolerant computing, and interconnection networks. Dr. Wu is a Member of ACM and a Senior Member of IEEE.

Contact information: Address: Department of Computer Science and Engineering Florida Atlantic University 777 Glades Road Boca Raton, FL 33431-0991, Email: jie@cse.fau.edu, Phone: 1-561-297-3491, Fax: 1-561-297-2800