

# Social Welfare Based Routing in Ad hoc Networks

Mingming Lu and Jie Wu \*

Department of Computer Science and Engineering  
Florida Atlantic University  
Boca Raton, FL 33431

## Abstract

*Due to resource scarcity, a paramount concern in ad hoc networks is to utilize the limited resources efficiently. The self-organized nature of ad hoc networks makes the social welfare based approach an efficient way to allocate the limited resources. However, the effect of instability of wireless links has not been adequately addressed in the literature. To efficiently address the routing problem in ad hoc networks, we introduce a new metric, maximum expected social welfare, and integrate the cost and stability of nodes in a unified model to evaluate the optimality of routes. The expected social welfare is defined in terms of expected benefit (of the routing source) minus the expected costs incurred by forwarding nodes. Based on our new metric, we design an optimal and efficient algorithm, and implement the algorithm in both centralized (optimal) and distributed (near-optimal) manners. We also extend our work to incorporate retransmission and study the effect of local and global retransmission restrictions on the selection of routes.*

**Keywords:** Ad hoc networks, distributed implementation, retransmission, routing, social welfare, stability.

## 1 Introduction

Ad hoc networks suffer from the power shortage of the network devices, thus a major concern in ad hoc networks is to save energy. Existing energy-efficient routing protocols save energy by selecting the lowest energy cost route. However, energy saving is not equal to energy efficiency. Consider a large number of packets to be delivered from a source to a destination. If the lowest cost path is very unstable, and hence most packets transmitted through this path are lost, it is energy waste rather than energy saving compared to a stable but more costly path. Therefore,

energy-efficient routing protocols should take both energy consumption and link stability into account.

A major task of sensor networks is to monitor environmental change and report unexpected events to an information sink. With the improvement of sensor techniques, a single sensor can monitor different events simultaneously. These events may have different priorities to the sink because an event like a fire alarm is more emergent than the increment of temperature. It is reasonable to send the higher priority event through a more stable route even at the expense of more energy cost.

From the above observations, we realize that there is a tradeoff between stability and energy cost in the selection of an optimal route, and this tradeoff depends on the priority of the data to be sent. To model the priority and the tradeoff, we borrow the concept of *benefit(value)*, *cost*, and *social welfare* from economic theory.

The benefit is used to model a successful data delivery from the event observer (information source) to the information destination. If data is delivered from the information source to the information sink successfully, the system will achieve some benefit, whose value depends on the priority of the data. The cost is used to model the energy cost of links and routes. The social welfare is equal to the benefit minus the cost. From the economic point of view, the higher the social welfare, the better performance the system has.

Given the benefit of a successful data delivery, the tradeoff lies in the selection of the optimal route. To increase the chance to obtain the benefit, it is better to select the more stable route because the more stable a route, the higher the *expected benefit* ( $= \text{benefit} \times \text{delivery ratio}$ ). But the increment of the expected benefit may be at the expense of the increment of the energy cost because it is usually the case that the more stable a route, the higher the cost of the route.

We adopt the *expected social welfare* ( $= \text{expected benefit} - \text{expected cost}$ ), which integrates link cost, link stability, and the benefit, as the metric to evaluate the routing optimality. Based on this metric, we design an efficient routing algorithm, which can identify the optimal route from any source to a given destination for a given value of bene-

---

\*This work was supported in part by NSF grants ANI 0073736, EIA 0130806, CCR 0329741, CNS 0422762, CNS 0434533, and CNS 0531410. Contact address: jie@cse.fau.edu

fit. Although our algorithm can find optimal routes from all sources to a single destination, we consider a single source-destination pair in our model in order to avoid intractable analysis. We prove the optimality of our algorithm, and implement it in both centralized and distributed manners.

Our distributed implementation only propagates the summarized routing information (the expected social welfare) from the destination to the source, instead of all the routing information. Our scheme is easy to implement based on existing reactive routing protocols, such as AODV [1] or DSR [2], without introducing additional cost.

We also extend our model by incorporating a retransmission mechanism. Retransmission can increase the link stability, which in turn increases the stability of routes, but this increase is at the expense of cost. In the extreme case of unlimited retransmission, the packet delivery ratio is 100% and the cost can be larger than the benefit, which causes negative social welfare.

Intuitively, there exists a trade-off between stability and cost in retransmission. But in terms of expectation, it can be proved that the larger the upper bound of the number of transmissions, the larger the expected social welfare. However, in reality, the number of retransmissions is constrained by local and global restrictions. The intermediate nodes have motivation to set a local quota (a local retransmission restriction) because too many retransmissions is not beneficial to them. The source has reason to set a global quota (a global retransmission upper bound) over the routing path, if it intends to control the number of transmissions and the total transmission cost. Protocols like TCP have a constant hop-by-hop (local) and end-to-end (global) retransmission restriction. Because of the above arguments, we extend our model to allow retransmission and study the effect of the local quota and global quota on the route selection.

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 [14, 18] address the issue on how to collect this information. 2) We allow hop-to-hop retransmission and assume that each forwarding attempt consumes the same cost.

## 2 Related work

The concept of the social welfare [10] is borrowed from economics. There exist works that use social welfare as the optimization objective. Li, Xue, and Nahrstedt [8] present a price-based scheme to effectively allocate resources among multiple multi-hop flows. Their approach maximizes the aggregated utility of flows (the social welfare), while maintaining basic fairness among multiple flows. Qiu and Marbach [11] propose a market-based approach to efficiently allocate bandwidth in wireless ad hoc networks. Although there are numerous existing works applying social welfare

related approaches, as far as we know, none of them combines stability and link cost and designs an optimization model to maximize the expected social welfare.

In a market-based model, another concern is to determine the price of service, which is called payment in literature. Existing works use a first-price scheme [7] or a second-price scheme [3, 18], which can be classified as an auction-based scheme, to determine the payment. There exist many other payment methods like Nuglets [5] and Sprite [17]. We do not address the implementation of payment because the expected social welfare does not depend on the payment scheme.

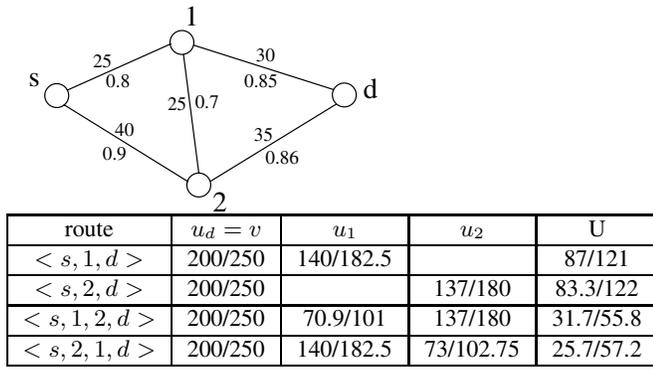
Various existing routing protocols [4, 18] pursue the minimum hop count or minimum cost. As has been discussed elsewhere [6, 15], these metrics are not necessary ideal because they did not take link stability into account.

Another routing algorithm [13] 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 [16] adopts “best worst link”, which identifies the least stable link (the bottleneck link) over any route and selects the route with the best “bottleneck”. Although these schemes consider the link stability issue, they fail to differentiate between two routes with the same bottleneck link but different link characteristic.

ETX [6] and MintRoute [15] adopt path delivery ratio (PDR) by measuring hop-by-hop link delivery ratio (LDR) along the path. ETX weights links with a metric called minimum expected transmission number, which is equal to the link cost divided by LDR. However, this metric requires an unrealistic assumption that retransmission is unlimited.

Wang, Martonosi, and Peh [14] propose a simple method, called link quality indicator (LQI), to evaluate the link stability. Through LQI, the link stability can be measured over the reception of a single packet in a realistic environment, which makes the collection of link stability information practical.

In ad hoc networks, retransmission schemes are applied to increase the reliability in routing. Lou and Wu [9] discussed the trade-off between broadcast redundancy (including retransmission) and delivery ratio. Scott and Yasinsac [12] proposed a routing protocol that dynamically adjusts retransmission probability according to the local topology. In this paper, we adopt a hop-by-hop retransmission scheme and integrate the retransmission scheme into our maximum social welfare based model, and design an optimal algorithm for the introduction of the local quota and a heuristic for the introduction of the global quota.



**Figure 1. The effects of the link stability and the benefit on the selection of the optimal route. The table shows the RESWs of nodes on each route. In each cell, there are two values separated by ‘/’. The left value is the RESW under benefit of 200, while the right one is the one corresponding to benefit 250.**

### 3 The model

#### 3.1 Basic definitions

We consider a source-destination pair  $(s,d)$ , in which the destination  $d$  expects to receive data from the source  $s$ . The per packet benefit is denoted as  $v$ . That is, the system will obtain  $v$  for each successfully delivered packet. The network is modeled as a unit disk graph  $(\mathcal{N}, E)$ , where  $\mathcal{N} = \{1, 2, \dots, N\}$  is the set of nodes and  $E$  is the set of links. For each link  $(i, j) \in E$ , there are two properties: link cost and link stability. Link cost  $c_{i,j}$  is the minimal energy level to connect  $i$  and  $j$ , while link stability  $p_{i,j}$  (also called link delivery ratio (LDR) [15]) is the ratio of received packets by node  $j$  to transmitted packets by  $i$ .

To illustrate the basic idea of our new metric, we first consider a single-link route from  $s$  to  $d$  with link stability  $p_{s,d}$  and link cost  $c_{s,d}$ . Since  $d$  receives a packet with probability  $p_{s,d}$ , the system has  $p_{s,d}$  chance to obtain the benefit  $v$  at the cost of  $c_{s,d}$ . Note that the system obtains  $v$  if and only if the packet is delivered to  $d$ . From the economic point of view, the expected social welfare of this route is the expected benefit minus the route’s cost, i.e.,

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

However, Formula (1) should be carefully investigated before it can become a useful metric. In fact, Formula (1) cannot be directly applied to a multi-hop route. Consider the two-hop route  $\langle s, 1, d \rangle$  shown in Fig. 1, where the cost and the stability of a link are displayed above (left) and below (right) the link, respectively. We cannot simply set

$p_{s,d} = p_{s,1} \cdot p_{1,d} = 0.8 \times 0.85$ ,  $c_{s,d} = c_{s,1} + c_{1,d} = 25 + 30$ , and apply Formula (1). Actually,  $c_{s,d} = c_{s,1} + c_{1,d} \cdot p_{s,1} = 25 + 30 \times 0.8$  because the cost  $c_{1,d}$  is consumed if and only if a packet is delivered to 1.

Even if the link stability is taken into account in evaluating the link cost, it is still not easy to extend to the scenario of multi-hop routes. For example, consider the multi-hop route  $\langle s = 1, 2, \dots, k-1, d = k \rangle$ , the expected social welfare should be calculated as:

$$U = v \cdot \prod_{j=1}^{k-1} p_{j,j+1} - \sum_{i=1}^{k-1} c_{i,i+1} \prod_{j=1}^{i-1} p_{j,j+1} \quad (2)$$

To make Formula (2) scalable and easy to calculate in a distributed way, we design a better way in the next subsection to calculate a route’s expected social welfare.

#### 3.2 Metric for a multi-hop route

An important observation is that the implementation of the benefit  $v$  depends on the successful delivery of a packet to the destination node. Thus, from the destination’s point of view, we can view any intermediate node as the virtual source and calculate the corresponding expected social welfare from the virtual source to the destination. For example, in Fig. 1, we can view intermediate node 1 as the virtual source and the related expected social welfare is:  $u_1 = v \cdot p_{1,d} - c_{1,d}$  by Formula (1). Because  $u_1$  is not the real expected social welfare, we call  $u_1$  the *residual expected social welfare* (RESW) of node 1.

On the other hand, from the source’s point of view, we can view any intermediate node as the virtual destination and equate the benefit to the intermediate node’s RESW as if the system would obtain that amount of benefit if a packet is delivered to the intermediate node. For example, in Fig. 1, source  $s$  will get benefit  $u_1$  if the packet is delivered to 1. Thus, the expected social welfare can be calculated as:  $U = u_1 \cdot p_{s,1} - c_{s,1}$ , where  $u_1$  can be calculated from  $u_1 = v \cdot p_{1,d} - c_{1,d}$ .

The above method can be extended to the calculation of multi-hop routes. We can recursively apply Formula (1) starting from destination  $d$  to obtain the expected social welfare. For example, consider the 3-hop route  $\langle s, 1, 2, d \rangle$  in Fig. 1. For link  $(2, d)$ , we have  $u_2 = u_d \cdot p_{2,d} - c_{2,d}$ . By recursively applying Formula (1), we have  $u_1 = u_2 \cdot p_{1,2} - c_{1,2}$  and then  $U = u_1 \cdot p_{s,1} - c_{s,1}$ , which is the expected social welfare of the 3-hop route.

The correctness of the above recursive method can be verified by comparing the results with Formula (2). An important 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 depends on the value of the benefit. Consider the example in Fig. 1, there are four routes:  $\langle s, 1, d \rangle$ ,  $\langle s, 2, d \rangle$ ,  $\langle s, 1, 2, d \rangle$ , and

$\langle s, 2, 1, d \rangle$ . If the benefit  $v = 200$ , the optimal route is  $\langle s, 1, d \rangle$ , but if  $v = 250$ , the optimal route is  $\langle s, 2, d \rangle$ .

## 4 The solution

### 4.1 The algorithm and its complexity

In our algorithm, the calculation of the expected social welfare starts from the destination with the initial expected social welfare equal to the per packet benefit. The RESW 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 social welfare the least. A few additional notations are used in MaxUtility:

- $Q$ , the set of nodes whose RESWs have been maximized.
- $u_i$ , which maintains node  $i$ 's current RESW.
- $N_i$ , the set of  $i$ 's neighbors.

The input to MaxUtility is the node set  $\mathcal{N}$ , source  $s$ , destination  $d$ , and the per packet 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.

---

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

---

- 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: For each node  $j \in N_i$  and  $j \in \mathcal{N}$ , Relax( $i, j$ );
- 7: **end while**

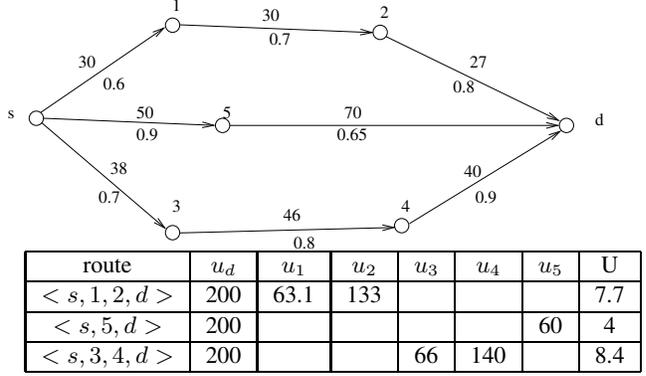
#### Relax( $i, j$ )

- 1:  $u_j \leftarrow u_i \cdot p_{i,j} - c_{i,j}$  if  $u_j < u_i \cdot p_{i,j} - c_{i,j}$ ;
- 

Initially, the RESWs of all nodes except  $d$  are set to  $-\infty$ .  $d$ 's RESW is set to  $v$ . In the beginning,  $d$ 's RESW is the highest, thus,  $d$  is fetched.  $d$  will relax the RESWs of its neighbors and then be put into  $Q$ . The relaxation consists of two steps: first, node  $i$  calculates the RESW of each neighbor according to Formula (1); second, node  $i$  compares a neighbor's calculated RESW with its original RESW and saves the larger value as the neighbor's new RESW.

MaxUtility repeatedly removes the node with the highest RESW from  $\mathcal{N}$ , inserts it into  $Q$ , and relaxes its neighbors until node  $s$  is inserted into  $Q$  or no path with positive expected social welfare can be found. In the later case, the

algorithm terminates because negative or zero expected social welfare means that on average the system wastes its resource.



**Figure 2. An example to illustrate the MaxUtility algorithm(The RESWs of nodes of each route are enumerated in the table).**

To illustrate our idea, we give an example in Fig. 2, in which the integer number above a link and the float number below the link are the cost and the stability of the link, respectively. Besides  $s$  and  $d$ , there are five other nodes (nodes 1, 2, 3, 4, and 5). If the per packet benefit is 200, the optimal route is  $\langle s, 3, 4, d \rangle$ . Note that in Fig. 2, the lowest cost path is  $\langle s, 1, 2, d \rangle$ .

If  $\mathcal{N}$  is implemented with a binary heap, the total execution time of step 3 is  $O(n \log n)$ , where  $n$  is the number of nodes. Each Relax( $i, j$ ) takes also  $O(\log n)$  time, since fetching or storing  $u_i$  costs time  $O(\log n)$ . Relax( $i, j$ ) executes at most time of  $e$ , where  $e$  is the number of links. So Relax( $i, j, C$ ) has a total time of  $O(e \log n)$ . Therefore, MaxUtility can be implemented in  $O((e + n) \log n)$ . Due to space limitation, the proof of optimality of our algorithm is omitted.

### 4.2 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] and DSR [2].

#### 4.2.1 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. Destination sends out a flooding message.
2. Each intermediate node responds to the first request by replying to the message.
3. The global directed flooding tree is formed rooted at the destination. 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 destination 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 RESWs at the destination in a centralized way. It requires each node to maintain local link state information.

Although the above centralized implementation can find the optimal route, 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. In the following, we present a distributed implementation, which requires much less message transmission.

#### 4.2.2 Distributed implementation

The distributed implementation, unlike the above centralized implementation, computes RESW in a distributed manner. RESW could be treated as the summary of local link state information. Each node need not propagate all available local link state information to its upstream node. Instead, it propagates a summarized routing information (RESW) to its upstream nodes.

The distributed implementation can be gracefully integrated in a reactive routing protocol, such as AODV [1] and DSR [2], 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 gets to the destination, which then initiates a RREP (route reply) containing relevant information following the reverse link leading to the source.

1. The destination broadcasts its RESW to initialize a route discovery phase that will form a global directed flooding tree rooted at the destination.
2. Each node, including the source, sets a timer  $w_j = v - u_j$  on receiving the first RESWs. Before timeout it

$(s, 1)$	$(s, 5)$	$(s, 3)$	$(1, 2)$	$(5, d)$	$(3, 4)$	$(2, d)$	$(4, d)$
32.4	54	42.6	33.6	77	51.5	30.24	43.2
.84	.99	.91	.91	.88	.96	.96	.99

**Table 1. The new link cost and link stability of Fig. 2 with  $\theta_l = 2$  (The second and third rows are link cost and link stability, respectively. )**

improves its RESW based on the received RESWs of its neighbors and adjusts its timer.

3. After timeout, each intermediate node computes and sends out its RESW to all neighbors.

If there is no transmission delay, the node with maximum RESW will always broadcast *RREP* first, which includes its RESW. This will enable the distributed implementation to find the optimal route. However, due to transmission delay, the node with larger RESW 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 RESW is still on the way, the RESW of the node is not maximized.

The distributed implementation is an approximation, using a timeout mechanism. Thus, the RESW of a node is not always optimal. But it has two desirable features: First, the calculation is distributed, each node decides its own transmission cost and relay set; Second, it greatly reduced the transmission overhead, as only the RESW, which summarizes the link state information, will be propagated.

## 5 The extension

In this section, we extend our model to incorporate the hop-by-hop retransmission. The retransmission can increase a link's stability, but it also introduces additional cost for the link. For each link, whether the retransmission is beneficial (in terms of increasing the expected social welfare) or not is an interesting problem. Moreover, if a retransmission attempt of a packet fails over a link, should we retransmit the packet over the link again? How many retransmission is appropriate?

To answer these questions, we adopt two transmission upper bounds, local quota and global quota. The local quota of an intermediate node is the maximum number of allowed transmissions for the intermediate node, while the global quota of a route is the number of total allowed transmissions over the route. For simplicity, we consider homogenous local quotas for all intermediate nodes, homogenous global quotas for all available routes, and assume each transmission cost is the same.

We first consider the local quota. The local quota can be finite or infinite. The original problem (the case without

retransmission) is a special case (the local quota = 1). Let  $\theta_l$  denote the local quota. The new stability of link  $(i, j)$  is  $1 - (1 - p_{i,j})^{\theta_l}$ , because the packet will lose if and only if all  $\theta_l$  transmission attempts fail. The probability for  $i$  transmitting the packet exactly  $l$  times is  $(1 - p_{i,j})^{l-1} p_{i,j}$ . The expected number of transmission from node  $i$  to node  $j$  is  $\sum_{l=1}^{\theta_l} l(1 - p_{i,j})^{l-1} p_{i,j}$ . Thus, the expected cost of link  $(i, j)$  is  $c_{i,j} \sum_{l=1}^{\theta_l} l(1 - p_{i,j})^{l-1} p_{i,j}$ . By replacing the link stability and link cost in Formula (1) with the new link stability and link cost, we can directly apply MaxUtility algorithm to solve the problem. Note that if the local quota  $\theta_l$  is unlimited, i.e.,  $\theta_l \rightarrow \infty$ , the problem is reduced to the lowest cost path problem [6, 18].

We use the same example in Fig. 2 to illustrate the change of link stability, link cost, and the optimal route. The new link stability and link cost in case of  $\theta_l = 2$  are given in Table 1. The new optimal route is  $\langle s, 1, 2, d \rangle$ .

The transmission upper bound can also be controlled by the global quota. The recurrence described by Formula (1) does not hold with the existence of the global quota, because an intermediate node's expected cost will depend on the number of transmissions of previous nodes on the route.

We design a heuristic algorithm that addresses the case with the existence of both local quota and global quota. Our heuristic uses  $\theta_l$  rounds to select the objective route. In each round, the heuristic sets the local quota to  $i$  and uses a modified MaxUtility algorithm to find the optimal route with the local quota  $i$ . The route with the maximum expected social welfare and the sum of local quotas (path length multiplied by the local quota) less than or equal to the global quota will be selected as the routing path. Our heuristic algorithm is shown in Algorithm 2.

---

**Algorithm 2** ExtMaxUtility

---

```

1:  $U(R^*) \leftarrow -\infty; R^* \leftarrow \emptyset;$ 
2: for  $i = 1$  to  $\theta_l$  do
3:    $R_i \leftarrow \text{MaxUtility}(G, s, d, v, i);$ 
4:    $R^* \leftarrow R_i$  if  $|R_i| \cdot i \leq \theta_g$  and  $U(R_i) > U(R^*);$ 
5: end for
6: return  $R^*;$ 

```

---

We use the same example shown in Fig. 2 to illustrate that the restriction of global quota does affect the selection of the routing path. In Fig. 2, if the global quota  $\theta_g = 4$  and the local quota  $\theta_l = 2$ , the optimal route is  $\langle s, 5, d \rangle$  instead of  $\langle s, 1, 2, d \rangle$  because the sum of the local quota on path  $\langle s, 5, d \rangle$  is equal to the global quota, while the sum of the local quota on path  $\langle s, 1, 2, d \rangle$  is larger than the global quota.

## 6 Simulation

In this section, we evaluate our metric and our distributed implementation in NS2. 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 our distributed implementation. The metrics include: (1) minimum hop count, (2) highest path stability, (3) maximum expected social welfare, and (4) minimum cost.

### 6.1 Simulation environment

All approaches are simulated on NS-2.29. We set up the simulation in a  $900m \times 900m$  area, which is the target field. We assume 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. In our experiments, the energy cost between any two nodes is proportional to their distance. More specifically, for any two nodes  $i$  and  $j$  with distance  $dist_{ij}$ , the energy cost of  $i$  sending a message to  $j$  is defined as the function  $c_{i,j} = dist_{ij}^\gamma + cons$ , where  $\gamma = 2$  and  $cons$  is the energy constant. The stability of each link is randomly generated (uniform distribution) in the range  $[\alpha, \beta]$ , where  $0 \leq \alpha \leq \beta \leq 1$ .

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.

### 6.2 Simulation results

To illustrate that expected social welfare (ESW) can be used to efficiently allocate energy cost over networks, we compare it with other three 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 Fig. 3, the optimal routes under maximum ESW, minimum hop count, lowest cost, and highest stability are abbreviated as maxESW, minHop, lowCost, highSTA, respectively. Fig. 3 (a), (b), (c), and (d) compare the four optimal paths under ESW metric, cost metric, path stability metric, and hop count metric, respectively.

Fig. 3 (a) shows that the maxESW path computed by our MaxUtility algorithm has the best performance in terms of expected social welfare. From Fig. 3 (b) and (c), we can see that the maxESW path's performance is second to best in terms of cost, and path stability, respectively. The results show that our ESW metric is a good metric to evaluate

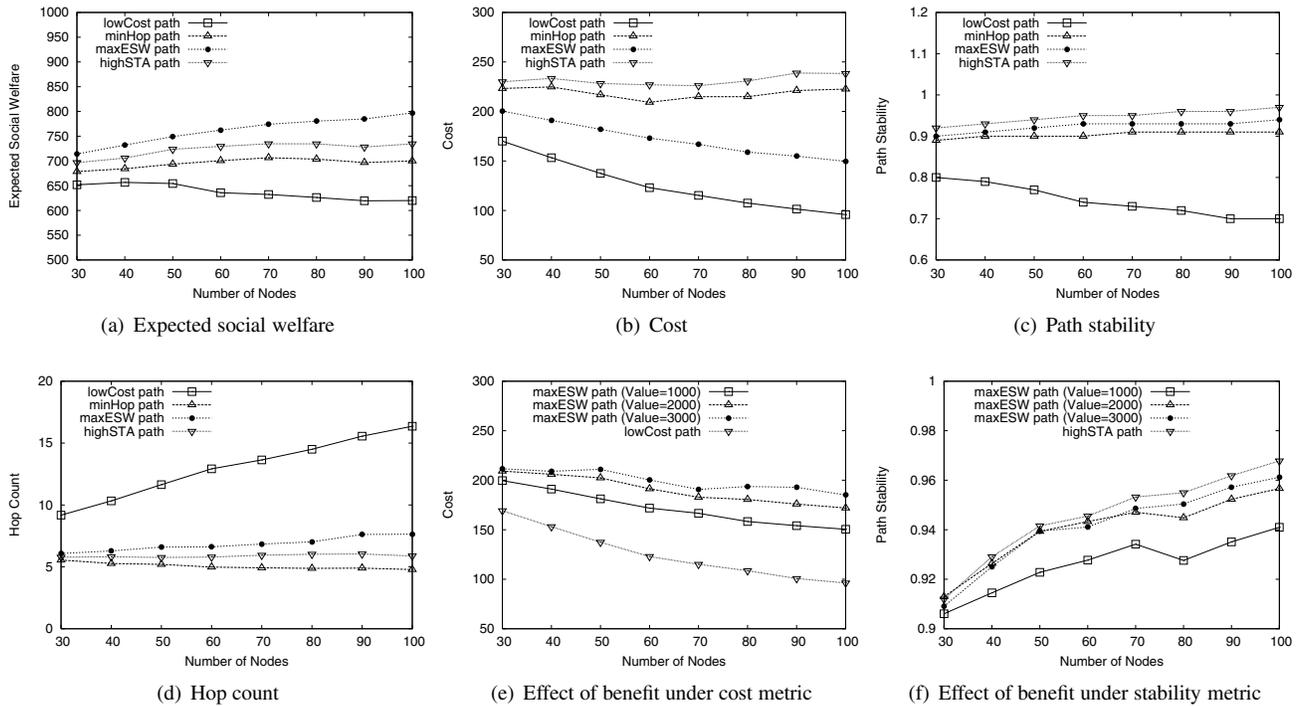


Figure 3. Simulation results.

routing performance in wireless ad hoc network. Our Max-ESW algorithm can achieve a good trade-off between cost and stability.

In Fig. 3 (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. Fig. 3 (c) and (d) verify the relation between the hop count and the path stability.

In Fig. 3 (a), in terms of ESW, the highSTA path has better performance than the lowest cost path. The reason is that the stability has more effect on ESW than the cost. In Formula 1, if the benefit  $v$  is large enough, the ESW will decrease by half with the stability  $p_{s,d}$  decreasing by half, but the ESW will not decrease too much with the cost  $c_{s,d}$  doubled. Fig. 3 (a) also shows that the ESWs of the max ESW path, the most reliable path, and the minimal hop count path increase with the increment of the number of nodes, but the ESW 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 Fig. 3 (b), both the maxESW 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 Fig. 3 (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 ESW balances the trade-off between stability and cost well. It has the advantages of both stability metric and cost metric.

In Fig. 3 (c), except for the lowest cost path, the path stability of the paths increases with the increment of the number of nodes. We have discuss the reason in the argument for Fig. 3 (a). The desirable result is that the maxESW path shows a good path stability. The lowest cost path is the worst of the four algorithms in terms of ESW. 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 ESW. The above experiment illustrates that ESW is an efficient metric to assess the utilization of network resource.

We also evaluate the effect of the value of benefit on the computation of the optimal route. Roughly speaking, a source with higher benefit is more likely to avoid taking risks by selecting a less stable but low cost path. If the value of  $v$  reflects the priority of a routing task, a higher priority routing task should select more stable but probably

the most costly routing path. Fig. 3 (e) and (f) verify our analysis. Fig. 3 (e) compares the cost of the selected routes under three different values of  $v$  (the bottom line is the lowest cost path), while Fig. ?? (f) compares the path delivery ratio of the selected routes (the bottom line is the most reliable path).

## 7 Conclusion

In this paper, we study the routing problem in ad hoc networks. Considering resource scarcity and the unstable nature of mobile nodes in ad hoc networks, we use a model different from existing resource efficient routing and adopt a new metric called maximum expected social welfare to assess the optimality of a potential route from a source to a destination. By studying the relationship between energy cost and stability, we successfully combine these two different metrics and design an optimal algorithm to find the optimal route. We also extend our model to incorporate retransmission and study the effect of the local quota and the global quota on the selection of the routing path. In the future, we will explore the effect of signal strength on stability, study the effects of the global quota theoretically, and analyze the effect of the node stability on the selection of the routing path.

## 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] L. Anderegg and S. Eidenbenz. Ad hoc-VCG: A truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents. In *Proceedings of ACM MOBICOM'03*, pages 245–259, 2003.
- [4] S. Biswas and R. Morris. ExOR: Opportunistic multi-hop routing for wireless networks. In *Proceedings of ACM SIGCOMM'05*, pages 133–144, 2005.
- [5] L. Buttyan and J. P. Hubaux. Nuglets: a virtual currency to stimulate cooperation in self-organized mobile ad hoc networks. Technical Report DSC/2001/001, EPFL-DI-ICA, January 2001.
- [6] D. S. J. D. 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.
- [7] N. Immorlica, D. Karger, E. Nikolova, and R. Sami. First-price path auctions. In *EC '05: Proceedings of the 6th ACM Conference on Electronic Commerce*, pages 203–212, 2005.
- [8] 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.
- [9] W. Lou and J. Wu. Double-covered broadcast (dcb): A simple reliable broadcast algorithm in manets. In *Proceedings of IEEE INFOCOM'04*, 2004.
- [10] A. Mas-Colell, W. Whinston, and J. Green. *Microeconomic Theory*. Oxford University Press, 1995.
- [11] Y. Qiu and P. Marbach. Bandwidth allocation in wireless ad hoc networks: A price-based approach. In *Proceedings of IEEE INFOCOM'03*, 2003.
- [12] D. J. Scott and A. Yasinsac. Dynamic probabilistic retransmission in ad hoc networks. In *International Conference on Wireless Networks*, pages 158–164, 2004.
- [13] 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.
- [14] 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.
- [15] 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.
- [16] 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.
- [17] S. Zhong, J. Chen, and Y. Yang. Sprite: A simple, cheatproof, credit-based system for mobile ad hoc networks. In *Proceedings of IEEE INFOCOM'03*, 2003.
- [18] 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.