

# Constrained Shortest Paths in Wireless Networks

Xiang-Yang Li\* Peng-Jun Wan\* Yu Wang\* Ophir Frieder\*

*Abstract*— In this paper, we address several QoS routing problems in wireless ad hoc networks. Due to mobility or limited battery power, a link between two nodes can last for a certain period, which is represented by a life parameter. On the other hand, each link has cost parameters which can represent the delay of this link, the transmission power needed to support this link, and so on. The life of a path is the shortest life of all links in this path; the cost of a path is the sum of the costs of all links. We first consider the problem of finding a path between a given pair of nodes with the maximum life while the cost of the path does not exceed a pre-specified bound. This problem can be solved in  $O((n \log n + m) \log n)$  time. Here  $n$  is the number of nodes and  $m$  is the number of links. A distributed version of the algorithm that is suitable for the wireless ad hoc networks is also presented. In addition, we study the problem of finding a path between a given pair of nodes with the maximum battery energy level while the cost of the path does not exceed a pre-specified bound. We also developed both centralized and distributed polynomial-time algorithms for this problem. Specifically, when the cost represents the transmission power needed to support the link, we give a distributed algorithm with time complexity  $O(n \log n)$  using  $O(n^2 \log n)$  total messages.

*Keywords*— Wireless ad hoc networks, delay, power consumption, constrained shortest path, network optimization.

## I. INTRODUCTION

**Ad Hoc Wireless Network.** Mobile wireless networking received significant attention over the last few years due to its potential applications in various situations. There are no wired infrastructures or cellular networks in *ad hoc* wireless network. Mobile nodes communicate with each other either through a single-hop transmission if the receiver node is within the transmission range of the sender, or through multi-hop wireless links by using intermediate nodes to relay the message. In other words, each node in the network also acts as a router, forwarding data packets for other nodes. A single transmission by a node can be received by all nodes within its transmission range. In this paper, we assume that each node can adjust its transmission power independently according to its neighborhood information to possibly reduce the energy consumption.

**Constrained Shortest Path.** Assume that the ad hoc wireless network is modelled by a graph  $G$ . Unicast routing is to discover a path connecting a given pair of mobile nodes; multicast routing is to find a spanning tree that connects a given set of mobile nodes. In this paper, we are mainly interested in unicast routing. We assume that there are some cost parameters  $f_0, f_1, f_2, \dots, f_k$  associated with the links or nodes of the network  $G$ . The cost

function  $f_i$  of a path  $p$  is dependent<sup>1</sup> on the cost value  $f_i$  of each link (or node) of the path  $p$ . A path  $p$  is called *feasible* if the costs  $f_i(p)$ ,  $1 \leq i \leq k$ , satisfy given  $k$  constraints<sup>2</sup>. Hereafter, the cost functions  $f_i$ ,  $1 \leq i \leq k$ , are called the *constraint functions*. The positive function  $f_0$  is denominated as the objective function. A feasible path  $p^*$  is called the *shortest path* if  $f_0(p^*)$  is the optimum among all feasible paths.

For example, to find a path with the longest node life while its energy consumption is no more than a given value  $E_0$ , we model the problem as follows. We define two cost functions  $f_0$  and  $f_1$ . For each link  $e$ , its cost function  $f_1(e)$  is the energy needed to support this link. For each node  $u$ , its cost function  $f_0(u)$  is its battery level. A path  $\pi(u, v) = u_0 u_1 \dots u_{h-1} u_h$ , where  $u = u_0$ ,  $v = u_h$ , is feasible if the total energy consumption  $\sum_{i=0}^{h-1} f_1(u_i u_{i+1})$  is at most  $E_0$ . A feasible path  $\pi^*(u, v) = u_0 u_1 \dots u_{h-1} u_h$  is optimum if  $\sum_{i=0}^{h-1} f_1(u_i u_{i+1}) \leq E_0$  and  $\min_{i=1}^{h-1} f_0(u_i)$  is the maximum among all feasible paths.

**Network Model.** We assume that each node has a low-power Global Position System (GPS) receiver, which provides the position information of the node itself. In addition, we also assume that each node knows the positions of all nodes that are within its transmission range<sup>3</sup>. We model a wireless network by a weighted directed graph  $G_t = (V, E)$ . Here  $V$  is the set of all mobile nodes, and edge  $(u, v) \in E$  if and only if the node  $v$  is in the transmission range of the node  $u$ . Hereafter, we call  $G_t$  the *transmission graph*.<sup>4</sup> We assume that  $G_t$  is *strongly connected*, i.e., there is a directed path from any node to any other node. A central challenge in the design of *ad hoc* networks is the development of dynamic routing protocols that can efficiently find routes between two communication nodes. In recent years, a variety of routing protocols [1], [2], [3], [4], [5], [6] targeted specifically for *ad hoc* environment have been developed. For the review of the state of the art routing protocols, see surveys by E. Royer and C. Toh [7] and by S. Ramanathan and M. Steenstrup [8].

**Path Cost: Power and Life.** Energy conservation is a critical issue in *ad hoc* wireless network for the node and network life, especially in the sensor networks, as the

<sup>1</sup>For example, the dependence could be the maximum (or the minimum) value of all link costs, or the summation of link costs.

<sup>2</sup>Examples of constraints are: the cost of the path is no more than a given value or not less than a given value.

<sup>3</sup>It is often assumed that every node has the same maximum transmission range  $R$ . A node  $v$  is within the transmission range of a node  $u$  if the Euclidean distance between  $u$  and  $v$  is at most  $R$ .

<sup>4</sup>It is also called *unit graph* when all nodes have the same largest transmission range.

\* Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60526. Email: {xli, wan, wangyu, ophir}@cs.iit.edu.

nodes are powered by batteries only. Each mobile node typically has a portable set with transmission and reception processing capabilities. To transmit a signal from a node to the other node, the power consumed by these two nodes consists of the following three parts. Firstly, the source node needs to consume some power to prepare the signal. Secondly, in the most common power-attenuation model, the power required to support the transmission between two nodes is dependent on their distance. Finally, when a node receives the signal, it needs consume some power to receive, store and then process that signal [9]. In this paper, we proposed several routing protocols that are optimized for minimizing (or bounding) the power consumed by all mobile nodes involved in one communication session. The cost of an edge  $(u, v)$  is then defined as the power consumed for transmitting signal from  $u$  to node  $v$  and possibly the energy consumed by node  $u$  and node  $v$  to process the signal.

Due to the mobility of the wireless node, a link between two nodes can last for only a certain time period, which is represented by the *link life* in this paper. The link life can be approximated from the moving speed and direction of two nodes if these information is known, or from the moving history of two nodes. Su *et al.* recently proposed some heuristics to predict the mobility of the wireless nodes by assuming some degree of regularity in the nodes' moving pattern [10], [11]. Given two nodes, assume that a route is constructed and each involved node stores its next relay node information in its local table. When a link of the constructed route is broken, we have to initiate the route-finding algorithm to construct another route to connect the source and the destination, or to find a path to connect the two end-nodes of the broken link. Therefore, it is always worth to construct a route such that every link of the route has a long duration period.

In addition, as the mobile nodes are powered by batteries only, each mobile node can function for only a certain period time also. Here, we assume that it is impossible to recharge the battery when it is drained out during operations. Therefore, the routing algorithm must avoid extensive using of some nodes to relay the signal. On the other hand, when it is power-efficient to use some other nodes to relay the signal from the source, it may be better to use relaying as long as it will not drain out the relaying nodes. Notice that there are two different objectives to be optimized by energy efficient routing protocols. One is to minimize the energy used by all nodes involved in one communication session. Using this approach, however, some nodes used for relaying maybe drained out if it is the critical node for several paths. The other objective is to maximize the network life. In other words, the routing should be distributed to all network nodes. Here the *overall life period* of the network is defined as the time period when the first mobile node is out of transmission energy since the initiation of the network.

**Our Results.** In this paper, we described centralized

and distributed position-based networking protocols optimized for energy consumption and network life in mobile wireless networks. We consider how to find the *shortest* path or the *widest* path between two nodes subject to various constraints. We consider the following cost functions of a path between two nodes: (1) the total number of relaying nodes used in the path; (2) the total energy consumed by all nodes in the path; (3) the minimum remaining life of the batteries of all nodes in the path; or (4) the least durable links in the path. Notice that the number of relaying nodes often corresponds to the delay of the path in the mobile network. The first two are the *length type* objectives and the last two are the *width type* objectives. Any of the above cost measurements could be used as the objective function to be optimized.

We first review some algorithms on how to find a path with the minimum number of hops while its transmission power cost is bounded; on how to find a path with the minimum energy cost while its link life (or battery energy level) is bounded from below by a pre-specified value. We then study the problems of finding a path connecting a given pair of nodes with the maximum link life (or battery energy level) while the energy cost of the path does not exceed a pre-specified bound. We give centralized algorithms to solve these problems in  $O((n \log n + m) \log n)$  time. In addition, we also present distributed algorithms to solve them in time  $O(n \log n)$ . When the power cost function of a link  $e$  is defined as  $\|e\|^\alpha$ ,  $2 \leq \alpha$ , we give a centralized algorithm to find the route with the maximum battery level while its energy consumption is bounded by a given value in time  $O(n \log^2 n)$ . A distributed version of the algorithm that is suitable for the wireless ad hoc networks is also presented.

The rest of the paper is organized as follows. In Section II, we first review some preliminary definitions and results related to the constrained shortest path routing. We then discuss several routing problems regarding energy conservation and mobility measurement and provide centralized and distributed algorithms in Section III. We conclude our paper in Section IV.

## II. PRELIMINARIES

### A. Energy Consumption Model

In the most common power-attenuation model, the signal power falls as  $\frac{1}{x^\alpha}$ , where  $x$  is the distance from the transmitter antenna and  $\alpha$  is a real constant between 2 and 5 dependent on the wireless transmission environment.<sup>5</sup> This is typically called the *path loss*. We always assumed that all receivers have the same power threshold for signal detection, which are then typically normalized to one. The

<sup>5</sup>To make this model meaningful, we always assume that the distance between any two nodes is at least one unit so the above model does not violate the energy-consumption law. We also assume that the unit of power and the unit of distance between nodes satisfy the path loss formula. Notice that in [12], they assume that the power falls as  $\frac{1}{(1+x)^\alpha}$ . Therefore, it is always meaningful for all  $x$ .

power required to support a link between two nodes  $u$  and  $v$  separated by distance  $x$  is therefore  $x^\alpha$ . By a simple geometry computing, it is easy to see that relaying signal between nodes may result in lower power consumption than communication over a large distance due to the non-linear power attenuation. For convenience, hereafter, we use  $\|uv\|$  to denote the Euclidean distance between two geometry nodes  $u$  and  $v$ .

For a simple illustration, consider three nodes  $s$ ,  $r$ , and  $d$  on the plane as in Figure 1. Assume that all three nodes use identical transmitters and receivers and  $\alpha = 2$ . The power to transmit a signal from  $s$  to  $d$  is therefore  $\|sd\|^2$ . If we use the node  $r$  to relay the signal, the total power used by the route  $srd$  is  $\|sr\|^2 + \|rd\|^2$ , which is less than  $\|sd\|^2$ . In other words, if  $s$  wants to send a message to any node  $d$  lying in the right side of the line  $l$ , relaying through the node  $r$  always consumes less power than directly transmitting to  $d$ .

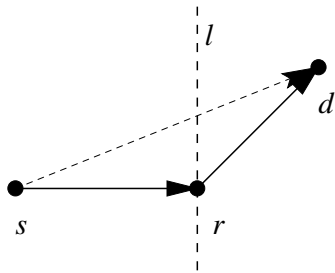


Fig. 1. Relaying through other node  $r$  consumes less power than directly transmitting from  $s$  to  $d$ .

There is also another source of power consumption that we must consider in addition to the path loss. When a node receives a signal from other nodes, it needs to consume some power to receive, store and then process that signal [9]. Hereafter, we will denote all power consumed at a node besides actually transmitting the signal by a real non-negative constant  $c$ . In this paper, most of our centralized and distributed algorithms assume that  $c$  is negligible, i.e.,  $c = 0$ .

### B. Minimum Energy Topology

Given two nodes in a mobile wireless network, one of the basic routing problem is to find a route connecting these two nodes such that the route consumes the least power. This problem can be solved by applying the shortest path algorithm, such as Dijkstra's algorithm implemented using Fibonacci heap, on the transmission graph  $G_t$  of the network. The graph  $G_t$  may have  $m = O(n^2)$  links. Thus, the fast centralized algorithms still have time complexity  $O(n^2)$ . However, this simple approach does not utilize any property of the path consuming the least power. Recently, Li and Wan [13] proposed an algorithm to solve this in  $O(n \log n)$  time when  $c = 0$ .

Given a node  $u$ , the transmission neighbors  $N_1(u)$  is the set of nodes that are within the transmission range of node

$u$ . Any node of  $N_1(u)$  could serve as an immediate relay node for the signal from  $u$ . However, it is not power-efficient that using all such nodes as candidates to directly relay the messages from  $u$  to other nodes. In other words, it is not necessary to check all neighbor nodes of  $u$  to find the next hop for node  $u$ . Given a node  $u$ , call a node  $v$  a *relay neighbor* of  $u$  if there is no node  $w$  such that it is power efficient to relay the signal from  $u$  to  $v$  using  $w$ . The directed graph connecting each node to its relay neighbors is called the *enclosure graph* [13], [9]. In [13], [9], they showed that the enclosure graph always contains the path, with the least power consumption, between any pair of mobile nodes. Li and Wan [13] then gave efficient localized algorithms to compute the enclosure graph. They also showed that the enclosure graph is a subset of the Gabriel graph over the set of wireless nodes if  $c = 0$ . Here a link  $uv$  belongs to the Gabriel graph if and only if the open circle using  $uv$  as a diameter does not contain any other nodes inside. It then implies that the enclosure graph is a planar graph with at most  $m \leq 3n - 6$  edges. The minimum energy topology of a set of mobile nodes is defined as the supergraph of the shortest paths of every pair of nodes.

### C. Problem Formulation

Assume that the wireless nodes are given as a finite point set  $V$  in a two-dimensional plane. Each node has the same largest transmission range, which is normalized to one. Each node also has the same transmission overhead cost  $c$ , which is also scaled accordingly. We model the wireless ad hoc network by a weighted unit disk graph  $G = (V, E, W)$ , where  $V$  is the set of wireless nodes, the set of links  $E$  contains all pairs of nodes  $(u, v)$  such that  $\|uv\| \leq 1$ . For each node  $u$ , let  $w(u)$  denote the remaining battery power of the node  $u$ . As we have described before, the power needed to support the link between two nodes  $u$  and  $v$  separated by distance  $r$  is  $r^\alpha + c$ , where  $2 \leq \alpha \leq 5$ . We then define the power cost  $p(e)$  for a link  $e$  as  $\|e\|^\alpha + c$ , where  $\|e\|$  is the Euclidean length of the link  $e$ . We also assume that each link  $e$  has a delay parameter  $d(e)$  to measure its transmission delay. We use  $l(e)$  to denote the life period of this link. This  $l(e)$  is usually computed from the mobility prediction of the wireless nodes. Taking the battery life into account, we also can define an adjusted link life  $l'(e) = \min(l(e), w'(u), w'(v))$ , where  $e = (u, v)$  and  $w'(u)$ ,  $w'(v)$  are the estimated time period that nodes  $u$ ,  $v$  can be supported by the batteries. For node  $u$ , assume its average transmission distance is  $r$  and the average transmissions per time unit is  $t$ , then we estimate  $w'(u) = \frac{w(u)}{l(r^\alpha + c)}$ .

Consider any unicast  $\pi$  from a node  $u \in V$  to another node  $v \in V$ :

$$\pi = p_0 p_1 \cdots p_{h-1} p_h, \text{ where } u = p_0, v = p_h.$$

Here  $h$  is the number of hops of the path  $\pi$ . The total *transmission power*  $p(\pi)$  consumed by this path  $\pi$  is defined

as

$$p(\pi) = \sum_{i=1}^h \|p_{i-1}p_i\|^\alpha + h \cdot c.$$

The *width*  $w(\pi)$  (i.e., the minimum remaining battery power of all nodes) of this path  $\pi$  is defined as

$$w(\pi) = \min_{0 \leq i \leq h} w(p_i).$$

The *life period*  $l(\pi)$  of this path  $\pi$  is defined as

$$l(\pi) = \min_{0 \leq i \leq h-1} l(p_i p_{i+1}).$$

The *delay*  $d(\pi)$  of this path  $\pi$  is defined as

$$d(\pi) = \sum_{0 \leq i \leq h-1} l(p_i p_{i+1}).$$

From now on, let  $n_G$  be the number of nodes of a graph  $G$ , and let  $m_G$  be the number of edges of a graph  $G$ . Given a node  $u$ , let  $N_G(u)$  be the set of nodes connected to node  $u$  in graph  $G$  and  $d_G(u)$  be the cardinality of  $N_G(u)$ , i.e., the degree of node  $u$  in  $G$ . When it is clear from the context, we will drop the label  $G$  from all notations.

#### D. Previous Results

The minimum energy routing problem was studied before. The simplest version is to minimize the total energy required to reach the destination, which can be done by applying any shortest path algorithm. However, previous methods merely consider minimizing or bounding the energy consumed by the route while the route satisfies some other given constraints. In [14], Chang and Tassiulas basically consider how to maximize the system lifetime given a set of sender nodes and receiver nodes, and the message generation rates by the sender nodes. They model it as the multicommodity flow problem, in which each commodity has its own destinations set. However, their work has limited applications because it needs to know the multicommodity flows before the routings are constructed. In our study, the routing is constructed for each online request.

Although finding the shortest path can be solved by several algorithms, finding a feasible path subject to two and more additive constraints had been shown to be a NP-Complete problem [15], [16], [17]. However, the problem can be solved in polynomial time if there is only one additive constraint in multiple constraints.

To cope with the NP-completeness of the two additive constraints shortest path problem, several heuristic based algorithms and quality guaranteed approximation algorithms had been proposed. For simplicity, we consider two nontrivial additive weights  $w_1(e)$  and  $w_2(e)$  defined for each link  $e$  first. Given two nodes  $u$  and  $v$ , we have to find a path  $\pi$  connecting  $u$  and  $v$  such that  $w_1(\pi) \leq c_1$

and  $w_2(\pi) \leq c_2$ . Here  $w_i(\pi)$ ,  $i = 1, 2$ , is the summation of the weight  $w_i(e)$  for every link  $e$  in the path  $\pi$ . Korkmaz and Krunz [18] proposed a randomized heuristic algorithm for finding  $\pi$  in  $O(n^2)$  time. However, their algorithm does not guarantee to find a feasible path even there exists one between the source and the destination. Korkmaz, Krunz and Tragoudas [19] then proposed an approximation algorithm that runs  $\log W$  iterations of Dijkstra's algorithm, where  $W$  is the upper bound of the longest path with respect to one of the link weights. The path  $\pi$  found by the algorithm satisfies that  $w_2(\pi) \leq c_2$  and  $w_1(\pi) \leq w_1(f) + \beta(w_2(f) - w_2(\pi))$ , where  $f$  is some feasible path and  $\beta$  is some real constant related to  $W$ .

One variation of finding a feasible route satisfying two constraints is to set one constraint as the objective function to be minimized. For example, we can find a path  $\pi$  with the minimum  $w_1(\pi)$  cost while  $w_2(\pi) \leq c_2$ . This problem is obviously also NP-hard problem. Hassin [20] and Philips [21] proposed two approximation algorithms for this problem. Unfortunately, both algorithms have some flaws. Then Hong, Chung and Park[22] gave a corrected version of the FPTAS that returns a path  $\pi$  such that  $w_2(\pi) \leq c_2$ , and  $w_1(\pi) \leq (1 + \epsilon)OPT$  in time  $O(mn\epsilon^{-1} + m^2 n^3 \log n)$  for any given positive real constant  $\epsilon$ . Here  $OPT$  is the optimum solution. Consequently, this algorithm can solve the two constrained path problem as long as that one of the constraints is not tight.

It is also often required to find a path satisfying one additive constraint such as energy consumption and one bottleneck constraint such as the width of the path. Wang and Crowcroft [16] basically gave an algorithm to find the shortest path whose width is at least a given bound. Their basic approach is to first eliminate all links that do not meet the width requirement and then apply any shortest path algorithm on the residue graph to compute the shortest path between the source and the destination. To the best of our knowledge, no algorithm has been proposed to find the path with the largest duration (or battery energy level) between two nodes and its energy consumption is no more than a predefined upper-bound. In other words, this problem can be viewed as the dual of the problem considered by Wang and Crowcroft [16].

In this paper, we mainly consider how to reduce the energy consumption and prolong the network life when routing messages between wireless nodes.

### III. ALGORITHMS

#### A. Min Hops Energy-bounded Path

We first consider how to find a route connecting two wireless nodes with the minimum number of hops while the energy consumed by the route is no more than a given bound. Notice that the number of hops of a path in ad hoc wireless networks often corresponds to the transmission delay of the path. The more hops, the larger delay of

the path. It has applications in multimedia systems when the delay of the path is more important than the energy consumed by the path. For example, sending the information in time is always the most important in the battlefield. Given the source node  $s$ , the destination node  $t$  and the energy consumption bound  $E_0$ , the minimum-hops energy-bounded (MHEB) routing problem is defined as following.

$$\begin{aligned} & \min h \\ & \text{s.t. } \sum_{i=0}^{h-1} p(v_i v_{i+1}) \leq E_0 \\ & \text{where } v_0 = s, v_h = t. \end{aligned}$$

Notice that here  $p(v_i v_{i+1})$  is the energy consumed to support the link  $v_i v_{i+1}$ . It is typically defined as  $\|v_i v_{i+1}\|^\alpha + c$ , where  $\alpha$  is the environmental propagation constant and  $c$  is the transmission overhead.

### A.1 Centralized Algorithm

The centralized version of the problem had been studied long time ago. For the completeness of our presentation, we still include the algorithm here. To find a path with the minimum number of hops whose total energy consumption is no more than a bound  $E_0$ , the basic idea is similar to the breadth first search. We use  $d_i(k)$  to denote the minimum energy consumed from node  $s$  to node  $i$  by passing exactly  $k$  edges. Then our algorithm has the following steps.

*Algorithm 1: MinHops\_BoundEnergy\_C( $G, E_0$ )*

1. Set  $d_i(k) = \infty$ , for  $1 \leq i \leq n$  and  $0 \leq k \leq n - 1$ ; set  $d_s(0) = 0$ . Let  $h = 1$ .
2. While the optimum path is not found for each node  $i$ , update

$$d_i(h) = \min(d_i(h), \min_{j \in N_G(i)} (d_j(h-1) + p(i, j))).$$

If  $d_t(h) \leq E_0$ , then the shortest path has been found with  $h$  edges. Otherwise,  $h = h + 1$ .

It is not difficult to show the correctness of the above algorithm based on dynamic programming. The above algorithm terminates in time  $O(mh)$ , where  $h$  is the least number of links of all feasible paths.<sup>6</sup> The above algorithm is therefore output sensitive. Remember that the unit disk graph  $G$  may have  $m = O(n^2)$  links. It implies that the above algorithm has time complexity  $O(n^2h)$  when applied to the unit disk graph  $G$ . However, if we apply the algorithm to the enclosure graph (or Gabriel graph when  $c = 0$ ) instead of the unit disk graph  $G$ , it will return a solution with  $h'$  links in  $O(nh')$  time. Notice that here it is possible that  $h' > h$ . Notice that the Gabriel graph is always a subgraph of the Delaunay triangulation and it can be constructed in  $O(n \log n)$  time. It then implies the overall time complexity of the above algorithm is  $O(n \max(h', \log n))$  when the Gabriel graph is used.

<sup>6</sup>Here a path is feasible if its energy consumption is no more than  $E_0$ .

### A.2 Distributed Algorithm

For wireless ad hoc networks, it is difficult to gather all nodes and links information by one central node. We then discuss a distributed implementation of the previous algorithm, which is suitable for hop-by-hop routing.

*Algorithm 2: MinHops\_BoundEnergy\_D( $G, E_0$ )*

1. Each node  $i$  sets  $d(h) = \infty$ , for  $0 \leq h \leq n - 1$ . The source node  $s$  sets  $d(0) = 0$ . Let  $h = 1$ .
2. While the optimum path is not found, each node  $i$  updates

$$d(h) = \min(d(h), \min_{j \in N(i)} (d_j(h-1) + p(i, j))),$$

where  $d_j(h-1)$  is retrieved from node  $j \in N(i)$ . If the destination node  $t$  has  $d(h) \leq E_0$ , then a shortest path with  $h$  edges has been found. Otherwise, each node sets  $h = h + 1$ .

Assume that we run the above distributed algorithm in a synchronized model. It is then not difficult to show that each node has to update the array  $d$  for the first  $h$  items. It is the property of the shortest path that the above algorithm is guaranteed to converge and terminate. The computed final path is also loop-free. Again, to improve the time complexity of the above algorithm, we suggest to run the algorithm on the enclosure graph. The communication complexity of the above distributed algorithm is  $O(nh)$  because each node only need broadcast its value  $d(h)$  once in each round.

### B. Min Energy Life-bounded Path

In this section, we consider how to find a path from a node  $s$  to a given node  $t$  that consumes the minimum energy while the duration of the path is no less than a given bound  $D_0$ . The life bounded route assures that the communication between  $s$  and  $t$  will not be interrupted because of the node movement. In addition, the path consuming the least power among all such feasible paths will prolong the network life. Wang and Crowcroft [16], [17] already proposed an elimination-based algorithm to solve a similar problem. For the completeness of the presentation, we still include the version suitable for wireless network environment here. In addition, we show how to find the optimum routing efficiently when  $c = 0$ . Given the source node  $s$ , the destination node  $t$  and the path duration bound  $D_0$ , the minimum-energy life-bounded (MELB) routing problem is defined as following.

$$\begin{aligned} & \min \sum_{i=0}^{h-1} p(v_i v_{i+1}) \\ & \text{s.t. } \min_{i=0}^{h-1} l(v_i v_{i+1}) \geq D_0 \end{aligned}$$

where  $v_0 = s, v_h = t$ .

## B.1 Centralized Algorithm

The approaches of [16], [17] are based on the following simple fact: a path has a duration no less than  $D_0$  if and only if  $l(e) \geq D_0$  for each link  $e$  in the path. Hence any link  $e$  with  $l(e) < D_0$  can not be in the optimum routing from the source  $s$  to the destination  $t$ . Consequently, we can first prune out every link  $e$  with  $l(e) < D_0$ , and then apply the shortest path algorithm on the residue graph. Notice that we do not have to explicitly prune out all links  $e$  with  $l(e) < D_0$ . It can be done during the relaxing step of Dijkstra's algorithm. The optimum routing can be computed in  $O(n \log n + m)$  time if we apply Dijkstra's shortest path algorithm, where  $m$  is the total number of links in the unit disk graph  $G$ . See Wang and Crowcroft [16], [17] for a more detailed algorithm.

Notice that the unit disk graph  $G$  could have  $m = O(n^2)$  edges. Therefore a simple approach, by directly applying the algorithm given in [16], [17] to the unit disk graph  $G$ , could have time complexity  $O(n^2)$ . However, we can improve it as follows when  $c = 0$ . Let  $H$  be the residue graph by eliminating all links with  $l(e) < D_0$ . We then further simplify the residue graph  $H$  by removing every link  $(u, v)$  such that there are links  $(u, w)$ ,  $(w, v)$  and node  $w$  is inside the open disk using  $uv$  as a diameter.

## B.2 Distributed Algorithm

The above centralized algorithm has no significant difference with the classic shortest path algorithm. One can run the distributed shortest path algorithm on the enclosure graph to find the optimum route with the least energy consumption. Li and Wan [13] have given a localized algorithm to compute the enclosure graph efficiently<sup>7</sup>. Here, we give another algorithm to find the relaying neighbors of a node when  $c = 0$ .

Let  $disk(u, v)$  be the disk with diameter  $(u, v)$ . Then the *Gabriel graph*  $GG(S)$  of a set of nodes  $V$  has an edge  $(u, v)$  if and only if  $disk(u, v)$  does not contain any other node from  $V$ . It is well known that the Gabriel graph contains the Euclidean minimum spanning tree of  $V$ . We first prove the following lemma, which leads us to another efficient distributed algorithm.

**Theorem 3:** The Gabriel graph  $GG(V)$  contains the minimum energy-consumption path connecting any two nodes when the overhead cost  $c = 0$ .

**PROOF.** Consider any pair of nodes  $s$  and  $t$  and a path  $\pi(s, t)$ , connecting  $s$  and  $t$ , that consumes the least energy. Consider any link  $(u, v)$  in  $\pi(s, t)$ . See Figure 2. Assume that  $disk(u, v)$  contains a node  $w$  inside. Notice that  $|uw|^\alpha + |wv|^\alpha < |uv|^\alpha$  for any  $\alpha \geq 2$ . It implies that the path by substituting link  $(u, v)$  by two links  $(u, w)$  and  $(w, v)$  will consume less power than  $\pi(s, t)$ . Consequently,  $disk(u, v)$  does not contain any node inside. Thus, the path  $\pi(s, t)$  is contained in the Gabriel graph.  $\square$

<sup>7</sup>A distributed algorithm is called *localized* if every node only uses the information about itself, its transmission neighbors and at most

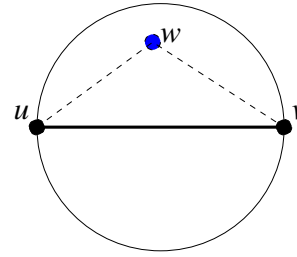


Fig. 2. The GG contains the shortest path if  $c = 0$ .

The edges connected to a node  $u$  in the Gabriel graph  $GG(V)$  can be computed in time  $O(d \log d)$ . Here  $d$  is the degree of the node  $u$  in the transmission graph  $G_t$ . A simple approach is to first construct the Delaunay triangulation of all transmission neighbors of  $u$  and  $u$ . Then eliminate some redundant edges.

We define the *Life-bounded Gabriel Graph (LGG)* over a set of nodes  $S$  as follows. Graph  $LGG_{D_0}(S)$  has an edge  $(u, v)$  if and only if  $l(u, v) \geq D_0$  and  $disk(u, v)$  does not contain any other node  $w$  such that  $l(u, w) \geq D_0$  and  $l(w, v) \geq D_0$ .<sup>8</sup> If it is clear from the context, we will drop the  $D_0$  from  $LGG_{D_0}(S)$  hereafter. Then by a similar argument, given any two nodes  $s$  and  $t$ , the graph  $LGG_{D_0}(S)$  contains the minimum energy path connecting  $s$  and  $t$  whose life is no less than  $D_0$ . In addition, each mobile node can compute its life-bounded relaying neighbors in  $LGG$  in time  $O(d \log d)$ . Consequently, we can run the distributed Bellman-ford algorithm on the computed life-bounded Gabriel graph  $LGG$  to solve the minimum-energy life-bounded routing problem.

### Algorithm 4: MinEnergy\_LifeBound\_D

1. Each node  $u$  prunes out links  $e = (u, v)$  with  $p(e) < D_0$ ,
2. Construct the links of  $LGG_{D_0}$  incident on  $u$  locally based on the remaining links.
3. Run the distributed shortest path algorithm on  $LGG_{D_0}$  to find the path with the least energy consumption connecting the source  $s$  and the destination  $t$ .

## C. Min Energy Width-bounded Path

In this section, we consider how to find a path from a node  $s$  to a given node  $t$  that consumes the minimum energy while the remaining battery energy of the path is at least  $W_0$ . This is important because such path will guarantee that the transmission will not be broken because some nodes are out of battery. Given the source node  $s$ , the destination node  $t$  and the battery requirement bound  $W_0$ , the minimum-energy battery-bounded (MEBB) routing

a constant number of additional information.

<sup>8</sup>Then we call node  $v$  a *life-bounded relaying neighbor* of  $u$ .

is defined as following.

$$\begin{aligned} \min \quad & \sum_{i=0}^{h-1} p(v_i v_{i+1}) \\ \text{s.t.} \quad & \min_{i=1}^{h-1} w(v_i) \geq W_0 \\ \text{where } & v_0 = s, v_h = t \end{aligned}$$

Notice that a straightforward solution is to define the duration  $l'(e)$  for each link  $e$ , then apply the algorithm proposed in the above subsection III-B. Here the life  $l'(e)$  is defined as  $\min(w(u), w(v))$ , where  $u, v$  are the two end nodes of the link  $e$ . The time complexity of this approach is  $O(n \log n + m)$ , where  $m$  is the number of edges of  $G_t$ .

It can be solved more efficiently based on the following simple fact: a path has battery energy no less than  $W_0$  if and only if  $w(u) \geq W_0$  for every node  $u$  in the path. Consequently, we can first prune out every node  $u$  with  $w(u) < W_0$ , and then apply the shortest path algorithm to the residue graph. We run the shortest path algorithm on the enclosure graph constructed after pruning out infeasible nodes  $u$  with  $w(u) < W_0$ . It is well-known that the Gabriel graph is a planar graph and it can be constructed in time  $O(n_1 \log n_1)$ . Here  $n_1$  is the number of nodes  $u$  with  $w(u) \geq W_0$ . For a set of mobile wireless nodes, the path consuming the least power is always from the transmission graph  $G_t$ . Therefore, we only have to consider the intersection graph, denoted by  $GG_t$ , of the transmission graph  $G_t$  with the Gabriel graph  $GG$ . Similar to Lemma 3, we can prove that the graph  $GG_t$  contains the minimum energy topology. It implies that the minimum energy battery-bounded path connecting two nodes can be constructed in time  $O(n \log n)$ .

The distributed algorithm, to find the path consuming the minimum energy while each node in the path has a battery energy level at least  $W_0$ , has the following steps. First, the source broadcasts the required energy level  $W_0$ . Call a node  $u$  feasible if  $w(u) \geq W_0$ . Then, each feasible node  $u$  computes all links  $(u, v) \in GG_t$ . This can be done in time  $O(d \log d)$ , where  $d$  is the degree of  $u$  in  $G_t$ . Then each feasible node  $u$  runs the distributed Bellman-ford algorithm using the links found in the second step. This approach will find the optimum path in  $O(n)$  rounds.

#### D. Max Life Energy-bounded Path

For the wireless ad hoc networks, energy conservation is a critical issue as the nodes are often powered by the batteries only. To make the network system last longer, every routing path between two nodes should consume less energy. Notice that the path will often be broken because of the mobile nodes movement. When the route is broken, we have to rediscover the path. Thus, it is desirable to route the message over the path with the largest duration while its energy consumption is not too large.

We then consider how to find a path from a node  $s$  to a node  $t$  with the maximum life duration while the energy consumed by this path is no more than  $E_0$ . Notice that this problem can be viewed as the dual version of the problem considered by Wang and Crowcroft [16], [17]. Given the source node  $s$ , the destination node  $t$  and the maximum energy consumption requirement  $E_0$ , the maximum-life energy-bounded (MLEB) routing is then defined as following.

$$\begin{aligned} \max \quad & \min_{i=0}^{h-1} l(v_i v_{i+1}) \\ \text{s.t.} \quad & \sum_{i=0}^{h-1} p(v_i v_{i+1}) \leq E_0 \\ \text{where } & v_0 = s, v_h = t \end{aligned}$$

Let  $\chi$  be all paths from  $s$  to  $t$  with energy consumption no more than  $E_0$ . In other words,  $\chi$  is the set of all feasible paths. The question is then to find the feasible path with the largest life period.

Our solution is based on pruning links. We first guess the maximum duration  $D_0$ , then prune out all links  $e$  with  $l(e) < D_0$ . We then apply the shortest path algorithm on the residue graph. If we find a path from the source to the destination with energy consumption no more than  $E_0$ , then it implies that the maximum life of all feasible paths  $\chi$  is at least  $D_0$ . Otherwise, we know that the maximum life is less than  $D_0$ . Then a binary search using the shortest path algorithm as oracle will narrow down the maximum life of all feasible paths.

#### D.1 Centralized Algorithm

We first present the centralized algorithm to find the path with the largest duration while its energy consumption is no more than  $E_0$ .

*Algorithm 5: MaxLife\_BoundEnergy\_C*

1. Sort the life durations of all  $m$  links. Let  $L_1, L_2, \dots, L_{b-1}, L_b$  be all distinguishable durations with  $L_i < L_j$  if  $i < j$ .
2. Set  $u = b$ , and  $l = 1$ .
3. Let  $L = L_{\lceil \frac{l+u}{2} \rceil}$ . Then prune out all links of  $G_t$  with duration less than  $L$ . Let  $G_L$  be the residue graph.
4. Apply the shortest path algorithm to compute the path  $\pi$  with the minimum energy consumption  $E_L$  from  $s$  to  $t$  on graph  $G_L$ .
5. If  $l = u$ , then return the  $\pi$  as the optimum path with the maximum life duration whose energy consumption is no more than  $E_L$ . The problem has an optimum solution if and only if  $E_L \leq E_0$ .
6. Otherwise if  $E_B \leq E_0$ , then set  $l = \lceil \frac{l+u}{2} \rceil$  else set  $u = \lfloor \frac{l+u}{2} \rfloor$  and go to the above step 3.

If we apply the Dijkstra's algorithm implemented using the Fibonacci heap to compute the shortest path, the above

algorithm has time complexity  $O((n \log n + m_p) \log n)$ . And the space complexity is  $O(m_p)$ . Here  $m_p$  is the maximum number of edges of all residue graphs, which could be as large as  $O(n^2)$ . Consequently, the worst time complexity of the above algorithm is  $O(n^2 \log n)$ .

## D.2 Distributed Algorithm

All above discussions are concentrated in giving a centralized algorithm to find the most durable path whose energy consumption is bounded from above by a given value. In this section, we propose a distributed algorithm which is suitable for the hop-by-hop routing. One approach is as follows. We store a set of life and energy consumption pair  $(L_i, E_i)$  at each node  $v$ . Pair  $(L_i, E_i)$  denotes that  $E_i$  is the least known energy needed by all paths from the source node to  $v$  with duration at least  $L_i$ . Notice that if  $L_i < L_j$  then we always have  $E_i \leq E_j$ .

*Algorithm 6:* MaxLife\_BoundEnergy\_D

1. Each node  $v$  sets  $E_i = \infty$  for duration  $L_i$  of a link incident on  $v$ .
2. Repeat the following steps until node  $t$  finds the optimum solution
3. For all neighbor nodes  $u \in N(v)$
4. Node  $v$  receives information  $(L_i, E_i(u))$  from  $u$
5. Node  $v$  computes, for all  $L_i \leq l(u, v)$ ,

$$E_i = \min(E_i, E_i(u) + p(u, v)).$$

If  $L_i > l(u, v)$ , node  $v$  updates the pair  $(l(u, v), E)$  as

$$E = \min(E, E_i(u) + p(u, v)).$$

6. If node  $t$  finds the optimum solution, then terminate.

Similar to the proof of the Bellman-ford algorithm, the above algorithm converges and the computed optimum path is loop-free. Notice that each node does not have to store all pairs  $(L_i, E_i)$  for all duration links. However, in worst case, some nodes could store almost  $O(m)$  pairs. The above algorithm could have large space cost. Actually, it is reasonable to see why: the above algorithm actually computes the maximum duration for all energy consumption bounds.

The other approach is to let the source node orchestrate the centralized algorithm as follows. First, the source node broadcasts the guessed maximum duration  $L_g$  to all mobile nodes. Then all nodes together run the distributed Bellman-ford algorithm to compute the path consuming the least power with life at least  $L_g$ . The destination node sends back the minimum energy consumption to the source node. The source node runs the binary search method to narrow down the optimum life duration of all paths whose energy consumption is no more than  $E_0$ . Notice that the Bellman-ford algorithm terminates in average  $O(n)$  rounds. It is then not difficult to show that the above approach finds a path with the maximum life while its energy consumption is no more than  $E_0$  in  $O(n \log n)$  rounds.

## E. Max Width Energy-bounded Path

To make the network system last longer, a routing path connecting two mobile nodes should consume less energy and the routing should avoid drain out some mobile nodes used for relaying. Therefore, it is worthwhile to find a path whose energy consumption is at most some given value, while the nodes used for relaying the message have the most remaining battery energy. Notice that this problem is different from the problem considered by previous subsection. Previous one is to find the maximum of minimum link-life, this one is to find the maximum of minimum node-life. Given the source node  $s$ , the destination node  $t$  and the maximum energy consumption requirement  $E_0$ , the maximum-battery energy-bounded (MBEB) routing is defined as following.

$$\begin{aligned} \max \quad & \min_{i=1}^{h-1} w(v_i) \\ \text{s.t.} \quad & \sum_{i=0}^{h-1} p(v_i v_{i+1}) \leq E_0 \\ & \text{where } v_0 = s, v_h = t \end{aligned}$$

Similar to the question discussed in previous subsection, one simple solution is to prune out nodes with less remaining battery. We first guess the optimum energy level  $L_0$ , then prune out all nodes  $u$  with  $w(u) < L_0$ . We then apply any shortest-path algorithm on the residue graph. If we find a path from the source to the destination with energy consumption no more than  $E_0$ , then it implies that the maximum of the minimum battery life is at least  $L_0$ . Otherwise, we know the optimum battery life is less than  $L_0$ .

## E.1 Centralized Algorithm

We first study the centralized algorithm to find the path with the most remaining battery energy while its energy consumption is no more than  $E_0$ .

*Algorithm 7:* Maxwidth\_BoundEnergy\_C

1. Sort the battery energy levels of  $n$  nodes. Let  $B_1, B_2, \dots, B_{b-1}, B_b$  be all distinguishable values with  $B_i < B_j$  if  $i < j$ .
2. Set  $u = b$ , and  $l = 1$ .
3. Let  $B = B_{\lceil \frac{l+u}{2} \rceil}$ , and prune out all mobile nodes of  $G_t$  with battery energy level less than  $B$ . Let  $G_B$  be the residue graph.
4. Apply the shortest path algorithm to compute the path  $\pi$  with the minimum energy consumption  $E_B$  from  $s$  to  $t$  on graph  $G_B$ .
5. If  $l = u$ , then return the  $\pi$  as the optimum path with the maximum width whose energy consumption is no more than  $E_B$ . The problem has an optimum solution if and only if  $E_B \leq E_0$ .
6. If  $E_B \leq E_0$ , then set  $l = \lceil \frac{l+u}{2} \rceil$ ; otherwise set  $u = \lceil \frac{l+u}{2} \rceil$  and go to the above step 3.



If we apply the Dijkstra's algorithm implemented using the Fibonacci heap to compute the shortest path, the above algorithm has time complexity  $O((n \log n + m_p) \log n)$ . And the space complexity is  $O(m_p)$ . Here  $m_p$  is the maximum number of edges of all residue graphs, which could be as large as  $O(n^2)$ . However, if we use the enclosure graph (or Delaunay triangulation) instead of the residue graph, the time complexity will be improved to  $O(n \log^2 n)$  as the enclosure graph (or Delaunay triangulation) can be constructed in  $O(n \log n)$  time and it has only  $O(n)$  edges if the overhead  $c = 0$ .

## E.2 Distributed Algorithm

In this section, we discuss a distributed implementation of the centralized algorithm which is suitable for the mobile ad hoc routing. Both approaches used for finding the path with the maximum link life and bounded energy consumption could be used to solve this problem. However, we found that the second approach, using the source node to coordinate the protocol, is more time efficient. It operates as follows. First, the source node  $s$  guesses a battery level  $L_g$ , which could be optimum. Then  $s$  starts to run the distributed Bellman-ford algorithm to compute the path consuming the least power with life at least  $L_g$ . The destination node  $t$  sends back the minimum energy consumption to  $s$ . Notice that here, the nodes with battery level less than  $L_g$  will not participate the Bellman-ford algorithm. Then  $s$  runs the binary search method to narrow down the optimum battery life. It is not difficult to show that the above approach finds the optimum solution in  $O(n \log n)$  rounds. To speed up the Bellman-ford algorithm, each mobile nodes only process the information received from the neighbors in the enclosure graph (or Delaunay triangulation). It is showed in [13] that the neighborhood information can be decided efficiently by a localized approach.

Assume the optimum battery level is  $L_{opt}$ . Then it is not difficulty to show that all nodes with energy level larger than  $L_{opt}$  will almost always participate the Bellman-ford algorithm. However, the nodes with energy level less than  $L_{opt}$  will only participate a few rounds of Bellman-ford algorithm. So the rounds algorithm itself also avoids to drain out the batteries with less energy.

## IV. CONCLUSION

In this paper, we address several QoS routing problems in wireless ad hoc networks. We first considered how to find a path with the minimum number of hops while its transmission power cost is bounded. Secondly, we consider how to find a route connecting two nodes with the minimum energy among all paths whose life (or width) is bounded from below. Thirdly, we considered finding a path connecting a given pair of nodes with the maximum life (or battery energy) while the energy cost of the path does not exceed a pre-specified bound. Distributed versions of the algorithms that are suitable for the wireless ad hoc networks are also

Algorithms	Centralized	Distributed
MinHops BndEnergy	$O(mh)$	$O(h)$
MinEnergy BndLife	$O(m + n \log n)$	$O(n)$
MinEnergy BndWidth	$O(n \log n)$	$O(n)$
MaxLife BndEnergy	$O((m + n \log n) \log n)$	$O(n \log n)$
MaxWidth BndEnergy	$O(n \log^2 n)$	$O(n \log n)$

TABLE I  
ALGORITHMS SUMMARIATION.

presented. Notice that the algorithms presented in this paper can be combined together to find a path between two given pair of nodes with minimum links subject to the two constraints: the life of the path is no less than a given lower bound, while the cost of the path does not exceed a pre-specified bound. When two and more additive constraints are presented, the problem is already known to be NP-complete. We summarize the results in the following table I.

We showed that the maximum width energy-bounded path can be found in time  $O(n \log^2 n)$  using a centralized approach. While our algorithm finds the maximum life energy-bounded path in time  $O((n \log n + m) \log n)$ , which could be as large as  $O(n^2 \log n)$ . Let graph  $G(D)$  be the supergraph of all paths consuming the least energy while its life is bounded from below by a given value  $D$ . We would like to study the property of the shortest path with bounded life duration. If we can prove that  $G(D)$  is planar graph or has  $O(n)$  edges, and this graph can be constructed in time  $O(n \log n)$ , then the maximum-life energy-bounded routing problem can also be solved in  $O(n \log^2 n)$  time. We leave it as an open problem whether the graph has  $O(n)$  edges and it can be constructed in time  $O(n \log n)$ .

## REFERENCES

- [1] J. Broch, D. Johnson, and D. Maltz, "The dynamic source routing protocol for mobile ad hoc networks," 1999.
- [2] S. Murthy and J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks," *ACM Mobile Networks and Applications Journal, Special issue on Routing in Mobile Communication Networks*, vol. 1, no. 2, 1996.
- [3] V. Park and M. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in *IEEE Infocom*, 1997.
- [4] C. Perkins, "Ad-hoc on-demand distance vector routing," in *MILCOM '97*, Nov. 1997.
- [5] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing," in *In Proc. of the ACM SIGCOMM, October*, 1994.
- [6] P. Sinha, R. Sivakumar, and V. Bharghavan, "Cedar: Core extraction distributed ad hoc routing," in *Proc. of IEEE INFOCOM*, 1999.
- [7] E. Royer and C. Toh, "A review of current routing protocols for ad-hoc mobile wireless networks," *IEEE Personal Communications*, Apr. 1999.
- [8] S. Ramanathan and M. Steenstrup, "A survey of routing techniques for mobile communication networks," *ACM/Baltzer Mobile Networks and Applications*, pp. 89-104, 1996.

- [9] Volkan Rodoplu and Teresa H. Meng, "Minimum energy mobile wireless networks," in *Proceedings of the 1998 IEEE International Conference on Communications, ICC'98*, 1998, vol. 3.
- [10] W. Su, S. Lee, and M. Gerla, "Mobility prediction and routing in ad hoc wireless networks," *International Journal of Network Management*, 2000.
- [11] W. Su, S. Lee, and M. Gerla, "Mobility prediction in wireless networks," in *Proceedings of the IEEE Military Communications Conference (MILCOM)*, 2000.
- [12] I. Stojmenovic and X. Lin, "Power-aware localized routing in wireless networks," in *IEEE Int. Parallel and Distributed Processing Symp.*, May 2000.
- [13] Xiang-Yang Li and Peng-Jun Wan, "Constructing minimum energy mobile wireless networks," 2001.
- [14] Jae-Hwan Chang and Leandros Tassiulas, "Energy conserving routing in wireless ad-hoc networks," in *Infocom*, 2000.
- [15] M. R. Garey and D. S. Johnson, *Computers and Intractability*, W.H. Freeman and Co., NY, 1979.
- [16] Zheng Wang and Jon Crowcroft, "Bandwidth-delay routing algorithm," in *Proceedings of IEEE Globecom*, 1995, pp. 2129–2133.
- [17] Zheng Wang and Jon Crowcroft, "Quality-of-service routing for supporting multimedia applications," *Journal of Selected Areas in Communication*, vol. 14, no. 7, pp. 1228–1234, 1996.
- [18] Turgay Korkmaz and Marwan Krunz, "A randomized algorithm for finding a path subject to multiple qos constraints," in *Proceedings of IEEE Globecom*, 1999, pp. 1694–1698.
- [19] Turgay Korkmaz, Marwan Krunz, and Spyros Tragoudas, "An efficient algorithm for finding a path subject to two additive constraints," 1999.
- [20] R. Hassin, "Approximation schemes for the restricted shortest path problem," *Math. Oper. Res.*, vol. 17, 1992.
- [21] C. A. Phillips, "The network inhibition problem," in *Proc. 25th Ann. ACM Symp. on Theory of Comp.*, ACM, 1993, pp. 776–785.
- [22] Sung-Pil Hong, Sung-Jin Chung, , and Bum Hwan Park, "On "strongly" and fully polynomial approximation schemes for restricted shortest path problem," in *ISMP 2000*, 200.