On-Line Disjoint Path Routing for Network Capacity Maximization in Ad Hoc Networks

Weifa Liang
Department of Computer Science
Australian National University
Canberra, ACT 0200, Australia

Yuzhen Liu
School of Computer Science
Wuhan University
Wuhan, 430072, P. R. China

Xiaoxing Guo
Department of Computer Science
Australian National University
Canberra, ACT 0200, Australia

***Abstract*— In this paper we consider on-line disjoint path routing in energy-constrained ad hoc networks. The objective is to maximize the network capacity, i.e., maximize the number of messages routed successfully by the network without any knowledge of future disjoint path connection request arrivals and generation rates. We first present two on-line algorithms for the problem. One is based on maximizing the network lifetime and another is based on an exponential function of energy utilization at nodes. We then conduct extensive experiments by simulations to analyze the performance of the proposed algorithms. The experimental results show that the proposed algorithms outperform those existing algorithms that do not take into account the power load balancing among the nodes.**

Keywords: *Wireless communication network, on-line algorithm, power awareness, ad hoc networks, energy consumption optimization, disjoint path routing, load-balanced routing, network capacity.*

I. Introduction

In recent years multi-hop wireless ad hoc networks (ad hoc network for short) have been receiving significant attention due to their potential applications from civil to military domains. An ad hoc network consists of a collection of mobile nodes equipped with energy limited batteries. The communication between two mobile nodes in such a network can be either in a single hop transmission where both nodes are within the transmission ranges of each other or in a multi-hop transmission where the message is relayed by intermediate mobile nodes. In general, nodes in ad hoc networks are mobile as well, though in this paper we primarily concern relatively static ad hoc networks, a prevalent example of which are the sensor networks. The motivations for finding energy-efficient disjoint paths in ad hoc networks are as follows.

On one hand, there is a great need for the reliability in ad hoc networks. This need stems from the unpredictable nature of the wireless environment, which unlike its wired counterparts, is more easily prone to either link failures due to channel fading or obstructions or node failures due to node power expired or node mobility. As a result, the routing path will break down. To increase the routing resiliency against link or/and node failures, one way is to route a message along multiple disjoint paths simultaneously. Thus, the destination node is still able to receive the message even there is only one survivable routing path available. This is especially true in real-time data transmission, whereby if routing along a single path, just one node or link failure suffices to cause path failure and transmission interruption. On the other hand, mobile nodes, especially sensors, tend to use small batteries for energy supply that are in many instances non-replenishable. While wireless communication consumes significant amount of battery power [7], the limited battery lifetime imposes a strict constraint on the network performance. Therefore, energy conservation in such networks is a vital factor in prolonging the network lifetime.

A. Related Work

Energy-aware routing optimization has been addressed in recent years. Most existing work focuses on broadcast/multicast routing with minimizing the total energy consumption [23], [2], [12], [3], [22]. However, in many practical applications, the general interest is not only to minimize the total transmission energy consumption but also to maximize the lifetime at nodes because the failure of a node (running out of its battery) may lead to the network partitioned and any further service will be interrupted. To avoid the extinction of nodes, any energy-efficient routing algorithm should distribute the transmission energy load evenly among the nodes to prolong the network lifetime. This leads to the *network lifetime* concept, which is the time of the first node failure [5]. There have been several studies aiming to prolong the network lifetime [5], [4], [8]. For example, Chang and Tassiulas [5], [4] proposed maximizing network lifetime for on-line unicast through avoiding using low power nodes and choosing a shortest path in terms of the energy consumption if the packet rate in the network is given in advance. Kang and Poorvendran [8] considered maximizing the network lifetime broadcast tree problem. Gupta and Kumar [6] discussed the critical power at which a node needs to transmit in order to ensure the network is connected. In reality, for a given connection request, either minimizing the transmission energy consumption or maximizing the network lifetime for the request is insufficient. The better way is to take both of them into consideration, and there are a number of studies of broadcast requests that tradeoff the two optimization objectives [24], [8]. A closely related study is that Li et al [11] considered an on-line unicast problem, which aims to maximize the network lifetime while keep the total energy consumption of each connection request bounded. Liang and Yang [13] provided an improved algorithm for the problem, which improves the running time in [11]. Sankar and Liu [18] recently provided an analytical solution for the problem, using the multicommunity flow approach. Kar et al [9] re-examined the on-line unicast problem by introducing the *network capacity* concept, which aims to maximize the total number of connection messages successfully routed by the network without any knowledge of future message arrivals and generation rates. They devised an on-line algorithm for the problem with a guaranteed competitive ratio, using the ideas developed for wired networks [16].

Although several multiple disjoint path routing protocols for ad hoc networks have been proposed [14], [17], [10], [15], none of these protocols incorporate energy efficiency into the design of the protocols. Srinivas and Modiano [19] first considered the minimum-energy node/edge-disjoint path problem in ad hoc networks with an objective to minimize the total transmission energy consumption for each node/edge-disjoint path connection request. They provided an algorithm for finding minimum-energy k node-disjoint paths with $k \geq 2$ and an algorithm for finding minimum-energy two edge-disjoint paths. In contrast, it is well known that the k node-disjoint path problem in a wired network can be reduced to the k edge-disjoint path problem in another wired network for any given k, $1 \leq k \leq n-1$, while this reduction in wireless ad hoc networks seems not working at all. This can be seen that it is still open that whether finding minimum-energy k edge-disjoint paths is polynomially solvable with any $k \geq 3$, despite that finding minimum-energy k node-disjoint paths are polynomially solvable with any $k \geq 1$.

B. Motivations

In this paper we will focus on the design of algorithms for finding energy-efficient two node/edge-disjoint paths for each node/edge-disjoint path connection request. Our objective is to maximize the total number of node/edge-disjoint path connection requests successfully routed by the network without any knowledge on future request arrivals and generation rates. We refer to this problem as the *network capacity maximization problem* for on-line disjoint path connection. The motivation behind the problem is twofold. On one hand, we would like to route a message along two disjoint paths with the maximal minimum fraction of residual power after the message transmitted, which means that the message will be routed along the paths, in which the nodes have high residual power. This may be expensive because too much power consumption will decrease the overall residual power of the system and thus will decrease the network capacity. On the other hand, we would like to route the message along the two disjoint paths with minimizing the total transmission energy consumption in the paths. If the message is always routed along these two paths that have the minimum energy consumption, it may lead to quick battery depletion at critical nodes located in the paths, and the network is then partitioned. Therefore, for a given disjoint path connection request, it is generally difficult to find two node/edge-disjoint paths that meet these two opposite optimization objectives simultaneously. As a matter of fact, any algorithm for the problem is a trade-off between these objectives. Unlike the work by Srinivas and Modiano [19] that aims to minimize the total transmission energy consumption for a given disjoint path connection request, our objective is to maximize the network capacity through routing each message along energy-efficient two node/edge-disjoint paths for an unknown sequence of node/edge disjoint path connection requests. This means that the network capacity is maximized after the realization of a sequence of node/edge-disjoint path connection requests, despite that the total transmission energy consumption for a specific node/edge-disjoint path connection request may not be optimal.

C. Contributions

In this paper we study on-line disjoint path routing for maximizing network capacity by proposing two on-line algorithms. One is based on minimizing the transmission energy consumption of the two node/edge-disjoint paths for each node/edge disjoint path connection request, with the constraint that the network lifetime is guaranteed no less than γ times of the optimum, where γ is constant with $0 < \gamma \leq 1$. Another is based on the exponential function of energy utilization at nodes, which delivers a solution with a provably guaranteed competitive ratio. We also conduct extensive experiments by simulations to analyze the performance of the proposed algorithms against the existing algorithms in terms of network capacity. The experimental results show that the proposed algorithms outperform those existing algorithms and the on-line algorithm based on maximizing network lifetime with $\gamma = 0.8$ performs significantly better than all the other algorithms.

The rest of the paper is organized as follows. In Section II the wireless communication model and problem definitions are introduced. In Section III algorithms for finding two maximal minimum node/edge-disjoint paths in ad hoc networks are proposed. In Section IV, two heuristic algorithms for on-line disjoint path routing with maximizing network capacity are given. In Section V extensive experiments by simulations are conducted to compare the performance of the proposed algorithms against the existing algorithms. The conclusion is given in Section VI.

II. PRELIMINARIES

We consider a wireless network consisting of n nodes that have omnidirectional antennas and can vary their transmission power dynamically. Specifically, each node has a maximum transmission power level $\mathcal{E}_{\max}$, and we assume that transmission can take place at any power level in the range [0, $\mathcal{E}_{\max}$]. In other words, the wireless ad hoc network can be modeled by a directed graph $M = (N, A)$, where N is the set of nodes with $|N| = n$ and there is a directed edge $\langle u, v\rangle$ in A if node v is within the transmission range of u when u uses its maximum power level to broadcast a message. For a transmission from node u to node v, separated by a distance $d_{u,v}$, to guarantee that v can receive the message from u within one hop, the transmission power at node u is modeled to be proportional to $c' d^{\alpha}_{u,v}$, assuming that the proportionality constant c' is 1 for notational simplicity, where α is a parameter that typically takes on a value between 2 and 4 depending on the characteristics of the communication medium. The topology of this model is entirely dependent on the power at which nodes transmit. Links can be added or removed by a node changing its transmission range.

Given a wireless ad hoc network $M = (N, A)$, a source node s and a destination node t, the *minimum-energy two* *node/edge-disjoint path problem* is to find two node/edge-disjoint paths in M from s to t such that the sum of transmission energy at the nodes in the paths is minimized. The problem involves the choice of transmission nodes as well as the power level at every chosen transmission node. Given a wireless ad hoc network $M = (N, A)$, a connection request from a source node s to a destination node t, the *maximal minimum path problem* is to find a simple path in M from s to t such that the network lifetime is maximized after the realization of the request. Given a wireless ad hoc network $M = (N, A)$, a disjoint path connection request from a source node s to a destination node t, the *two maximal minimum node/edge-disjoint path problem* is to find two node/edge-disjoint paths in M from s to t such that the network lifetime is maximized after the realization of the request. Given a wireless ad hoc network $M = (N, A)$ and a sequence of disjoint path connection requests (s_i, t_i, τ_i) where s_i and t_i represent the source and destination nodes of the request, and τ_i is the length of the message, $1 \leq i \leq k$, the *network capacity maximization problem for on-line disjoint path connection* is to not only find two node/edge-disjoint paths for each request with minimizing the total transmission energy consumption but also maximize the number of node/edge-disjoint path connection messages successfully transferred by the network. We assume that the system has to make a decision whether or not a disjoint path connection request should be routed without any knowledge about future disjoint path connection request arrivals and generation rates when that disjoint path connection request arrives. Unless otherwise specified, in this paper *edges* and *links* are referred to *directed edges* and *directed links* respectively, and *edges* and *links* are used inter-exchangeably. To distinguish the connection request considered in this paper from the unicast connection request, we here use the terminology *disjoint path connection request*. A disjoint path connection request is either *a node-disjoint path connection request* or *an edge-disjoint path connection request*.

III. Algorithms for Finding Two Maximal Minimum Node/Edge-Disjoint Paths

In this section we present an algorithm for finding two maximal minimum node/edge-disjoint paths in an ad hoc network from s to t. In other words, we aim to achieve the maximum network lifetime after the realization of a node/edge-disjoint path connection request.

A. Find a maximal minimum path

Let RC_{opt} be the optimal solution of the maximal minimum path problem and $RE(v)$ the residual power capacity at node v when the current connection request arrives. Initially, $RE(v) = E(v)$ and $E(v)$ is the battery capacity at node v. The basic idea for finding a maximal minimum path is outlined as follows. We construct a directed graph $G' = (N, A', w)$ for a wireless ad hoc network M. There is a directed edge from u to v in A' if v is within the transmission range of u and the weight assigned to this edge is $w(u,v) = RE(u) - d^{\alpha}_{u,v}$, which is the amount of residual energy capacity at u if u uses this link to transmit one unit message to v within one hop. Let T be a directed tree in G' rooted at the source s that is constructed as follows.

Initially, T contains s only. Let V_s be the set of vertices in T including the source s and $V' = N - V_s$. $V_s = \{s\}$. The algorithm proceeds as follows. It picks up a directed edge $e = \langle u, v \rangle \in V_s \times V'$ in G' and includes it to the tree if the weight $w(e)$ of e is the maximum one among all the directed edges from $u \in V_s$ to $v \in V'$, and sets $V_s = V_s \cup \{v\}$ and $V' = V' - \{v\}$. This procedure continues until $t \in V_s$. Let P_{st} be the path in T from s to t. It can be seen that $RC_{opt} = \min_{e \in P_{st}}\{w(e)\}$, i.e., P_{st} is the maximal minimum path in G' from s to t. Clearly, the Dijkstra algorithm can be employed to find P_{st}, which takes $O(m + n \log n)$ time.

B. Find two maximal minimum disjoint paths

We will use the above algorithm as a subroutine in the design of algorithms for finding two maximal minimum node/edge-disjoint paths in an ad hoc network from s to t if they exist. For the sake of simplicity we deal with the two maximal minimum edge-disjoint path problem first. We then show how to extend the approach to solve the two maximal minimum node-disjoint path problem.

Let $G(V, E, w)$ be the current ad hoc network. The proposed algorithm first finds a maximal minimum path P in G from s to t. It then reverses all the edges of P in G and assigns them the identical weights as the original edges. Denote by G' the resulting graph. It finally finds a maximal minimum path P' in G' from s to t. As a result, a solution $P \oplus P'$ is obtained, where $\oplus$ is an exclusive union operator that removes such a pair of edges from the union of two edge sets that both of them are in the union. The correctness of the proposed algorithm is based on the following lemma.

Lemma 1: Let OPT be the maximum network lifetime for the edge-disjoint path connection request (s,t) and $w(e_0) = \min_{e \in P \oplus P'}\{w(e)\}$. Then, $OPT = w(e_0)$.

Proof: Let Q be a simple path in the ad hoc network and $W(Q)$ the weight of the minimum weighted edge in Q, i.e., $W(Q) = \min_{e \in Q}\{w(e)\}$. Following the algorithm, it is obvious that $OPT \leq W(P)$ because P is a maximal minimum path in G from s to t.

In what follows we focus on constructing the two maximal minimum edge-disjoint paths in G from s to t. (i) If P is one of the two paths, then another path from s to t in the optimal solution does not share any edges with P. This means that P' is a maximal minimum path in $G - P$ from s to t without using any edges in P. In other words, P' is the best possible path from s to t except P in terms of the network lifetime. P and P' form a solution to the problem and $OPT = \min_{e \in P \cup P'}\{w(e)\}$. (ii) To form two edge-disjoint paths from s to t and some of the edges in P are included by either or both the two paths, P' must be necklaced with P. Here we assume that they share l segments $L_1, L_2, \ldots, L_l$, which is shown in Fig. 1.

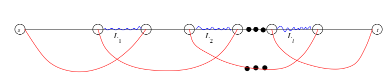


Fig. 1. The necklacing between P and P'

We here consider $l = 1$ only, which is illustrated in Fig. 2. We claim that $OPT \leq W(L_1)$ by contradiction. Assume that $OPT > W(L_1)$. We already know that L is a segment of P. Therefore, $W(P) \leq W(L)$. Following the assumption, we have $OPT > W(L_1) \geq W(P)$, which contradicts the fact that $OPT \leq W(P)$. Therefore, $OPT \leq W(L_1)$. Now,

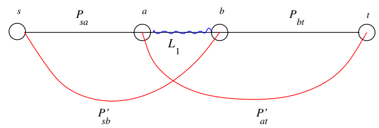


Fig. 2. One necklace between P and P'

if $OPT < W(L_1)$, then the removal of the edges in L_1 does not change the maximum network lifetime and $OPT = \min_{e \in P \oplus P'}\{w(e)\}$. Otherwise ($OPT = W(L_1)$), there must be a link $e' \in P - L_1$ such that $w(e') = OPT$, using the similar argument as above, i.e., $OPT = \min_{e \in P \oplus P'}\{w(e)\} = w(e')$. The solution consists of two edge-disjoint paths, segments P'_{sb} and P_{bt}, and segments P_{sa} and P'_{at}.

When $l \geq 1$, the claim is still true by induction on l, omitted. ■

For convenience we refer to the above algorithm as algorithm `max-min edge-disjoint`. We now show how to extend this approach to solve the two maximal minimum node-disjoint path problem. We adopt a well known node-splitting approach, which reduces the two node-disjoint path problem in the original network to the two edge-disjoint path problem in another auxiliary graph, and the solution for the latter gives a solution for the former. The auxiliary graph is constructed as follows.

For each node v in the original network, there are two corresponding nodes v' and v'' and there is a link from v' to v'' in the auxiliary graph. All incoming links to v now are the incoming links to v' and all outgoing links from v now are the outgoing links from v''. Fig. 3 illustrates the node splitting. It is clear that the outgoing degree of v' and the incoming degree of v'' in the auxiliary graph are ones. The weights assigned to all the other links in the auxiliary graph are identical to the weights in the original network. Each new link $\langle v', v''\rangle$ derived from node v is assigned a weight $w(v', v'') = RE(v)$.

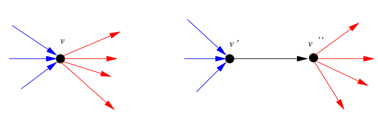


Fig. 3. The split of node v

Lemma 2: Given the above transformation, the maximum network lifetime after the realization of the two maximal minimum node-disjoint paths in the original network G from s to t is equal to the maximum network lifetime after realization of the two maximal minimum edge-disjoint paths in the auxiliary graph G' from s'' to t'.

Proof: We first show that the two maximal minimum edge-disjoint paths in G' from s'' to t' correspond to two node-disjoint paths in G from s to t. It is not difficult to see that the corresponding two paths in G are either edge-disjoint or node-disjoint. We claim that they must be node-disjoint by contradiction. Assume that they are edge-disjoint. Then, there is at least one node v'' in the two paths such that both $\langle v'', u'\rangle$ and $\langle v'', w'\rangle$ are the edges in the two paths, whereas the immediately precedent edge of these two edges is the edge $\langle v', v''\rangle$ because it is the only incoming edge of v''. This means that these two paths share one edge at least, which contradicts the fact that they are supposed to be edge-disjoint. We then show that the network lifetime obtained through the finding of these two paths is the maximum one. It is not hard to show that each edge in G' derived from the node splitting does not affect the value of the maximum network lifetime due to that the weight $RE(v)$ assigned to each edge $\langle v', v''\rangle$ is larger than the weight of any outgoing link of v''. If $\langle v', v''\rangle$ is included in one of the two paths, then, one of the outgoing edges of v'' must be included in one of the paths too. Thus, the maximum network lifetime is no greater than the weight of that outgoing edge. Therefore, the maximum network lifetime is less than $RE(v)$. ■

In summary we have the following theorem.

Theorem 1: Given an ad hoc wireless network and a pair of nodes s and t, there is an algorithm for finding two maximal minimum node/edge-disjoint paths in it from s to t, which takes $O(m + n \log n)$ time if the network contains m links.

IV. Algorithms for Network Capacity Maximization Problem

In this section we study the network capacity maximization problem for on-line disjoint path connection by proposing two algorithms for the problem. One is based on maximizing network lifetime, and another is based on an exponential function of energy utilization at nodes.

A. Algorithm based on maximizing network lifetime

1) Two node-disjoint paths: We first propose an on-line algorithm for the problem in the node-disjoint version. We then show how to extend the technique to solve the edge-disjoint version.

For each node-disjoint path connection request, the general strategy is to not only minimize the transmission energy consumption of the two node-disjoint paths but also guarantee that the network lifetime is no less than γ times of the optimum after realizing the request, where γ is constant with $0 < \gamma \leq 1$. The basic idea of the proposed algorithm is as follows. When a node-disjoint path connection request (s_i, t_i, τ_i) arrives, the algorithm is to find two maximal minimum node-disjoint paths in the network from s_i to t_i first. Having the two paths, the maximum network lifetime is obtained. A subnetwork is then induced from the original network and its lifetime is no less than γ times of the optimum after realizing the node-disjoint path connection request. The algorithm finally finds minimum-energy two node-disjoint paths in the subnetwork from s_i to t_i. The detailed algorithm follows.

Algorithm Bounded_Lifetime_Node-Disjoint_Paths(s_i, t_i, τ_i, γ)

1. Construct an auxiliary graph $G' = (N, A', w)$ for the current ad hoc network, where the weight $w(u, v)$

associated with a link $\langle u, v \rangle$ is $RE(u) - \tau_i d^{\alpha}_{u,v}$. Find the maximum network lifetime RC_{opt} by finding two maximal minimum node-disjoint paths in G' from s_i to t_i.
2. A subgraph graph $G_1 = (V, E_1, w)$ is induced from G' by removing each link $\langle u, v \rangle$ from G' if $w(u, v) < \gamma RC_{opt}$.
3. An auxiliary graph $G_2 = (V, E_2, \omega_2)$ that is identical to G_1 is constructed, where $E_2 = E_1$. The difference between G_1 and G_2 is that each link $\langle u, v \rangle \in E_2$ is assigned a new weight $\omega_2(u, v) = \tau_i d^{\alpha}_{u,v}$.
4. Find minimum-energy two node-disjoint paths in G_2 from s_i to t_i using the algorithm in [19].

For convenience, we refer to the above algorithm as algorithm `BLN node-disjoint`, and we then have the following theorem.

Theorem 2: Given an ad hoc network with n nodes and m links, there is a heuristic algorithm for on-line node-disjoint routing with an objective to maximize the network capacity. The algorithm takes $O(mn + n^2 \log n)$ time for each node-disjoint path connection request.

Proof: Following algorithm `BLN node-disjoint`, Step 1 takes $O(m)$ time to construct the auxiliary graph G' and $O(m+n \log n)$ time to find two maximal minimum node-disjoint paths in G'. Steps 2 and 3 take $O(m)$ time to construct graphs G_1 and G_2. Step 4 takes $O(mn + n^2 \log n)$ time. The theorem then follows. ■

2) Two edge-disjoint paths: We now show how to extend the above approach to solve the edge-disjoint version of the problem. Recall that there is an algorithm called Optimal Common Node Decomposition (`OCND` for short), for the minimum-energy edge-disjoint path problem [19]. The running time of algorithm `OCND` is $O(n^3 m + n^3 \log n)$, which can further be improved to $O(n^2 m+n^3 \log n)$ if the Suurballe and Tarjan algorithm [21] is applied. Therefore, given an ad hoc network, the minimum-energy two edge-disjoint path problem can be reduced to the minimum-energy two node-disjoint path problem.

We incorporate algorithm `OCND` into our solution to the on-line edge-disjoint routing problem for maximizing the network capacity as follows.

First, apply our improved algorithm for finding maximal minimum two edge-disjoint paths in the ad hoc network from s_i to t_i. As a result, the maximum network lifetime is obtained. Then, a subnetwork is induced from the original network by removing those edges $\langle u, v \rangle$ if $w(u, v) = RE_i(u) - \tau_i d^{\alpha}_{u,v} < \tau_i RC_{opt}$, where RC_{opt} is the maximum network lifetime. Finally, find minimum-energy two edge-disjoint paths in the subnetwork using the algorithm in [19]. We thus have the following theorem.

Theorem 3: Given an ad hoc network with n nodes and m links, there is a heuristic algorithm for on-line edge-disjoint path routing with an objective to maximize the network capacity. The algorithm takes $O(n^2 m+n^3 \log n)$ time for each edge-disjoint path connection request.

Proof: Following the proposed algorithm, finding the maximum network lifetime takes $O(m + n \log n)$ time. The construction of the subnetwork takes $O(m)$ time, and finding minimum-energy two edge-disjoint paths in the subnetwork takes $O(n^2 m + n^3 \log n)$ time. The theorem then follows. ■

B. Algorithm based on an exponential function of energy utilization at nodes

In this subsection we provide another heuristic algorithm for the concerned problem based on an exponential function of energy utilization at nodes. We assume that the disjoint path connection requests are indexed in the order they arrive. Let (s_i, t_i, τ_i) be the ith node/edge-disjoint path connection request. Denote by $RE_i(v)$ the residual energy capacity at node v at the time the ith node/edge-disjoint path connection request arrives (but before it is routed). Obviously, $RE_1(v) = E(v)$. Let $\alpha_i(v) = 1 - RE_i(v)/E(v)$ be the fraction of energy at node v that has been used when the node/edge-disjoint path connection request i arrives, which is referred to as the *energy utilization* at node v, $1 \leq i \leq k$. Let $e_{max} = \max_{\langle u,v \rangle \in A}\{d^{\alpha}_{u,v}\}$ and $e_{min} = \min_{\langle u,v \rangle \in A}\{d^{\alpha}_{u,v}\}$, which are respectively the maximum and minimum energies expended by a message of unit length when traversing some link in the current network. Let $\rho = e_{max}/e_{min}$. Define $\lambda = 2n\rho$, $\sigma_v = ne_{\max}$, and $\sigma_e = 3ne_{\max}$.

The basic idea of the on-line algorithm is to find two node/edge-disjoint paths with minimizing the transmission energy consumption for each node/edge-disjoint path connection request i, using an auxiliary weighted graph $G(V, E, \omega_1)$, where each link in E is assigned a weight that is exponential of the load of that link (total power consumption if the link is used for transmission of one unit message). Such weight assignment is very useful in the analysis of approximation algorithms as well as on-line algorithms [1], [9]. Specifically, $G(V, E, \omega_1)$ is defined as follows. If $RE_i(u) \geq \tau_i d^{\alpha}_{u,v}$, then $\langle u, v \rangle \in E$ and assign a weight $\omega_1(u, v) = \tau_i d^{\alpha}_{u,v}(\lambda^{\alpha_i(u)} - 1)$ to it. The proposed algorithm for a disjoint path connection request i is as follows.

Algorithm Two_Disjoint_Paths(s_i, t_i, τ_i, σ_v, σ_e)
1. An auxiliary graph $G = (V, E, \omega_1)$ is constructed, when a disjoint path connection request (s_i, t_i, τ_i) arrives.
2. Find minimum-energy two node/edge-disjoint paths in G from s to t using the corresponding algorithms in [19]. Note that the weight function on links now is ω_1.
3. Let PW be the sum of transmission energy of the two node/edge-disjoint paths. If $PW \leq \sigma_v$ (for node-disjoint case) or $PW \leq \sigma_e$ (for edge-disjoint case), then the request is realized; otherwise, reject the request.

For the sake of convenience, we refer to algorithm `Two_Disjoint_Paths` as `TDP`. The competitive ratio analysis of algorithm `TDP` is as follows. Let $L(k)$ be the total length of messages successfully routed by algorithm `TDP` till the arrival of the kth disjoint path connection request, and let $L_{opt}(k)$ be the total length of messages successfully routed by an optimal off-line algorithm till the arrival of the kth disjoint path connection request. We have the following theorem.

Theorem 4: In a wireless ad hoc network $M(N, A)$, for every node/edge-disjoint path connection request (s_i, t_i, τ_i), we assume that $\tau_i \leq \frac{\min_{v \in N}\{E(v)\}}{e_{max} \log \lambda}$, $1 \leq i \leq k$. Then, $\frac{L(k)}{L_{opt}(k)} \geq \log \lambda$. The running times of the algorithms are

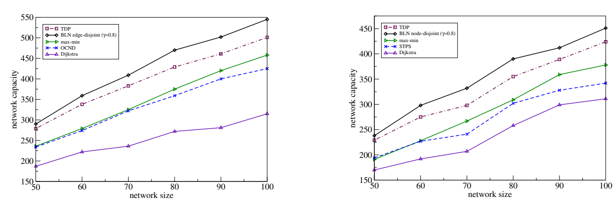
$O(nm + n^2 \log n)$ and $O(n^2 m + n^3 \log n)$ respectively for finding two node-disjoint paths and two edge-disjoint paths for each node/ edge disjoint path connection request.

Proof: The proof as well as the time complexity analysis is similar to the one given in [9], omitted. ■

V. Performance Evaluation

In this section we evaluate the performance of the proposed algorithms. We simulate networks that comprise 50, 60, 70, 80, 90 and 100 nodes respectively. We assume that the nodes in the network are uniformly distributed in a 1000 × 1000 square meters region. Path loss exponent α is set to 2, and the transmission power to maintain a directed link from nodes u to v with distance $d_{u,v}$ is $0.005 \times d_{u,v}^2$. The initial battery capacity $E(v)$ at each node $v \in N$ is set to be one of the values in $\{9000, 9500, 10000, 10500, 11000\}$ randomly. The maximum transmission range of each node is limited to 500 meters. In each experiment, 1000 disjoint path connection requests with a unit length message are injected into the system one by one.

We first compare the performance of the two proposed algorithms for edge-disjoint case with the other existing algorithms `OCND` and the algorithm based on Dijkstra's algorithm in [19]. From Fig. 4(a), we found that the performance of algorithm `BLN` for edge-disjoint case with $\gamma = 0.8$ is the best, while the network capacity delivered by algorithm `TDP` is below the one given by `BLN` but outperforms all the other algorithms. We then compare the performance of the two proposed algorithms for node-disjoint case against the other existing algorithms, the Source Transmit Power Selection `STPS` in [19] and an approach based on Dijkstra's algorithm. Through the observation from Fig. 4(b), we found that algorithm `BLN` for node-disjoint case delivers a solution with the largest network capacity when $\gamma = 0.8$. We finally analyze the impact of different γs on the performance of algorithm `BLN`. We vary the value of γ from 0.1 to 1.0 with an interval of 0.1, and observe its impact on algorithm `BLN` with network sizes 50 and 100 respectively, which is illustrated in Fig. 5. We observed that the network capacity delivered by algorithm `BLN` is maximized when $\gamma \approx 0.8$.



(a) (b)

Fig. 4. Network capacity under various edge-disjoint algorithms (a) and node-disjoint algorithms(b)

VI. Conclusions

In this paper we have considered the network capacity maximization problem for on-line disjoint path connection by proposing energy-efficient algorithms for it. We have also conducted extensive experiments by simulations, which show that the performance of the proposed algorithms outperform those of the existing ones.

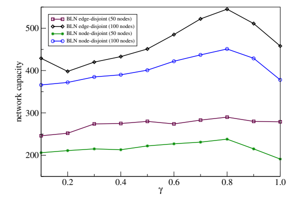


Fig. 5. Network capacity of various γs for algorithm `BLN node/edge-disjoint`

References

[1] J. Aspnes, Y. Azar, A. Fait, S. Plotkin, and O. Warrts. On-line routing of virtual circuits with applications to load balancing and machine scheduling. *J. of the ACM*, Vol. 44, pp. 486–504, 1997.

[2] M. Cagalj, J-P Hubaux, and C. Enz. Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues. *Proc. of MobiCom'02*, ACM, 2002.

[3] J. Cartigny, D. Simplot, and I. Stojmenovic. Localized minimum-energy broadcasting in ad hoc networks. *Proc. of INFOCOM'03*, IEEE, 2003.

[4] J-H Chang and L. Tassiulas. Energy Conserving routing in wireless ad hoc networks. *Proc. of INFOCOM'00*, IEEE, 2000.

[5] J-H Chang and L. Tassiulas. Fast approximate algorithms for maximum lifetime routing in wireless ad-hoc networks. *IFIP-TC6/European Commission Int'l Conf.*, Lecture Notes in Computer Science, Vol. 1815, pp.702-713, Springer, 2000.

[6] P. Gupta and P. R. Kumar. Critical power for asymptotic connectivity in wireless networks. *Analysis, Control, Optimization and Applications: A Volume in Honor of W. H. Fleming*, pp. 547–566, 1998.

[7] C.E. Jones, K. M. Sivalingam, P. Agrawal, and J.C. Chen. A survey of energy efficient network protocols for wireless networks. *Wireless Networks* Vol. 7, 1999, pp. 343–358, 2001.

[8] I. Kang and R. Poovendran. Maximizing static network lifetime of wireless broadcast adhoc networks. *Proc. of ICC'03*, IEEE, 2003.

[9] K. Kar, M. Kodialam, T. V. Lakshman, and L. Tassiulas. Routing for network capacity maximization in energy-constrained ad-hoc networks. *Proc. of INFOCOM'03*, IEEE, 2003.

[10] S.-J. Lee and M. Gerla. AODV-BR: Backup routing in ad hoc networks. *Proc. WCNC'00*, IEEE, 2000.

[11] Q. Li, J. Aslam, and D. Rus. Online power-aware routing in wireless ad hoc networks. *Proc. SIGMobile'01*, ACM, pp. 97–107, 2001.

[12] W. Liang. Constructing minimum-energy broadcast trees in wireless ad hoc networks. *Proc. of MOBIHOC'02*, ACM, pp.112–122, 2002.

[13] W. Liang and Y. Yang. Maximizing battery life routing in wireless ad hoc networks. *Proc of 37th Hawaii Int'l Conf on System Sciences*, IEEE, 2004.

[14] A. Nasipuri and S. R. Das. On-demand multipath routing for mobile ad hoc networks. *Proc. of ICCCN'99*, IEEE, pp.64–70, 1999.

[15] V.D. Park and M. S. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. *Proc. of INFOCOM'97*, IEEE, pp. 1405-1413, 1997.

[16] S. Plotkin. Competitive routing of virtual circuits in ATM networks. *IEEE J. SAC*, Vol.13, pp.1128–1136, 1995.

[17] J. Raju and J. J. Garcia-Luna-Aceves. A new approach to on-demand loop-free multipath routing. *Proc. of ICCCN'99*, IEEE, pp.522–527, 1999.

[18] A. Sankar and Z. Liu. Maximum lifetime routing in wireless ad-hoc networks. *Proc. of INFOCOM'04*, IEEE, 2004.

[19] A. Srinivas and E. Modiano. Minimum energy disjoint path routing in wireless ad-hoc networks. *Proc. of MOBICOM'03*, ACM, 2003.

[20] J. W. Suurballe. Disjoint paths in a network. *Networks*, Vol. 4, pp.125–145, 1974.

[21] J. W. Suurballe and R. E. Tarjan. A quick method for finding shortest pairs of disjoint paths. *Networks*, Vol.14, pp.325–336, 1984.

[22] P.-J Wan, G. Calinescu, and X-Y Li. Minimum-energy broadcast routing in static ad hoc wireless networks. *Proc. of INFOCOM'01*, IEEE, 2001.

[23] J.E. Wieselthier, G. D Nguyen and A. Ephremides. On construction of energy-efficient broadcast and multicast trees in wireless networks. *Proc. of INFOCOM'00*, IEEE, 2000.

[24] J.E. Wieselthier, G. D Nguyen and A. Ephremides. Resource management in energy-limited, bandwidth-limited, transceiver-limited wireless networks for session-based multicasting. *Computer Networks*, Vol. 39, pp. 113–131, 2002.

