Available online at www.sciencedirect.com
ScienceDirect
Ad Hoc Networks 5 (2007) 272–285
Ad Hoc Networks
www.elsevier.com/locate/adhoc

On-line disjoint path routing for network capacity maximization in energy-constrained ad hoc networks

Weifa Liang *, Yuzhen Liu

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

Received 24 August 2004; received in revised form 22 September 2005; accepted 21 November 2005
Available online 27 December 2005

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. Specifically, in this paper we first present two centralized on-line algorithms for the problem. One is based on maximizing local network lifetime, which aims to minimize the transmission energy consumption, under the constraint that the local network lifetime is no less than γ times of the optimum after the realization of each disjoint path connection request, where γ is constant with $0 < \gamma \leqslant 1$. Another is based on the exponential function of energy utilization at nodes, and the competitive ratio of this latter algorithm is also analyzed if admission control mechanism is employed. We then conduct extensive experiments by simulations to analyze the performance of the proposed algorithms, in terms of network capacity, network lifetime, and the transmission energy consumption for each disjoint path connection request. The experimental results show that the proposed algorithms outperform those existing algorithms that do not take into account the power load balancing at nodes in terms of maximizing the network capacity.

Keywords: Wireless communication network; On-line algorithm; Power awareness; Ad hoc networks; Energy consumption optimization; Disjoint path routing; Load-balanced routing; Network capacity

1. Introduction

In recent years, the multi-hop wireless ad hoc network (ad hoc network for short) has been receiving significant attention due to its potential applications from civil to military domains. An ad hoc network consists of a collection of mobile nodes equipped with energy-limited batteries. In such a network each mobile node serves as both a host to process information and a router to transmit and receive messages from other mobile nodes within its transmission range. The communication between two mobile nodes in the network can be either in a single hop transmission in which case both nodes are within the transmission ranges of each other or in multi-hop transmission where the message is relayed by intermediate mobile nodes. In general, nodes in ad hoc networks are mobile as well, though

* Corresponding author. Tel.: +61 2 6125 3019; fax: +61 2 6125 0010.
E-mail address: wliang@cs.anu.edu.au (W. Liang).

1570-8705/$ - see front matter
doi:10.1016/j.adhoc.2005.11.004

in this paper we primarily concern with relatively static ad hoc networks, a prevalent example of which is the sensor network.

The motivations for finding energy efficient multiple disjoint paths in ad hoc networks are as follows.

First, there is a great need for reliability in ad hoc networks, which stems from the unpredictable nature of the wireless environment. Unlike its wired counterpart, the ad hoc network 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 from a source node to a destination node along multiple disjoint paths simultaneously. Thus, the destination node is still able to receive the message even if there is one survivable routing path. This is especially true in real-time data transmission, whereby if a message routes along a single path, just one node or link failure suffices to cause path failure and transmission interruption.

Second, secure communication is an especially significant challenge in ad hoc networks. The distinctive feature of ad hoc networks is that mobile nodes need to collaborate with their peers in supporting the network functionality. In such an environment, malicious or selfish nodes can disrupt or even deny the communications of potentially any other node in the network, because every node in the network is not only entitled, but also in fact required, to assist the network establishment, the network maintenance, and the network operation. The challenge in addressing these security vulnerabilities is due to the particular ad hoc networks characteristics and due to the fact that traditional security mechanisms may not be applicable. Therefore, to secure the message transmission in ad hoc networks, one possible solution is to partition the message into a certain number of segments and route the segments along multiple disjoint paths to the destination, the message is then restored at the destination by reassembling these segments, one such a protocol called *secure message transmission protocol* was proposed recently [18].

Finally, mobile nodes, especially sensors, tend to use small batteries for energy supply that are in many instances non-replenishable. It is well known that wireless communication consumes the significant amount of battery energy [9]. The limited battery lifetime imposes a strict constraint on the network performance, and energy conservation in such networks is a vital factor to prolong the network lifetime. Therefore, one of the optimization objectives of multiple disjoint path routing is to minimize the total transmission energy consumption at nodes in the multiple disjoint paths. One popular strategy for finding k ($k \geqslant 1$) disjoint paths is to find the first k shortest paths in the network in terms of transmission energy consumption. However, in practice this strategy may not work at all because (i) it may miss some routing paths even if they exist; and (ii) the critical nodes that are located in the shortest paths will expend their power very quickly, which will lead to the network partitioned easily even if the other nodes may have plenty of residual energy.

1.1. Related work

In ad hoc networks, energy efficiency is paramount of importance. Energy-aware routing optimization has been addressed in recent years. Most of them focus on broadcast/multicast routing with minimizing the total energy consumption [2–4,7,14,26,27]. However, in many practical applications, the general interest to a given routing problem is not only to minimize the total transmission energy consumption but also to maximize the lifetime at nodes due to the fact that 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 node extinction, 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 referred as to the time of the first node failure [6]. There are several studies that aim to prolong the network lifetime [6,5,10]. For example, Chang and Tassiulas [6,5] proposed maximizing network lifetime for on-line unicast through avoiding using low power nodes and choosing a shortest path in terms of energy consumption if the packet rate in the network is given in advance. Kang and Poovendran [10] considered maximizing network lifetime for the broadcast tree problem. Gupta and Kumar [8] 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 is insufficient, in terms of saving energy and prolonging the network lifetime. The better way is to take both of them into consideration. There are a number of studies for broadcast requests that tradeoff the two optimization objectives very well [28,10]. The work closely related to this study is given by Li et al. [13] who considered on-line unicasting, which aims to maximize the network lifetime while keeping the total energy consumption for each connection request is bounded. Liang and Yang [15] provided an improved algorithm for the problem which improves the running time in [13]. Sankar and Liu [22] recently provided an analytical solution for the problem, using the multicommunity flow approach. Kar et al. [11] re-examined on-line unicasting 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 provably guaranteed competitive ratio, using the ideas developed for wired networks [20].

Although several multiple disjoint path routing protocols for ad hoc networks have been proposed [12,16,19,21], none of these protocols incorporate energy efficiency into consideration in the design of routing protocols. Srinivas and Modiano [23] first considered 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 any $k \geqslant 2$ 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, $1 \leqslant k \leqslant n-1$, while this reduction in wireless ad hoc networks seems not working at all. It can be seen whether finding minimum-energy k edge-disjoint paths is polynomially solvable with any $k \geqslant 3$ is still open, despite that finding minimum-energy k node-disjoint paths is polynomially solvable with any $k \geqslant 1$. In this paper we will focus on the design of on-line algorithms for a sequence of disjoint path connection requests in energy-constrained ad hoc networks, assuming the request sequence is unknown in advance. Our objective is to maximize the total number of disjoint path connection messages successfully routed by the network without any knowledge of future message arrivals and generation rates. We refer to this problem as the *network capacity maximization problem* for on-line disjoint path connection, which is equivalent to minimizing the number of disjoint path connection messages that cannot be routed for the unknown sequence of disjoint path connection requests.

The motivation behind the concerned problem is twofold. For a given disjoint path connection request, 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 consisting of high residual power nodes. The cost associated with this approach may be expensive because too much power consumption will decrease the overall power level of the system and thus will shorten the network lifetime. On the other hand, we would like to route the message along two disjoint paths with minimizing the total transmission energy consumption. If the message is always routed along two paths with the minimum energy consumption, it may lead to that some critical nodes located in the routing paths will expend their power very soon, and the network is then partitioned. Therefore, it is generally difficult to find two node/edge-disjoint paths for the request that meet these two opposite optimization objectives simultaneously. As a matter of fact, any algorithm for the problem is a trade-off between the two optimization objectives. Unlike the work given by Srinivas and Modiano [23] 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 power load-balanced, 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.

1.2. Contributions

In this paper we study on-line disjoint path routing for network capacity maximization by proposing two centralized 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 local network lifetime is no less than γ times of the optimum, where γ is constant with $0 < \gamma \leqslant 1$. 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, network lifetime, and the transmission energy consumption per disjoint path connection request. The experimental results demonstrate that (i) the proposed algorithms outperform those existing algorithms that do not take into account the power load balancing at nodes, in terms of prolonging the network capacity for the concerned problem. (ii) The performance of the on-line algorithm based on maximizing local network lifetime with $\gamma \approx 0.8$ outperforms all the other algorithms.

The rest of the paper is organized as follows. In Section 2 the wireless communication model and problem definitions are introduced. In Section 3 algorithms for finding two maximal minimum node/edge-disjoint paths in ad hoc networks are proposed. In Section 4 two on-line algorithms for the problem with network capacity maximization are given. In Section 5 extensive experiments by simulations are conducted in order to compare the performance of the proposed algorithms against the existing algorithms. The conclusion is given in Section 6.

2. Preliminaries

2.1. Wireless communication model

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 $\mathscr{E}_{\max}$ and the transmission can take place at any power level in the range $[0, \mathscr{E}_{\max}]$. 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 is within the transmission range of u, the transmission power at node u is modeled to be proportional to $d^{\alpha}_{u,v}$, assuming that the proportionality constant 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. Denote by $e_{u,v}$ the amount of energy consumed for transmitting a unit message from u to v along a link $\langle u, v\rangle \in A$ within one hop, i.e. $e_{u,v} = d^{\alpha}_{u,v}$.

The topology of this model is entirely dependent on the ranges at which nodes transmit. Links can be added to or removed from the network by a node changing its transmission range. Therefore, there is a trade-off between reaching more nodes in a single hop by using higher power versus reaching fewer nodes in a single hop by using lower power.

In this paper we assume that the network is at least stable during the execution of the proposed algorithm to respond to each disjoint-path connection request.

2.2. On-line algorithms vs off-line algorithms

Suppose that our optimization objective (network capacity) is a metric. In this paper we assume the disjoint path connection requests arrive one by one, and there is not any knowledge of future disjoint path connection request arrivals and generation rates. If the sequence of the disjoint path connection requests is known ahead of time, it is possible to develop an off-line algorithm for it to maximize the metric. Thus, the off-line algorithm is usually used as a benchmark to measure the performance of an on-line algorithm. The *competitive ratio* of an on-line algorithm is the worst case ratio of the metric of the on-line algorithm to the metric of the off-line algorithm over all instances. Ideally, such a competitive ratio is expected to be a small constant, which however is impossible in general, as shown in [1,20]. In fact, if there is no control on the message lengths of the disjoint path connection requests, then, the competitive ratio can be as bad as O(n). Intuitively, the on-line algorithm can be made to perform arbitrarily bad by an *adversary* who injects disjoint path connection requests into the system if (i) the message length can be arbitrarily long; and (ii) the algorithm is not allowed to take admission control. Therefore, to obtain the competitive ratio of an on-line algorithm, we assume that the algorithm can perform admission control, i.e. it is allowed to reject some "expensive" disjoint path connection requests even if there are sufficient resources in the network to enable the requests to be realized.

2.3. Problem definitions

In the following we provide notations and problem definitions, which will be used in this paper.

A (disjoint path) connection request: consists of a source node s, a destination node t and the length τ of the message that needs to be transmitted from s to t, which is referred to (s, t, τ).

Local network lifetime: Given a wireless ad hoc network $M = (N, A)$, a (disjoint path) connection request (s, t, τ), the *local network lifetime* is defined as the minimum residual energy among the nodes in the network if the request is realized. Clearly the lifetime of a node is proportional to the residual power of its battery. In contrast to the local network lifetime, the (global) network lifetime is defined as the time the first node dies after the network has realized a sequence of (disjoint path) connection requests, assuming that the request sequence is unknown in advance and the request arrives one by one randomly.

The maximal minimum path problem: Given a wireless ad hoc network $M = (N, A)$, a connection request (s, t, τ), the *maximal minimum path problem* is to find a simple path in M from s to t such that the local network lifetime is maximized after the realization of the request.

The minimum-energy two node/edge-disjoint path problem: Given a wireless ad hoc network $M = (N, A)$, a disjoint path connection request (s, t, τ), 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 transmitter-power level at every chosen transmission node.

The two maximal minimum node/edge-disjoint path problem: Given a wireless ad hoc network $M = (N, A)$, a disjoint path connection request (s, t, τ), the problem is to find two node/edge-disjoint paths in M from s to t such that the local network lifetime is maximized after the realization of the request.

The network capacity maximization problem for on-line disjoint path connection: 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 transmitted message, $1 \leqslant i \leqslant k$, the problem is to 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 the request should be realized when a disjoint path connection request arrives. We further assume that there is not any knowledge of future disjoint path connection request arrivals and generation rates.

Unless otherwise specified, in this paper *edges* and *links* are referred to *directed edges* and *directed links* respectively, and *edges* and *links* are used interchangeably. 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*.

3. Algorithms for finding two maximal minimum node/edge-disjoint paths

In this section we assume that the length of message transmitted is τ, we present algorithms for finding two maximal minimum node/edge-disjoint paths in an ad hoc network from s to t only. In other words, we aim to achieve the maximum local network lifetime after the realization of a node/edge-disjoint path connection request (s, t, τ).

3.1. Find a maximal minimum path

Let RC_{opt} be the value of the optimal solution of the maximal minimum path problem and $RE(v)$ the residual power capacity at node v when a new connection request arrives. Initially, $RE(v) = E(v)$ where $E(v)$ 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 the current 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 $RE(u) - \tau d^{\alpha}_{u,v} \geqslant 0$, and the weight assigned to the edge is $w(u, v) = RE(u) - \tau d^{\alpha}_{u,v}$, which is the amount of residual power capacity at u if u uses this link to transmit a message of length τ to v directly. Let T be a directed tree in G' rooted at the source s which is constructed as follows.

Let V_s be the set of vertices in T and $V' = N - V_s$. Initially, T contains s only and $V_s = \{s\}$. 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 T if the weight $w(e)$ of e is the maximum one among all the directed edges from $u \in V_s$ to $v \in V'$.

It then sets $V_s = V_s \cup \{v\}$ and $V' = V' - \{v\}$. This procedure continues until $t \in V_s$.

Let $P_{\rm st}$ be the unique path in T from s to t. It is easy to show that $RC_{\rm opt} = \min_{e \in P_{\rm st}}\{w(e)\}$, i.e. $P_{\rm st}$ is the maximal minimum path in G' from s to t. Clearly, the Dijkstra algorithm can be employed to find $P_{\rm st}$, which takes $O(m + n\log n)$ time, where n and m are the number of nodes and links in the network respectively.

This algorithm will be used as a subroutine later in the design of algorithms for finding two maximal minimum node/edge-disjoint paths in ad hoc networks from s to t if the paths exist.

3.2. A simple algorithm

The algorithm for finding two maximal minimum node/edge-disjoint paths in an ad hoc network proceeds as follows.

First, the source node collects the energy information of links in the network and sorts these link information in increasing order, where the weight assigned to link $\langle u,v\rangle$ is $w(u,v) = RE(u) - \tau d^{\alpha}_{u,v}$. Let $w(l_1), w(l_2), \ldots, w(l_{m_p})$ be the sorted energy sequence of the links, assuming that there are m_p links in G' and l_i is a link with $1 \leqslant i \leqslant m_p$. Then, it finds a maximal index i on the sorted energy sequence using binary search such that there are two node/edge-disjoint paths from s to t in the subgraph $G(V, E_i)$, induced by the links in G' that the weight of each of them is no less than $w(l_i)$. Such a subgraph must exist, because G' itself is one of the subgraphs. It can be seen that $w(l_i)$ is the value of the maximum local network lifetime for the two maximal minimum node/edge path problem, $1 \leqslant i \leqslant m_p$.

The running time of the above simple algorithm thus is $O(m\log n + n\log^2 n)$, which is analyzed as follows.

Sorting takes $O(m_p \log m_p) = O(m\log n)$ time since $m_p \leqslant m$ and $m = O(n^2)$, and determining whether a graph contains two node/edge-disjoint paths from s to t takes $O(m_p + n\log n) = O(m + n\log n)$ time using Suurballe's algorithm [24], where m is the maximum number of links in the network. There are $\lceil \log m_p \rceil \leqslant \lceil \log n^2 \rceil \leqslant 2\lceil \log n \rceil$ rounds of iterations for finding the maximal index i on the sorted energy sequence, using binary search. Once the maximum local network lifetime is found, a subgraph of $G(V, E_i)$ is then obtained, and two minimum-energy node/edge-disjoint paths in it can be found later, using the algorithms due to Srinivas and Modiano [23]. To collect the energy information of each link in the network, it is appropriate to build a broadcast tree rooted at the source s for such a propose. However, this simple algorithm needs lots of coordinations among the nodes and consumes much energy to find the maximum local network lifetime. In the following we present an improved algorithm for the problem to overcome the drawbacks brought by this simple algorithm.

3.3. An improved algorithm

In this subsection we provide an improved algorithm for the maximal minimum node/edge-disjoint path problem, which takes much less running time. For the sake of simplicity, we deal with the two maximal minimum edge-disjoint path problem first. We then show how to extend the proposed approach to solve the two maximal minimum node-disjoint path problem. Our method is described as follows.

Let $G(V, E, w)$ be the current ad hoc network. It 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 the reversed edges with the identical weights as the original ones. 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 each pair of edges $\langle u,v\rangle$ and $\langle v,u\rangle$ from the union of two edge sets if both of them are there. The correctness of the proposed algorithm is based on the following lemma.

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

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 proposed algorithm, it is obvious that $\mathrm{OPT} \leqslant W(P)$ 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 by considering two cases: (i) P is one of the two paths; and (ii) the edges in P are included by either one of the two paths.

(i) If P is one of the two paths, then another path from s to t in the optimal solution does not

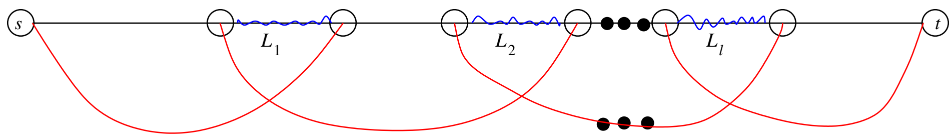

Fig. 1. The interlacing between P and P'.

include any edges in P. This means that P' is a maximal minimum path in $G - P$ from s to t without use of any edges in P. In other words, P' is the best possible path from s to t except P in terms of maximizing the local network lifetime. P and P' form a solution of the problem and $\mathrm{OPT} = \min_{e \in P \cup P'}\{w(e)\}$.

(ii) To form two edge-disjoint paths from s to t and some edges in P are included by either one of the two paths, P' must interlace with P. Here we assume that they share l segments $L_1, L_2, \ldots, L_l$, which is shown in Fig. 1.

We here consider $l = 1$ only, which is illustrated in Fig. 2.

We show $\mathrm{OPT} \leqslant W(L_1)$ by contradiction. Assume that $\mathrm{OPT} > W(L_1)$. We already know that L_1 is a segment of P. Thus, $W(P) \leqslant W(L_1)$. Following the assumption, we have $\mathrm{OPT} > W(L_1) \geqslant W(P)$. This contradicts the fact that $\mathrm{OPT} \leqslant W(P)$. Therefore, $\mathrm{OPT} \leqslant W(L_1)$.

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

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

For convenience, we refer to the above algorithm as algorithm `max–min`. We now show how to extend this approach to solve the two maximal minimum node-disjoint path problem. Here we adopt a well known node-splitting approach to reduce 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' in the auxiliary graph is one and the incoming degree of v'' is one too. The weights assigned to all the other links in the auxiliary graph are identical to the ones given in the original network. Each new link $\langle v', v''\rangle$ derived from node v is assigned a weight $w(v', v'') = RE(v)$.

We claim that the maximum local network lifetime of the two maximal minimum node-disjoint paths in the original network from s to t is equal to the maximum local network lifetime of the two maximal minimum edge-disjoint paths in the auxiliary graph from s'' to t' by the following lemma.

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

Proof. We first show that two maximal minimum edge-disjoint paths in G' from s'' to t' correspond

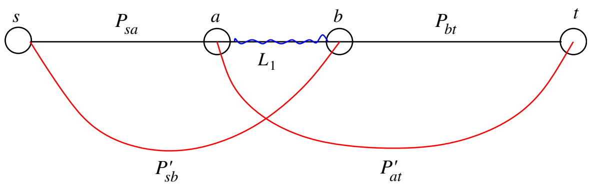

Fig. 2. One interlacing between P and P'.

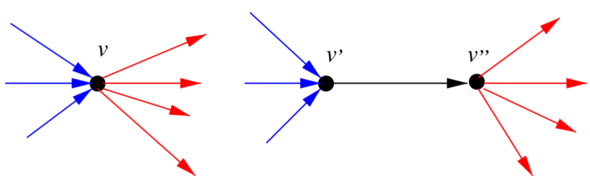

Fig. 3. The splitting of node v.

to two node-disjoint paths in G from s to t. It is not difficult to see that the two corresponding paths in G are either edge-disjoint or node-disjoint. The rest is to show that they must be node-disjoint by contradiction. Assume that they are edge-disjoint but not node-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 respectively, whereas the immediately precedent edge of these two edges in the paths 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 edge-disjoint. We then show that the local 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 local 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 by one of the two paths, then, one of the outgoing edges of v'' must be included by the path as well. Thus, the maximum local network lifetime is no greater than the weight of any outgoing edge of v''. Therefore, the maximum local network lifetime is less than $RE(v)$. □

In summary we have the following theorem.

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

Obviously, the proposed algorithm improves the running time of the simple algorithm by an $O(\log n)$ factor and is more efficient in terms of energy consumption. Moreover, it can be implemented distributively, using any distributed algorithm for the single source shortest path problem.

4. Algorithms for network capacity maximization problem

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

4.1. Algorithm based on maximizing local network lifetime

4.1.1. 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 proposed approach to solve the problem in the edge-disjoint version.

For each node-disjoint path connection request, the general strategy is to minimize the transmission energy consumption of the two node-disjoint paths under the constraint that the local network lifetime is no less than γ times of the optimum after realizing the request, where γ is a constant with $0 < \gamma \leqslant 1$.

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 local network lifetime is obtained. A subnetwork is then induced from the original network such that the local network lifetime of the subnetwork as well as the original network is no less than γ times of the optimum after realizing the node-disjoint path connection request on it. The algorithm finally finds minimum-energy two node-disjoint paths in the subnetwork from s_i to t_i. Denote by $RE_i(v)$ the residual energy capacity at node v when node-disjoint path connection request i arrives (but before it is routed). The detailed algorithm is given below.

Algorithm. Bounded_Lifetime_Node-Disjoint_Paths $(s_i, t_i, \tau_i, \gamma)$

begin

1. Construct an auxiliary directed graph $G' = (N, A', w)$ for the current ad hoc network, where there is a link $\langle u, v \rangle$ in A' if $RE_i(u) - \tau_i d^{\alpha}_{u,v} \geqslant 0$, and the weight $w(u, v)$ associated with the link is $RE_i(u) - \tau_i d^{\alpha}_{u,v}$. Find the maximum local network lifetime RC_{opt} through 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 those links $\langle u, v \rangle$ from it if $w(u, v) < \gamma\ RC_{\text{opt}}$.
3. An auxiliary graph $G_2 = (V, E_1, \omega_2)$ is constructed, and each link $\langle u, v \rangle \in E_1$ 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 [23].

end.

For convenience, we refer to the above algorithm as algorithm `BLN node-disjoint`. We 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 path 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' from s to t. 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 to find minimum-energy two node-disjoint paths. The theorem then follows. □

4.1.2. Two edge-disjoint paths

We now show how to extend the above approach to solve the problem in the edge-disjoint version. Recall that there is an algorithm for the minimum-energy edge-disjoint path problem in ad hoc networks called Optimal Common Node Decomposition (`OCND` for short) [23]. For the sake of completeness, we reproduce algorithm `OCND` as follows.

It first computes minimum-energy two node-disjoint paths for each pair of nodes in the network if such paths exist. It then constructs an auxiliary graph that has the same set of nodes as the original network. There is an edge in the auxiliary graph from u to v if there are two node-disjoint paths from u to v in the original network, and the weight assigned to the edge is the sum of transmission energy of the two paths. It finally finds a shortest path in the auxiliary graph from s to t, which corresponds to minimum-energy two-edge disjoint paths in the original network from s to t.

The running time of algorithm `OCND` is $O(n^3 m + n^4 \log n)$, which is analyzed as follows. Step 1 takes $O(n^3 m + n^4 \log n)$ time due to the fact that it takes $O(n(m + n \log n))$ time to find minimum-energy two node-disjoint paths for a pair of nodes, while each node may have up to $n - 1$ neighboring nodes. Step 2 takes $O(n^2)$ time. Step 3 takes $O(n^2 + n \log n)$ time. It is easy to see that the dominant running time of algorithm `OCND` is at Step 1. The running time of algorithm `OCND` can be further improved to $O(n^2 m + n^3 \log n)$ if the Suurballe and Tarjan algorithm [25] is applied. Thus, the minimum-energy two edge-disjoint path problem can be reduced to the minimum-energy two node-disjoint path problem eventually.

We now solve on-line edge-disjoint path routing for network capacity maximization by incorporating algorithm `OCND` into our algorithm as follows.

First, apply our improved algorithm for finding maximal minimum two edge-disjoint paths in the current ad hoc network from s_i to t_i. As a result, the maximum local network lifetime is obtained. Then, a subnetwork is induced by removing those edges $\langle u, v \rangle$ from the original network if $w(u, v) = RE_i(u) - \tau_i d^{\alpha}_{u,v} < \gamma RC_{\text{opt}}$, where RC_{opt} is the value of the maximum local network lifetime for the request of length τ_i. Finally, find minimum-energy two edge-disjoint paths in the subnetwork using algorithm `OCND`. 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 local 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. □

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

In this subsection we provide another heuristic algorithm for the problem of concern 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. Recall that $E(v)$ is the initial energy capacity and $RE_i(v)$ is the residual energy capacity at node v when the node/edge-disjoint path connection request i arrives. Obviously, $RE_1(v) = E(v)$. Let $\beta_i(v) = 1 - RE_i(v)/E(v)$ be the fraction of energy at node v that has been used when node/edge-disjoint path connection request i arrives, which is referred to *energy utilization* at node v, $1 \leqslant i \leqslant k$. Let $e_{\max} = \max_{\langle u,v\rangle \in A}\{e_{u,v}\}$ and $e_{\min} = \min_{\langle u,v\rangle \in A}\{e_{u,v}\}$, which are respectively the maximum and minimum energy expended by transferring a message of unit length along the links in the network. Let $\rho = e_{\max}/e_{\min}$. Define $\lambda = 2n\rho$, $\sigma_v = ne_{\max}$, and $\sigma_e = 3ne_{\max}$, *where* σ_v and σ_e are the upper bounds of routing costs for two node-disjoint and two edge-disjoint paths, respectively.

The basic idea of the on-line algorithm is to find two node/edge-disjoint paths for the request such that the transmission energy consumption of the two paths is minimized, and the minimum residual energy among the nodes is maximized, for each node/edge-disjoint path connection request i, $1 \leqslant i \leqslant k$. To do so, it is generally difficult due to the involvement of the two optimization objectives. It is shown that a heuristic, based on an exponential function of the resource utilization for other routing problems with two optimization objectives, is very useful [1,11]. Here we use a heuristic based on an exponential function of energy utilization at each node to provide a solution for the problem. To incorporate the residual energy at each node into the optimization objectives, we define the weight function $\omega_1 : E \mapsto \mathscr{R}$ for links in the energy directed graph G. If $RE_i(u) \geqslant \tau_i d^{\alpha}_{u,v}$, then $\langle u,v\rangle \in E$ and is assigned a weight $\omega_1(u,v) = \tau_i d^{\alpha}_{u,v}(\lambda^{\beta_i(u)} - 1)$. The proposed algorithm for a disjoint path connection request i is as follows.

Algorithm. Two_Disjoint_Paths($s_i, t_i, \tau_i, \sigma_v, \sigma_e$)

begin
1. An auxiliary graph $G = (V, E, \omega_1)$ is constructed, when a node/edge-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 due to Srinivas and Modiano [23]. 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 $PW \leqslant \sigma_v$ (for node-disjoint case) or $PW \leqslant \sigma_e$ (for edge-disjoint case), then the request is realized; otherwise, the request is rejected.

end.

For the sake of convenience, we refer to algorithm `Two_Disjoint_Paths` as `TDP`. In algorithm `TDP`, Step 1 is to construct an auxiliary graph and assign a weight to each directed edge. The weight on a link $\langle u,v\rangle$ in G is determined by two factors: one is the energy consumption on it if the link is in one of the two routing paths, and another is the energy utilization $\beta_i(u)$ at node u. The weight will increase when either $\beta_i(u)$ increases or the distance between u and v becomes larger. This means that the algorithm tries to avoid the links that require high energy for transmission and the nodes at which the residual energy fractions are low. Step 2 is to find minimum-energy two node/edge-disjoint paths in G from s_i to t_i for request i. Step 3 employs an admission control mechanism, a reject will be rejected if the cost of realizing the request is too high (above a given threshold σ_v or σ_e). Without this option to reject, an adversary can inject messages that consume too much resource destroying the competitive ratio of the algorithm. However, as stated in [11], this is not of practical consequence when messages are generated at random or by an adversary who do not know the routing policy. In practice, rejecting a message when sufficient energy is available is usually unacceptable. The later experimental results indicate that although setting σ_v or σ_e to its theoretically determined value might improve the capacity performance, setting σ_v or σ_e to infinity results in the promising performance with respect to both the network capacity and the network lifetime.

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 $L_{\rm opt}(k)$ 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 an ad hoc network $M(N, A)$, for every node/edge-disjoint path connection request (s_i, t_i, τ_i), we assume that $\tau_i \leqslant \frac{\min_{v\in N}\{E(v)\}}{e_{\max}\log\lambda}$, $1 \leqslant i \leqslant k$. Then, $\frac{L(k)}{L_{\rm opt}(k)} \geqslant \log\lambda$. In other words, the on-line disjoint path routing algorithm for maximizing network capacity delivers a solution with the competitive ratio of $O(\log n \log \frac{ne_{\max}}{e_{\min}})$. The running time of the algorithm for finding two node-disjoint/edge-disjoint paths for each node/edge disjoint path connection request takes $O(nm + n^2\log n)$ or $O(n^2 m + n^3 \log n)$ time respectively.*

Proof. The proof as well as the time complexity analysis is similar to the one given in [11], omitted. □

4.3. Distributed implementation

In this subsection we deal with the distributed implementation issue of the proposed centralized algorithms. It is more appropriate to provide distributed algorithms instead of centralized algorithms for an ad hoc network due to its nodes mobility and constantly changing topologies. In the following we show how to transform the proposed centralized algorithms into efficient, distributed algorithms.

Before we proceed, we assume that the network is stationary during the execution of the proposed algorithm. In other words, we assume that the nodes in the network are static and the communication channel is stable and no link failure is allowed during the execution of the proposed algorithm to respond to each disjoint path connection request. We further assume that each node in the network has only the knowledge of its neighboring nodes (which nodes being one of its neighbors and the distance between itself and the neighbor).

It is not difficult to see that the proposed algorithms in this paper can be treated as the single source shortest path algorithm eventually, while there are plenty of distributed algorithms for the single-source shortest path problem available. For example, there is such an efficient distributed algorithm for finding two node/edge disjoint paths with minimizing the weighted sum of the edges in [17]. Thus, the proposed algorithms based on either maximizing local network lifetime or the exponential function of energy utilization at nodes can be implemented distributively.

5. Performance evaluation

In this section we evaluate performance of the online algorithms of disjoint path routing for network capacity maximization in energy-constrained ad hoc networks, in terms of network capacity, network lifetime, and average energy consumption per disjoint path connection request. We simulate networks that comprise 50, 60, 70, 80, 90 and 100 nodes

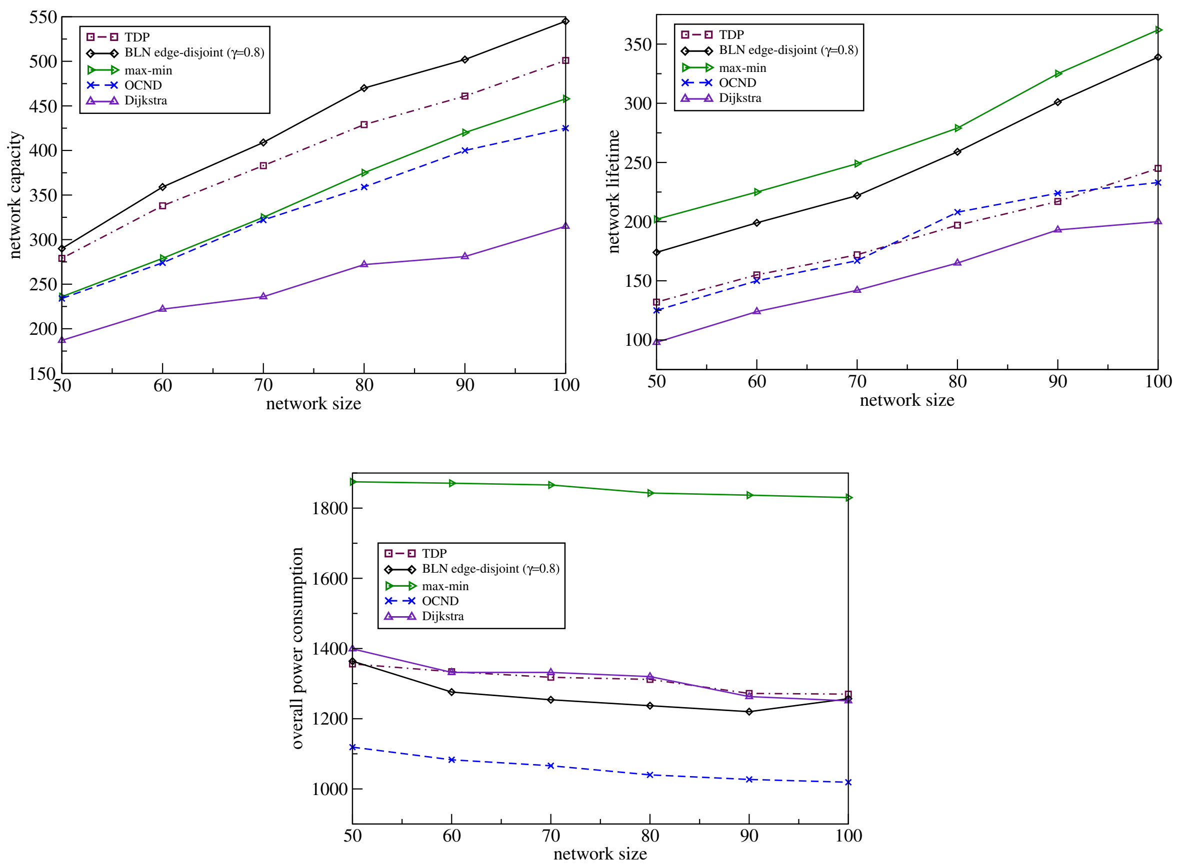

Fig. 4. Network capacity, network lifetime, and the energy consumption per request, under various edge-disjoint algorithms.

respectively and the nodes are uniformly distributed in a 1000×1000 square meters region. Path loss exponent α is set to be 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.

In our experiments, following the similar setting as given in [11], 1000 disjoint path connection requests with a unit length message are injected into the system one by one. The source-destination pair of each request is also randomly chosen. To measure the network capacity, we did not terminate the simulation until all the 1000 requests had been processed, i.e. each request is either realized or rejected. As for the implementation of algorithm TDP, an appropriate value of λ is chosen for a different network size, σ is set to be ∞, which represents an absence of admission control. Therefore, requests are only rejected due to insufficient resources (energy) for realizing them.

We first compare the performance of the two proposed algorithms for the problem in the edge-disjoint version with the other existing algorithms, algorithm OCND and the algorithm based on Dijkstra's algorithm in [23] through experimental simulations. Fig. 4 illustrates the network capacity, the network lifetime, and the average energy consumption per request by these algorithms. Here the network lifetime is the time when the first node failure due to its power expenditure, the network capacity is the total number of successfully delivered messages out of the 1000 random requests, and the energy consumption is the average of the total energy consumption of the two edge-disjoint paths per realized request. From this figure, we found that the solution delivered by algorithm max–min has the longest network lifetime in all cases at the expense of higher energy consumption per request. Algorithm BLN edge-disjoint with $\gamma \approx 0.8$ is the best in terms of maximizing network capacity, while its overall energy consumption per request

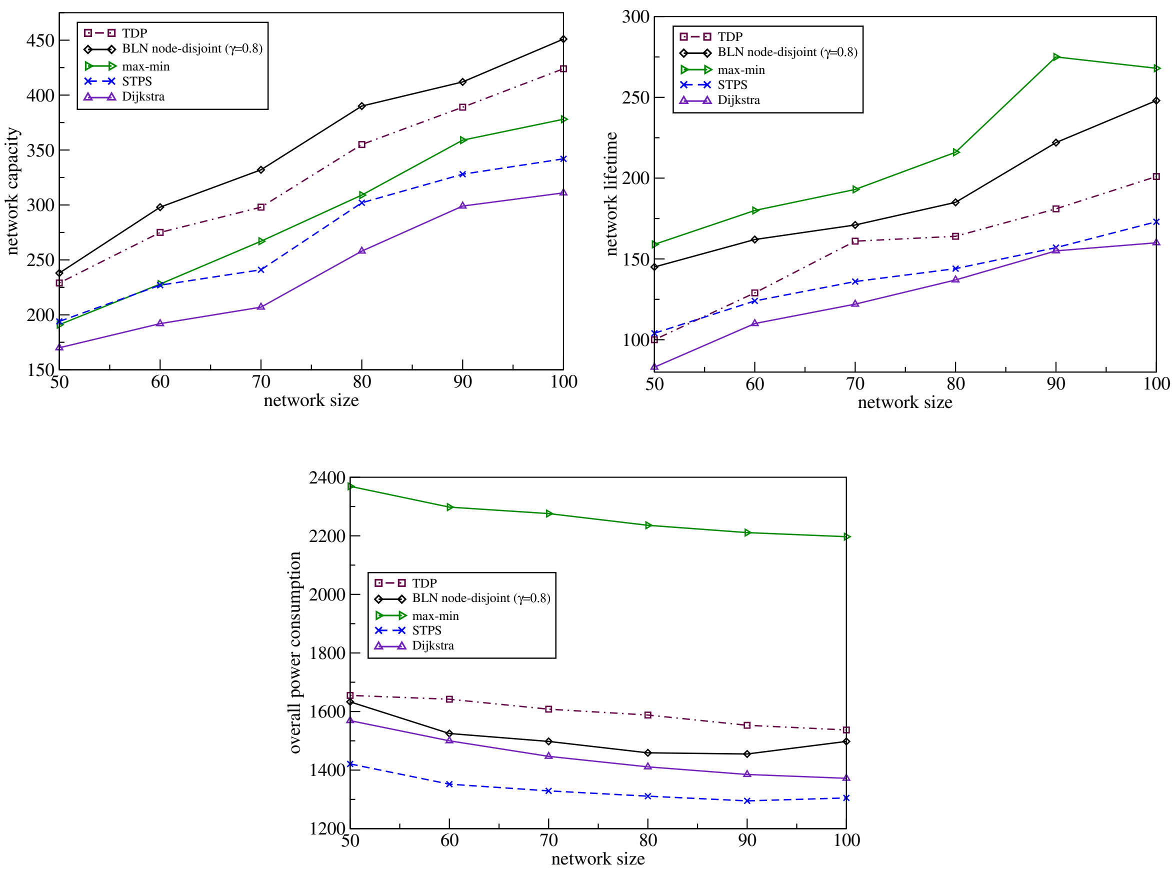

Fig. 5. Network capacity, network lifetime, and the energy consumption per request, under various node-disjoint algorithms.

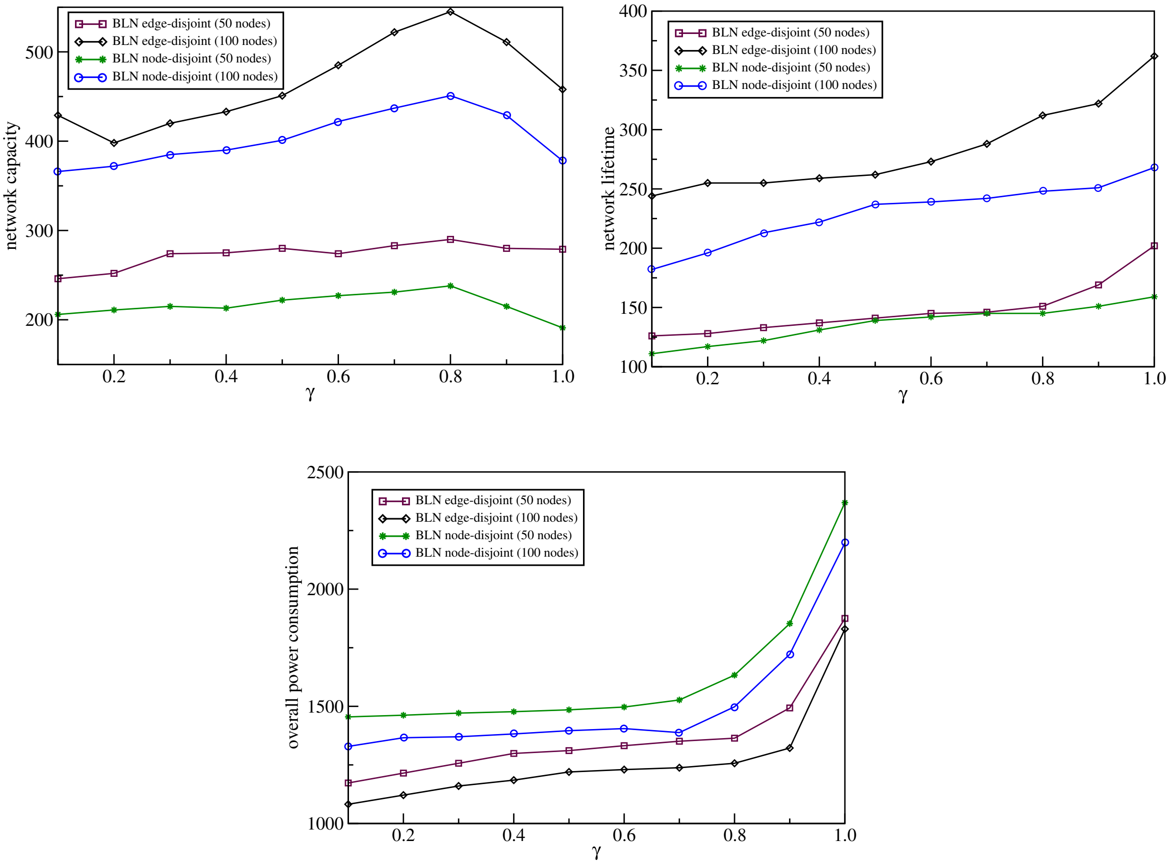

Fig. 6. Network capacity, network lifetime, and the energy consumption per request of various γs for algorithm BLN.

remains relatively low. The network capacity delivered by algorithm TDP is below the one by BLN but outperforms all the other algorithms. Although the network lifetime delivered by TDP is not as good as expected, it is much better than the one delivered by the approach based on Dijkstra's algorithm.

We then compare the performance of the two proposed algorithms for the problem in the node-disjoint version against the other existing algorithms, the Source Transmit Power Selection STPS in [23] and an approach based on Dijkstra's algorithm. The results in Fig. 5 show that algorithm max–min delivers a solution with the longest network lifetime in all the cases, and algorithm BLN delivers a solution with the largest network capacity when $\gamma \approx 0.8$.

We finally analyze the impact of different values of γ 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. 6. We observed that the network capacity delivered by algorithm BLN is maximized when $\gamma \approx 0.8$. Although algorithm BLN delivers the maximum local network lifetime when $\gamma = 1.0$, the energy consumption per request by it grows significantly compared with that when $\gamma \approx 0.8$.

6. Conclusions

In this paper we have considered on-line disjoint path routing for network capacity maximization by proposing energy-efficient on-line algorithms. We have also conducted extensive experiments by simulations. The experimental results show that the proposed algorithms outperform those existing algorithms in terms of maximizing network capacity.

Acknowledgement

We appreciate Mr. Xiaoxing Guo for implementing the proposed algorithms and evaluating their performance against the ones of existing algorithms. Also, we would like to thank the anonymous referees for their constructive comments and suggestions which have helped improve the quality and presentation of the paper.

References

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

[2] F. Bian, A. Goel, C.S. Raghavendra, X. Li, Energy-efficient broadcasting in wireless ad hoc networks: lower bounds and algorithms, J. Interconnection Networks 3 (2002) 149–166.

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

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

[5] J.-H. Chang, L. Tassiulas, Energy conserving routing in wireless ad hoc networks, in: Proc. INFOCOM'00, IEEE, 2000.

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

[7] A.K. Das, R.J. Marks, M. El-Sharkawi, P. Arabshahi, A. Gray, Minimum power broadcast trees for wireless networks: integer programming formulations, in: Proc. INFOCOM'03, IEEE, 2003, pp. 1001–1010.

[8] P. Gupta, P.R. Kumar, Critical power for asymptotic connectivity in wireless networks, Anal., Control, Optim. Appl.: A Volume in Honor of W.H. Fleming (1998) 547–566.

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

[10] I. Kang, R. Poovendran, Maximizing static network lifetime of wireless broadcast ad hoc networks, in: Proc. ICC'03, IEEE, 2003.

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

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

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

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

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

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

[17] R.G. Ogier, V. Rutenburg, N. Shacham, Distributed algorithms for computing shortest pairs of disjoint paths, IEEE Trans. Inform. Theory 39 (1993) 443–455.

[18] P. Papadimitratos, Z.J. Haas, Secure message transmission in mobile ad hoc networks, J. Ad Hoc Networks 1 (2003) 193–209.

[19] V.D. Park, M.S. Corson, A highly adaptive distributed routing algorithm for mobile wireless networks, in: Proc. INFOCOM'97, IEEE, 1997, pp. 1405–1413.

[20] S. Plotkin, Competitive routing of virtual circuits in ATM networks, IEEE J. SAC 13 (1995) 1128–1136.

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

[22] A. Sankar, Z. Liu, Maximum lifetime routing in wireless ad-hoc networks, in: Proc. INFOCOM'04, IEEE, 2004.

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

[24] J.W. Suurballe, Disjoint paths in a network, Networks 4 (1974) 125–145.

[25] J.W. Suurballe, R.E. Tarjan, A quick method for finding shortest pairs of disjoint paths, Networks 14 (1984) 325–336.

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

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

[28] J.E. Wieselthier, G.D. Nguyen, A. Ephremides, Resource management in energy-limited, bandwidth-limited, transceiver-limited wireless networks for session-based multicasting, Comput. Networks 39 (2002) 113–131.

Weifa Liang received the Ph.D. degree from the Australian National University in 1998, the M.Eng. degree from the University of Science and Technology of China in 1989, and the B.Sc. degree from Wuhan University, China in 1984, all in Computer Science. He is currently a Senior Lecturer in the Department of Computer Science at the Australian National University. His research interests include the design of energy-efficient routing protocols for wireless ad hoc and sensor networks, routing protocol design for WDM optical networks, design and analysis of parallel and distributed algorithms, data warehousing and OLAP, query optimization, and graph theory.

Yuzhen Liu received the M.Eng. degree and B.Sc. degree from Wuhan University in China, both in Computer Science. She is currently pursuing the Ph.D. degree in the Department of Computer Science. Her research interests include the design and analysis of routing algorithms for wireless ad hoc and sensor networks, design and analysis of network security protocols, trusted computing, embedded systems, and graph theory.