Power-aware localized routing in wireless networks

Ivan Stojmenovic and Xu Lin
Computer Science, SITE, University of Ottawa, Ottawa, Ontario K1N 6N5, Canada
ivan@site.uottawa.ca

Abstract

We discuss routing algorithms for wireless networks with the goal of increasing the network and node life. Recently, a cost aware metric based on remaining battery power at nodes was proposed and verified for non-localized shortest-cost routing algorithms, assuming constant transmission power. Two metrics where transmission power depends on distance between nodes were also recently proposed. We define a new power-cost metric based on the combination of both node's lifetime and distance based power metrics. We then propose power, cost, and power-cost GPS based fully distributed (i.e. localized) routing algorithms, where nodes make routing decisions solely on the basis of location of their neighbors and destination. Power aware localized routing algorithm attempts to minimize the total power needed to route a message between a source and a destination. Cost-aware localized algorithm is aimed at extending battery's worst case lifetime. The combined power-cost localized routing algorithm attempts to minimize the total power needed and to avoid nodes with short battery's remaining lifetime. We prove that the proposed localized power, cost, and power-cost efficient routing algorithms are loop-free, and show their efficiency by experiments.

1. Introduction

In this paper we consider the routing task, in which a message is to be sent from a source node to a destination node (in a sensor or ad hoc wireless network). The nodes in the network may be static (e.g. thrown from an aircraft to a remote terrain or a toxic environment), static most of the time (e.g. books, projectors, furniture, motors) or moving (vehicles, people, small robotic devices).

Wireless networks of sensors are likely to be widely deployed in the near future because they greatly extend our ability to monitor and control the physical environment from remote locations and improve our accuracy of information obtained via collaboration among sensor nodes and online information processing at those nodes. Networking these sensors (empowering them with the ability to coordinate amongst themselves on a larger sensing task) will revolutionize information gathering and processing in many situations. Sensor networks have been recently studied in [EGHK, HCB, HKB, KKP]. A similar wireless network that received significant attention in recent years is ad hoc network [IETF, MC]. Mobile ad hoc networks (MANETs) consist of wireless hosts that communicate with each other in the absence of a fixed infrastructure. Routes between two hosts in MANET may consist of hops through other hosts in the network. Some examples of the possible uses of ad hoc networking include soldiers on the battlefield, emergency disaster relief personnel, network of laptops at conferences and meetings etc.

The task of finding and maintaining routes in MANET is nontrivial since host mobility causes frequent unpredictable topological changes. Most previously proposed position based routing algorithms (e.g. [BCSW, KV]) for wireless ad hoc networks were based on forwarding the actual message along multiple paths toward an area where destination is hopefully located. The significant

communication overhead can be avoided if the routing strategy is changed, as suggested in [SV]. In the strategy [SV], the source node issues several search 'tickets' (each ticket is a 'short' message containing sender's id and location, destination's best known location and time when that location was reported, and constant amount of additional information) that will look for the exact position of destination node. When the first ticket arrives at the destination node *D, D* will report back to source with brief message containing its exact location, and possibly creating a route for the source. The source node then sends full data message ('long' message) toward exact location of destination. The efficiency of destination search depends on the corresponding location update scheme. A quorum based location update scheme is being developed in [SV]. Other schemes may be used, with various trade-off between the success and flooding rates (including an occasional fool flooding). If the routing problem is divided as described, the mobility issue is separated from the routing issue, which allows us to consider (in this paper) only the case of static networks with known destination in our algorithms and experiments. The choice is justified whenever the mobility rate is not very high (that is, when the transmission speed is significantly larger than the mobility speed, so that route between source and destination does not change during the destination search and message delivery) and information about neighboring nodes is regularly maintained. Another reason is that routing algorithm that performs well for moving nodes must also perform well on static networks, and therefore initial experiments and design of routing algorithm may begin with the static networks. Experiments with moving networks will then combine best routing methods (from experiments on static networks) with best location update schemes.

While the computational power of the devices used in MANETs is rapidly increasing, the lifetime of batteries is not expected to improve much in the future. The average life of batteries in an idle cellular phone is one day [MG]. The DEC Roamabout radio [h] consumes approximately 5.76 watts during transmission, 2.88 watts during reception and 0.35 watts when idle. The radio used in [GFM] consumes 15 watts while transmitting, 11 watts while receiving and 50mW in idle mode. Low power displays, I/O devices, CPU's and algorithms to reduce power consumption of disk drives are being developed (cf. [SWR]). We see a clear need for improvement in the power consumption in existing MAC protocols and routing algorithms: in almost all of the current protocols, nodes are powered on most of the time even when they are doing no useful work [SC, SR, SWR]. Power saving asynchronous communication protocols were discussed in [CZ, SC, SR].

Global Position System (GPS) provides location information (latitude, longitude and possibly height) to hosts in ad hoc wireless network. The location information allows each node to adjust the transmission power to the minimal needed for successful reception. GPS cards will be, in the near future, deployed in each car and possibly in every user terminal [K, NI]. For instance, NAVSTAR Global Positioning System has a potential accuracy of about 50-100 meters and Differential GPS offers accuracy of a few meters [N]. The recent low-power implementation of a GPS receiver [SSL] makes its presence a viable option in minimum energy network design.

To accurately model the attenuation of radio waves between antennas close to the ground, radio engineers typically use a model that attenuates the power of signal as $1/r^2$ at short distances (free space propagation model) and as $1/r^4$ at longer distances (two-ray ground reflection model), where *r* is the distance between antennas. The crossover point is called the reference distance, and is typically around 100 meters for outdoor low-gain antennas 1.5m above the ground plane operating in the 1-2GHz band [R]. In order for a message to be received correctly, the power of the signal should be above receive threshold. According to [PL], the power of signal received at a node is inversely proportional to the distance the receiver is from the transmitter, raised to an exponent known as the distance-power gradient, i.e. $P=T/d^\alpha$, where *P* is power of the received signal, *T* is the

power of the transmitted signal, d is the distance between the receiver and the transmitter, and α is the distance-power gradient. In an ideal situation $\alpha=2$. However, due to various environmental factors such as building materials, street layouts, terrain characteristics etc., the measured value of α may vary from less than two to more than six.

Several routing algorithms [HL, NK, TK] have used signal attenuation to determine probabilities of capturing transmitted signals, and proposed techniques to increase that probability. In order to reduce the negative impact of collisions, [CC] proposed the use of directional antennas (the packet is transmitted from the source within a particular angular range) instead of omnidirectional ones (the packet is transmitted to all nodes within a circle of given radius). In [KKKP], the signal attenuation was used to study the problem of assigning transmission ranges to the nodes of a multihop packet radio network so as to minimize the total power consumed under the constraint that adequate power is provided to the nodes to ensure that the network is strongly connected. It is proved in [GK] that n nodes placed in a disk of unit area define connected unit graph of radius R with probability one if and only if $R> (log\ n)/n$, when n approaches infinity. Various aspects of power management for wireless communication networks are reviewed in [RB].

Macker and Corson [MC] listed qualitative and quantitative independent metrics for judging the performance of routing protocols. Desirable qualitative properties include: distributed operation, loop-freedom, demand-based operation and 'sleep' period operation (for power conservation or other reasons, some nodes may become temporarily inactive, and routing protocol should be able to accommodate such sleep periods without overly adverse consequences). In this paper we use two quantitative metrics [BMJHJ, SL] (each of them is an average value):
- *hop count* (the number of edges, i.e. transmissions on the path from source to destination),
- *delivery rate* (the ratio of numbers of messages received by destination and sent by senders).

We shall also use several power related metrics, which will be discussed in coming sections.

Ad hoc and sensor networks are best modeled by *minpower* graphs constructed in the following way. Each node A has its transmission range $t(A)$. Two nodes A and B in the network are neighbors (and thus joined by an edge) if the Euclidean distance between their coordinates in the network is less than the minimum between their transmission radii (i.e. $d(A,B) < min \{t(A),\ t(B)\}$) [BCSW]. If all transmission ranges are equal, the corresponding graph is known as the *unit graph*. These models provide for acknowledgments for received messages. The minpower and unit graphs are valid models when there are no obstacles in the signal path (e.g. a building). Ad hoc and sensor networks with obstacles can be modeled by subgraphs of minpower or unit graphs. The properties of power metrics, the proposed algorithms and their loop free properties discussed in this paper are valid for arbitrary graphs. We have used, however, only unit graphs in our experiments.

Ad hoc networks consist of autonomous nodes that run their routines in asynchronous fashion. The communication algorithms between nodes are therefore all distributed. However, Estrin, Govindan, Heidemann, and Kumar [EGHK] defined the class of so called *localized* algorithms, as distributed algorithms where simple local node behavior achieves a desired global objective. Localized algorithms therefore resemble the class of greedy sequential algorithms. They argued that localized algorithms may be necessary for sensor and ad hoc network coordination, and described localized algorithms for clustering and object location. We propose a more formal definition of localized algorithms. In a localized algorithm, each node makes the decision to which neighbor(s) to forward the message based solely on the location of itself, its neighboring nodes, and a *constant* amount of additional information. In addition, each node is allowed to perform local computation (since it is equipped with a small but powerful processor). The decision depends on the task performed and the algorithm followed, which is part of an incoming message. In case of

routing, the additional information needed is the location of the destination. The sender uses the latest available location about the position of the destination and attaches it to the message. Intermediate nodes may use their more recent location information to replace the one that comes with the message (and also to update their own information otherwise). Note that the requirement for the location of a destination means that each node should update locations of all other nodes in the network, so that any routing that is initiated will have a reasonably good start. However, the path adjustments can be made as the message travels closer to the destination.

Non-localized algorithms, on the other hand, typically require the knowledge of the locations of all nodes in the network, and also the information about the existence of every edge in the graph. All non-localized routing algorithms proposed in literature are variations of shortest weighted path algorithm (e.g. [CN, LL, RM, SWR]). Although ad hoc network is fairly accurately modeled by unit graphs, nodes that are at distance less than R may have an obstacle between them blocking the communication, while two nodes at distance that exceeds R by a small amount may still be able to communicate between them (or a node may even choose whether to use that possible but power demanding link). If any edge in the network breaks or emerges (due to node mobility), it may have impact in the whole network. In the worst case scenario, a path that used to be shortest may not even exist, and the algorithm does not offer immediately a reasonable alternative path unless the shortest path algorithm is rerun. Therefore the maintenance of shortest weighted path requires that information about edges, in addition to location of nodes, is broadcast to the whole network, which is a significant quadratic communication overhead, and may cause significant delays in delivery (before new paths are available, shortest weighted paths need to be recalculated). Next, some nodes in the sensor or ad hoc network may be temporarily inactive, and non-localized algorithms need to know which of the nodes are active to make their best decisions. The activity information puts additional demand on the information update. For example, static nodes may need to broadcast such information to the whole network whenever they change their activity status while, at the same time, they have no need to update their location with the rest of the nodes. In order to preserve battery power over long periods of time, nodes may change their activity on a regular basis, or it may depend on the size of the job assigned to them. Thus, non-localized shortest path algorithms may not be the best choice for a routing algorithm even in case of static networks (e.g. some kinds of sensor networks). While the absence of a GPS receiver at nodes forces many problems (routing, for example) to be solved by only non-localized algorithms, their application, when GPS is available, may not be justified if efficient localized alternatives are available. The power savings obtained by applying a localized rather than non-localized algorithm outweighs even the additional power imposed to nodes for carrying and using GPS receiver. This paper deals solely with localized algorithms.

In accordance with [BCSW, BMSU, KV, KSU, SL, TK], it is assumed that each node, in its routing table, contains the geographic location of all other nodes in network, including the time when the location for each node is established. Since nodes may move, the actual location may differ from the one recorded in the routing table, and, for each node, its location information in other nodes may be different. However, as already discussed, the routing table (in localized algorithms) is only used to provide approximate position of destination, and accurate information about location of neighboring nodes.

A number of MANET protocols for achieving efficient routing have been recently proposed. They differ in the approach used for searching a new route and/or modifying a known route, when hosts move. The surveys of these protocols, that do not use geographic location in the routing decisions, are given in [BMJHJ, RS, RT]. Some of the protocols are available on the internet

[IETF]. The power awareness in these protocols is limited to the amount of control messages sent and degree of message flooding. We will discuss only GPS based or power aware approaches.

In the next section, we shall review known power aware metrics and routing algorithms. In section 3, existing routing protocols [BCSW, KV, KSU, HL, NK, SWR, TK, SL] which use geographic location or consider power in their route decisions are reviewed. Section 4 proposes a generalized power aware metric and discusses the effect of power awareness on the routing decisions in GPS based algorithms. Section 5 proposes three distributed (localized) algorithms aimed at extending node and/or network life. In section 6, we prove that these algorithms are loop-free. Their performance evaluation is given in sections 7 and 8.

2. Existing power aware metrics and routing algorithms

In most of routing protocols the paths are computed based on minimizing hop count or delay. Some nodes participate in routing packets for many source-destination pairs, and the increased energy consumption may result in their failure. For example, hierarchical and spine routing algorithms will (by their own design) exploit nodes that lie on the spine in order to reduce message overhead in routing table maintenance. This metric of reducing message overhead may be misguided in the long term, as observed in [SWR]. A longer path that passes through nodes that have plenty of energy may be a better solution [SWR].

The algorithm [SWR] proposes to use a function $f(A)$ to denote node A's reluctance to forward packets, and to choose a path that minimizes the sum of $f(A)$ for nodes on the path. This routing protocol [SWR] addresses the issue of energy critical nodes. As a particular choice for f, [SWR] proposes $f(A)=1/g(A)$ where $g(A)$ denotes the remaining lifetime, normalized to be in the interval [0,1]. Thus reluctance grows significantly when lifetime approaches 0. The other metric used in [SWR] is aimed at minimizing the total energy consumed per packet. However, [SWR] merely observes that the routes selected when using this metric will be identical to routes selected by shortest hop routing, since the energy consumed in transmitting (and receiving) one packet over one hop is considered *constant*. In other words, the algorithm [SWR] that minimizes the total energy consumption is equivalent to the shortest hop count, since it is assumed that each transmission requires equal power. For each of the two proposed power consumption metrics (cost and hop count), [SWR] assigns weights to nodes or edges, and then refers to non-localized Dijkstra's algorithm for computing shortest weighted path between source and destination. We also observed that the validation of power aware metrics in [SWR] was not done on minpower graphs but on random graphs where each pair of nodes is joined by an edge with a fixed probability p.

Let P and T be the powers of received and transmitted signals, respectively. The well known relation states that $P=T/d^{\alpha}$, where d is distance between sender and receiver nodes, and α is a constant [R, PL]. We first observe that the expression is somewhat imprecise. First of all, if d is expressed in absolute measure (say, in meters) that the two sides of the equation do not have the same physical meaning (that is, if the left side is in watts then the right side is in watts per meter). Therefore d is a relative unit, that is, the exact distance between nodes is divided by some unit distance. Next, we observe that, for $d<1$, the received power is greater than transmitted one, which is physically impossible. Thus this model was rejected in favor of models described below.

In the simple radio model [HCB], radio dissipates E_{elec}=50 nJ/bit to run the transceiver circuitry. Both sender and receiver node consume E_{elec} to transmit one bit between them. Assuming d^2 energy loss, where d is the distance between nodes, transmit amplifier at the sender node consumes further $E_{amp}d^2$, where $E_{amp} = 100$ pJ/bit/m^2. Thus, to transmit one bit message at distance

d, the radio expends $E_{elec} + E_{amp}d^2$, and to receive the message, the radio expends E_{elec}. In order to normalize the constants, divide both expressions by E_{amp}, so that radio expands $T=E + d^2$ for transmission and $P=E$ for reception, where $E= E_{elec} /E_{amp}=500$ m^2. Note that $T/P= 1+ d^2/E$ and $T+P=2E+ d^2$. Therefore, in this model, referred to as *HCB-model,* the power needed for transmission and reception at distance d is $u(d)= 2E+ d^2$.

Rodoplu and Meng [RM] considered, in their experiments, the model with $u(d)= d^4+2*10^8$, which will be referred to as *RM-model.* They also proposed a general model in which the power consumption between two nodes at distance d is $u(d)=d^\alpha+c$ for some constants α and c, and describe several properties of power transmission that are used to find neighbors for which direct transmission is the best choice in terms of power consumption. They discuss that large-scale variations (modeled by lognormal shadowing model) can be incorporated into the path loss formula, and that small-scale variations (modeled by a Rayleigh distribution) may be handled by diversity techniques and combiners at the physical layer.

Rodoplu and Meng [RM] described a power aware routing algorithm which runs in two phases. In the first phase, each node searches for its neighbors and selects these neighbors for which direct transmission requires less power than if an intermediate node is used to retransmit the message. This defines so called *enclosure* graph. In the second phase, each possible destination runs distributed loop-free variant of (non-localized) Bellman-Ford shortest path algorithm and computes shortest path for each possible source. The same algorithm is run from each possible destination. We observe that, since the energy required to transmit from node A to node B is the same as energy needed to transmit from node B to node A, the same algorithm may be applied from each possible source, and used to discover the best possible route to each destination node (or, alternatively, it may be used to find the location of destination and the best route to it).

Ettus [E] showed that minimum consumed energy routing reduces latency and power consumption for wireless networks utilizing CDMA, compared to minumum transmitted energy algorithm (shortest path algorithm was used in experiments). Heizelman, Chandrakasan and Balakrishnan [HCB] used signal attenuation to design an energy efficient routing protocol for wireless microsensor networks, where destination is fixed and known to all nodes. They propose to utilize a 2-level hierarchy of forwarding nodes, where sensors form clusters and elect a random clusterhead. The clusterhead forwards transmissions from each sensor within its own cluster. This scheme is shown to save energy under some conditions. However, the scheme does not apply to our case since destination is not fixed in our routing problem. Nevertheless, their simple radio model and metric is adopted in our paper.

3. Existing GPS based routing methods

Most existing routing algorithms do not consider the power consumption in their routing decisions. Several GPS based methods were proposed in 1984-86 by using the notion of progress. Define progress as the distance between the transmitting node and receiving node projected onto a line drawn from transmitter toward the final destination. A neighbor is in forward direction if the progress is positive (for example, for transmitting node S and receiving nodes A, C and F in Fig. 1); otherwise it is said to be in backward direction (e.g. nodes B and E in Fig. 1). In the random progress method [NK], packets destined toward D are routed with equal probability towards one intermediate neighboring node that has positive progress. The rationale for the method is that, if all nodes are sending packets frequently, probability of collision grows with the distance between nodes (assuming that the transmission power is adjusted to the minimal possible), and thus there is

a trade-off between the progress and transmission success. In the *NFP* method [HL], packet is sent to the nearest neighboring node with forward progress (for instance, to node *C* in Fig. 1). Takagi and Kleinrock [TK] proposed *MFR* (most forward within radius) routing algorithm, in which packet is sent to the neighbor with the greatest progress. In [HL], the method is modified by proposing to adjust the transmission power to the distance between the two nodes.
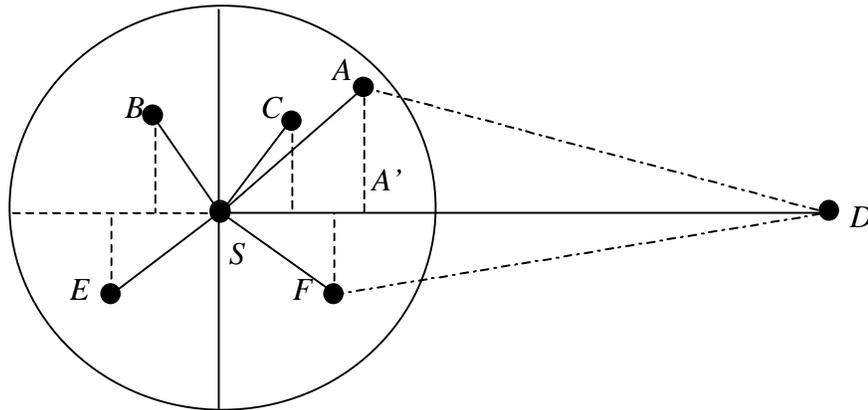

Figure 1. Progress based routing methods

Recently, three articles [BCSW, KV, KSU] independently reported variations of fully distributed routing protocols based on direction of destination. In the *compass routing* (or *DIR*) method proposed by Kranakis, Singh and Urrutia [KSU], the source or intermediate node *A* uses the location information for the destination *D* to calculate its direction. The location of one hop neighbors of *A* is used to determine for which of them, say *C,* is the direction *AC* closest to the direction of *AD*. The message *m* is forwarded to *C*. This process repeats until the destination is, hopefully, reached. A counterexample showing that undetected loops can be created in directional based method is given in [SL]. The method is therefore not loop-free.

[SL] introduced the *GEDIR* routing algorithm for a MANET based on the locations (latitude and longitude) of all nodes. When node *A* wants to send a message to node *B,* it uses the location information for *B* and for all its one hop neighbors to determine the neighbor *C* which is closest to *B* among all neighbors of *A*. The message is forwarded to *C,* and the same procedure is repeated until *B,* if possible, is eventually reached. *GEDIR* algorithm is inherently loop-free [SL]. The proof is based on the observation that distances of nodes toward destination are decreasing. Similarly, the proof of loop-free *MFR* algorithm is based on a decreasing dot product. All described GPS based routing algorithms are fully distributed, demand-based and adapt well to 'sleep' period operation.

The 2-hop variants of three mentioned GPS based routing algorithms were proposed in [SL]. The delivery rate of *GEDIR*, compass routing (or *DIR*) or *MFR* algorithms can be improved if each node is aware of its 2-hop neighbors (neighbors of its neighbors). In this case, the node *A* currently holding the message may choose the node closest to the destination among all direct (1-hop) and 2-hop neighbors, and forward the message to its neighbor that is connected to the choice. In case of ties (that is, more than one neighbor connected to the closest 2-hop neighbor), choose the one that is closest to destination. There are no multiple copies of the message in MANET at any transmission step in these 2-hop methods. A routing algorithm that guarantees delivery by finding a simple path between source and destination (without any flooding effect) is described in [BMSU].

This review did not include various flooding based or multiple paths routing algorithms or methods for sending control messages to update positions [BCSW, KV, SV]. Our primary interest in this paper is to examine power consumption under assumption that nodes have accurate information about the location of their neighbors and destination node (e.g. static networks or networks with superb location update scheme). Significant power savings are confirmed experimentally for networks of average node degree 10, which have very high delivery rates.

4. Some properties of power adjusted transmissions

In this section we shall study the optimality of power adjusted transmissions in MANETs, using a simple and general radio model. Three specific models (that is, metric for power consumption for sending and receiving message between two nodes at given distance) are considered in order to show the generality of some properties of power adjusted transmissions and generality of power aware routing algorithms to be described in the next section.

Two power aware metrics [RM, HCB] are described in section 2. We shall here describe a third one, by modifying the expression $P=T/d^{\alpha}$ as follows: $P=T/(1+d)^{\alpha}$; thus the received power is almost equal to the transmitted one when nodes are near each other (d near 0). Let d_0 be the distance that corresponds to the received power $P=T/2^{\alpha}$; that is, to $d=1$ in the expression. The value d_0 and the exponent a can be measured by physical experiments. The relative distance d is equal to the exact distance between nodes divided by d_0. In order to simplify our discussion, we shall assume $d_0=1$, thus d will be equated with the distance between nodes. Since all transmission powers are adjusted to minimum needed for successful transmission, P is a constant. Thus the minimum power needed at the sender's node for successful transmission is $T=P(1+d)^{\alpha}$. We shall assume $\alpha=2$ in the sequel. The results for different values of α can be derived similarly. Nodes consume energy to receive packets. That energy is also constant (for constant packet length). We shall assume that the power needed for packet reception is eP, where e is constant that depends on the kind of equipment. For DEC Roambout radio [h] $e=1/2$, which is the value used in our experiments. To simplify further, we assume $P=1$ in the expressions that refer to power consumption. Thus, in this model, the power needed for transmission and reception at distance d is $u(d)=e+(1+d)^2$.

In order to include models that attenuate the power of signal of various exponents, we shall further generalize the model of [RM] and assume that the power needed for the transmission and reception of a signal is $u(d)=ad^{\alpha} + bd +c$. This model will encompass all three discussed models. In *RM-model* $\alpha=4$, $a=1$, $b=0$, $c=2*10^8$, in the *HCB-model* $\alpha=2$, $a=1$, $b=0$, $c=2E$, while in the third model described above $a=1$, $b=2$, $c= e+1$. Constant factor c in this expression for total energy consumption may also include the energy consumed in computer processing and encoding/decoding at each station. Next, the leading coefficient a can be adjusted to the physical environment, unit of length considered, unit size of a signal (a bit, byte, or frame, for example) etc.
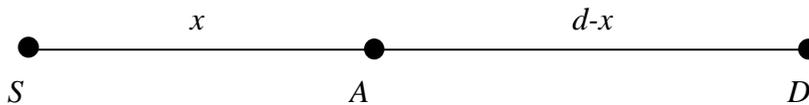

Figure 2. Saving energy by retransmission

Suppose that the sender S is able to transmit the packet directly to the destination D. Let us examine whether energy can be saved by sending the packet to an intermediate node A between the nodes and forwarding the packet from A to D (see Figure 2). Let $|SD|=d$, $|SA|=x$, $|AD|=d-x$.

Lemma 1. If $d>(c/(a(1-2^{1-\alpha})))^{1/\alpha}$ then there exists intermediate node A between source S and destination D so that the retransmission will save the energy. The greatest power saving is obtained when A is in the middle of SD.

Proof. The power needed to send message directly from S to D is $u(d)=ad^{\alpha} + bd + c$ while the power needed to send via A is $(ax^{\alpha} + bx + c) + (a(d-x)^{\alpha} + b(d-x) + c) = a(x^{\alpha} + (d-x)^{\alpha}) + bd + 2c$. $a(x^{\alpha} + (d-x)^{\alpha}) + bd + 2c < ad^{\alpha} + bd + c$ is satisfied for $g(x)=a(x^{\alpha} + (d-x)^{\alpha} - d^{\alpha}) + c < 0$. The minimum for $g(x)$ is obtained for $g'(x)=0$, i.e. $a(\alpha x^{\alpha-1} - \alpha(d-x)^{\alpha-1})=0$. Thus $x^{\alpha-1} = (d-x)^{\alpha-1}$, $x=d-x$, or $x=d/2$. The minimum is <0 if $g(d/2)<0$, i.e. $a((d/2)^{\alpha} + (d/2)^{\alpha} - d^{\alpha}) + c < 0$, or $ad^{\alpha}(2^{1-\alpha} -1)+c <0$, which is satisfied for $c< ad^{\alpha}(1-2^{1-\alpha})$, or $d^{\alpha} > c/(a(1-2^{1-\alpha}))$, and lemma follows. Note that this inequality has a solution in d if and only if $\alpha>1$.

When $\alpha=2$, the following corollary follows for the two methods that we adopted. We will give its full proof.

Corollary 1. If $d>(2c/a)^{1/2}$ then there exists intermediate node A between source S and destination D so that the retransmission will save the energy. The length of the interval of power saving intermediate nodes is $(d^2-2c/a)^{1/2}$. The greatest power saving is obtained when A is in the middle of SD. For the first method $c/a=e+1$, while for the second one $c/a=2E$.

Proof. The power needed to send message directly from S to D is $u(d)=ad^2 + bd + c$ while the power needed to send via A is $ax^2 + bx + c + a(d-x)^2 + b(d-x) + c = 2ax^2 -2adx + 2c + bd + ad^2$. $2ax^2 -2adx + 2c + bd + ad^2 < ad^2 + bd + c$ is satisfied for $2ax^2 - 2adx + c < 0$. The interval of power saving intermediate nodes is determined by solving the quadratic equation in x, and the lemma follows. Note that for $e=1/2$ in the first method one gets the threshold value $d> 3^{1/2}=1.73$, while for $E=500$ in the second method we receive $d > 2(500^{1/2})=44.72$.

Lemma 2. If $d>(c/(a(1-2^{1-\alpha})))^{1/\alpha}$ then the greatest power savings are obtained when the interval SD is divided into $n>1$ equal subintervals, where n is the nearest integer to $d(a(\alpha-1)/c)^{1/\alpha}$. The minimal power is then $bd + dc(a(\alpha-1)/c)^{1/\alpha} + da(a(\alpha-1)/c)^{(1-\alpha)/\alpha}$.

Proof. Let SD be divided into intervals of lengths $x_1, x_2, ..., x_n$ such that $d=x_1+ ... + x_n$. The energy needed for transmissions using these intervals is $(ax_1^{\alpha}+bx_1+c) + ... + (ax_n^{\alpha}+bx_n+c) = nc + bd + a(x_1^{\alpha} + ... + x_n^{\alpha})$. For fixed x_i+x_j, the expression $x_i^{\alpha} + x_j^{\alpha}$ is minimal when $x_i=x_j$ (see the proof of Lemma 1). Therefore the energy is minimal when $x_1=x_2= ... = x_n=d/n$ and is equal to $f(n)=cn + bd + an(d/n)^{\alpha} = nc + an^{1-\alpha}d^{\alpha} + bd$. This expression has the minimum when $f'(n)=0$, or when $c+a(1-\alpha)n^{-\alpha}d^{\alpha}=0$. i.e. $c=a(\alpha-1)n^{-\alpha}d^{\alpha}$, $n^{\alpha}=a(\alpha-1)d^{\alpha}/c$, $n=d(a(\alpha-1)/c)^{1/\alpha}$. Since n must be integer, one should round the value.

Corollary 2. Let $\alpha=2$. If $d > (2c/a)^{1/2}$ then the greatest power savings are obtained when the interval SD is divided into $n>1$ equal subintervals, where n is the nearest integer to $d(a/c)^{1/2}$.

Proof. Follows directly from Lemma 2. Let us verify whether power savings over direct transmission are obtained. For $n=d(a/c)^{1/2}$ the power consumption with n shorter transmissions is $2cd(a/c)^{1/2} + bd < ad^2 + bd + c$ since $2d(ac)^{1/2} < ad^2 + c$ follows from the inequality of arithmetic and geometric means. Thus power savings are indeed obtained.

Assuming that we can set additional nodes in arbitrary positions between the source and destination, the following theorem gives power optimal packet transmissions.

Theorem 1. Let d be the distance between the source and the destination. The power needed for direct transmission is $u(d)=ad^\alpha + bd +c$ which is optimal if $d \leq (c/(a(1-2^{1-\alpha})))^{1/\alpha}$. Otherwise (that is, when $d > (c/(a(1-2^{1-\alpha})))^{1/\alpha}$), $n-1$ equally spaced nodes can be selected for retransmissions, where $n= d(a(\alpha-1)/c)^{1/\alpha}$ (rounded to nearest integer), producing minimal power consumption of about $v(d)= bd + dc(a(\alpha-1)/c)^{1/\alpha} + da(a(\alpha-1)/c)^{(1-\alpha)/\alpha}$. For $\alpha=2$, the power needed for direct transmission is $u(d)=ad^2 + bd +c$ which is optimal if $d \leq (2c/a)^{1/2}$. Otherwise (that is, when $d > (2c/a)^{1/2}$), $n-1$ equally spaced nodes can be selected for retransmissions, where $n= d(a/c)^{1/2}$ (rounded to nearest integer), producing minimal power consumption of about $v(d)=2d(ac)^{1/2}+ bd$.

Theorem 1 announces the possibility of converting polynomial function in d (with exponent α) for power consumption (in case of direct transmission from sender to destination) to linear function in d by retransmitting the packet via some intermediate nodes that may be available in MANET.

5. *Power saving routing algorithms*

If nodes have information about position and activity of all other nodes in the network (or if the decisions are made centrally) then optimal power saving algorithm, that will minimize the total energy per packet, can be obtained by applying Dijkstra's single source shortest weighted path algorithm, where each edge has weight $u(d)=ad^\alpha + bd +c$, where d is the length of the edge (that is, the relative distance between the two nodes). This will be referred as the *SP-power algorithm.*
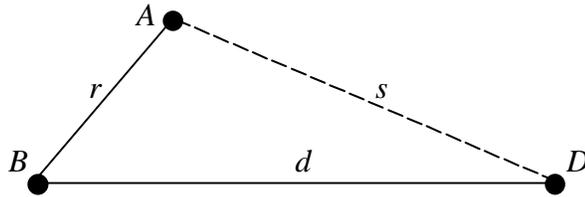

Figure 3. Distributed power-conserving routing algorithm

We shall now describe a corresponding localized routing algorithm. The source (or any intermediate node) S should select one of its neighbors to forward packet toward destination, with the goal of reducing the total power needed for the packet transmission. Let A be a neighbor of B, and let $r=|AB|$, $d=|BD|$, $s=|AD|$ (see Figure 3).

The power needed for transmission from B to A is $u(r)=ar^\alpha + br +c$, while the power needed for the rest of routing algorithm is not known. Assuming uniformly distributed network, we shall make a fair assumption that the power consumption for the rest of routing algorithm is equal to the optimal one, as outlined in Theorem 1. That is, the power needed for transmitting message from A

to D is estimated to be $v(s)= bs + sc(a(\alpha-1)/c)^{1/\alpha} + sa(a(\alpha-1)/c)^{(1-\alpha)/\alpha}$. For $\alpha=2$, $v(s)=2s(ac)^{1/2} + bs$. This is, of course, an unrealistic assumption. However, it is fair to all nodes. A more realistic assumption might be to multiply the optimal power consumption by a factor t, which is a constant that depends on the network, and is equal to the average power consumption per packet (obtained experimentally) divided by the optimal one.

The localized *power efficient routing algorithm* can be described as follows. Each node B (source or intermediate node) will select one of its neighbors A which will minimize $p(B,A)=u(r)+v(s)= ar^{\alpha} + br +c + bs + sc(a(\alpha-1)/c)^{1/\alpha} + sa(a(\alpha-1)/c)^{(1-\alpha)/\alpha}$. For $\alpha=2$ it becomes $u(r)+v(s)= ar^2 + br +c + 2s(ac)^{1/2} + bs$. If destination D is a neighbor of B then compare the expression with the corresponding one, $u(d)=ad^2 + bd +c$, needed for direct transmission. Deliver the packet directly to D if it reduces the energy. In fact, $s=0$ for D, and D can be treated as any other neighbor. The algorithm proceeds until the destination is reached, if possible. A generalized power efficient routing algorithm may attempt to minimize $p(B,A)=u(r)+tv(s)$, where t is a network parameter (however, we only experimented with $t=1$).

In the basic (experimental) version of the algorithm (and in the remaining localized algorithms presented below), the transmission stops if message is to be returned to a neighbor it came from (otherwise, a detectable loop is created). Various flooding based or multiple path techniques [SL] can be added to the protocol if delivery rate is to be improved; however, their total power consumption needs to be studied. The power-efficient routing algorithm may be formalized as follows.

Power-routing(S,D);
    A:=S;
    Repeat
        B:=A;
        Let A be neighbor of B that minimizes $p(B,A)=u(r)+ tv(s)$;
        Send message to A
    until $A=D$ (* destination reached *) or $A=B$ (* delivery failed *);

Let us now consider the second metric proposed in [SWR], measuring the nodes lifetime. Recall that the cost of each node is equal to $f(A)=1/g(A)$ where $g(A)$ denotes the remaining lifetime, normalized to be in the interval [0,1]. [SWR] proposed shortest weighted path algorithm based on this node cost. It is referred to as the *SP-cost* algorithm in experimental data on Table 2. The algorithm uses the cost to select the path, but the actual power is charged to nodes.

The localized version of this algorithm, assuming constant power for each transmission, can be designed as follows. The cost $c(A)$ of a route from B to D via neighboring node A is the sum of the cost $f(A) =1/g(A)$ of node A and the estimated cost of route from A to D. The cost $f(A)$ of each neighbor A of node B currently holding the packet is known to B. What is the cost of other nodes on the remaining path? We assume that this cost is proportional to the number of hops between A and D. The number of hops, in turn, is proportional to the distance $s=|AD|$, and inversely proportional to radius R. Thus the cost is ts/R, where factor t is to be investigated separately. Its best choice might even be determined by experiments. We have considered the following choices for factor t:

i) t is a constant number, which may depend on network conditions,
ii) $t= f(A)$ (that is, assuming that remaining nodes have equal cost as A itself),
iii) $t= f'(A)$, where $f'(A)$ is the average value of $f(X)$ for A and all neighbors X of A,

iv) $t=1/g'(A)$, where $g'(A)$ is the average value of $g(X)$ for *A* and all neighbors *X* of *A*,
v) $t=f'(B)$ *(that is, assuming that remaining nodes have equal cost to the cost of B),*
vi) $t=1/g'(B)$.

Note that $t=t(A)$ in i-iv is a function of *A* only, which is not the case for v-vi. Thus the cost *c(A)* of a route from *S* to *D* via neighboring node *A* is estimated to be *c(A)=f(A)+ts/R,* for the appropriate choice of *t.* Although this seems to be the natural choice, it is not clear whether such definition of *c(A)* will give the best experimental results. We therefore suggest to investigate also the product of two contributing elements instead of their sum, that is the cost definition *c(A)= f(A)ts/R.*

The localized *cost efficient routing algorithm* can be described as follows. If destination is one of neighbors of node *B* currently holding the packet then the packet will be delivered to *D.* Otherwise, *B* will select one of its neighbors *A* which will minimize *c(A).* The algorithm proceeds until the destination is reached, if possible, or until a node selects the neighbor the message came from as its best option to forward the message. The algorithm can be coded as follows.

> *Cost-routing(S,D);*
>   *A:=S;*
>   **Repeat**
>   *B:=A;*
>   Let *A* be neighbor of *B* that minimizes *c(A)*;
>   **If**   *D* is neighbor of *B*
>   **then** send to *D* **else** send to *A*
>   **until** *D* is reached or *A=B;*

We may incorporate both power and cost considerations into a single routing algorithm. A new power-cost metrics is first introduced here. What is the power-cost of sending a message from node *B* to node the neighboring node *A*? We propose two different ways to combine power and cost metrics into a single power-cost metric, based on the product and sum of two metrics, respectively. If the product is used, then the power-cost of sending message from *B* to a neighbor *A* is equal to *power-cost(B,A)=f(A)u(r)* (where *|AB|=r).* The *SP-power-cost* algorithm can find the optimal power-cost by applying single source shortest weighted path Dijkstra's algorithm (the node cost is transferred to the edge leading to the node). The sum, on the other hand, leads to a new metrics *power-cost(B,A)=αu(r) + βf(A),* for suitably selected values of α and β. For example, sender node *S* may fix $\alpha=f'(S)$ and $\beta=u(r')$, where *r'* is the average length of all edges going out of *S.* The values α and β are (in this version) determined by *S* and used, without change, by other nodes *B* on the same route. The corresponding SP-power-cost algorithm will also use the so defined metric, and is referred to as the *SP-power-cost1* algorithm in Table 2.

The *power-cost efficient routing* algorithm may be described as follows. Let *A* be the neighbor of *B* (node currently holding the message) that minimizes *pc(B,A)= power-cost(B,A) + v(s)f'(A)* (where *s=0* for *D,* if *D* is a neighbor of *B*). The algorithm is named *power-cost0* in Table 2 when *power-cost(B,A)=f(A)u(r),* and *power-cost1* when *power-cost(B,A)= f'(S)u(r)+u(r')f(A).* The packet is delivered to *A.* Thus the packet is not necessarily delivered to *D,* when *D* is a neighbor of *B.* The algorithm proceeds until the destination is reached, if possible, and may be coded as follows.

> *Power-cost-routing(S,D);*
>   *A:=S;*

Repeat

        B:=A;

        Let *A* be neighbor of *B* that minimizes *pc(B,A)= power-cost(B,A) + v(s)f'(A);*

        Send message to *A*

until *A=D (* destination reached *)* or *A=B (* delivery failed *);*

The algorithm may be modified in several ways. The second term may be multiplied by a factor that depends on network conditions. We tested also the version, called *power-cost2*, that minimizes *pc(B,A)=f(A)(u(r) + v(s)),* and an algorithm, called *power-costP,* that switches selection criteria from power-cost to power metric only whenever destination *D* is a neighbor of current node *A*.

6. Loop-free property

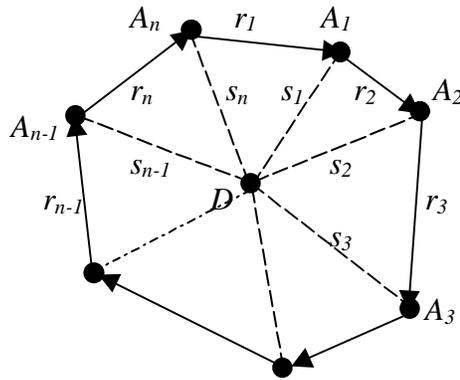**Theorem 2.** The localized power efficient routing algorithm is loop-free.

Figure 4. Power efficient routing algorithm is loop-free

Proof. Suppose that, on the contrary, there exists a loop in the algorithm. Let $A_1, A_2, ... A_n$ be the nodes in the loop, so that A_1 send the message to A_2, A_2 sends the message to A_3, ..., A_{n-1} sends the message to A_n and A_n sends the message to A_1 (see Fig. 4). Let $s_1, s_2, ..., s_n$ be the distances of A_1, $A_2, ... A_n$ from D, respectively, and let $|A_nA_1|=r_1$, $|A_1A_2|=r_2$, $|A_2A_3|=r_3$, ..., $|A_{n-1}A_n|=r_n$. Let $u(r)= ar^\alpha + br + c$ and $v(s) = bs + sc(a(\alpha-1)/c)^{1/\alpha} + sa(a(\alpha-1)/c)^{(1-\alpha)/\alpha}$ (for $\alpha=2$, $v(s)= 2s(ac)^{1/2} + bs$). According to the choice of neighbors in Fig. 4 it follows that $u(r_1)+v(s_1)<u(r_n)+v(s_{n-1})$ since the node A_n selects A_1, not A_{n-1}, to forward the message. Similarly $u(r_2)+v(s_2) < u(r_1)+v(s_n)$ since A_1 selects A_2 rather than A_n. Next, $u(r_3)+v(s_3) < u(r_2)+v(s_1)$, ..., $u(r_n)+v(s_n) < u(r_{n-1})+v(s_{n-2})$. By summing left and right sides we obtain $u(r_1)+u(r_2)+...+u(r_n)+v(s_1)+v(s_2)+...+v(s_n) < u(r_n)+u(r_1)+...+u(r_{n-1})+v(s_{n-1})+v(s_n)+...+v(s_{n-2})$ which is a contradiction since both sides contain the same elements. Thus the algorithm is loop-free.

In order to provide for loop-free method, we assume that (for this and other mentioned methods below), in case of ties for the choice of neighbors, if one of choices is the previous node, the algorithm will select that node (that is, it will stop or flood the message). Note that the above proof may be applied (by replacing '+' with '*') to an algorithm that will minimize $p(A)=u(r)tv(s)$.

Theorem 3. Localized cost efficient algorithms are loop-free, for parameter values *t* given by i-iv.

Proof. Note that the cost $c(A)$ of sending message from B to A in cases i-iv is only the function of A *(that is, $t=t(A)$),* and is independent on B (this is not the case with options v-vi). The proof is by contradiction, similar to the proof of previous theorem. Suppose that there exists a loop in the algorithm. Let $A_1, A_2, \ldots A_n$ be the nodes in the loop (see Fig. 4). Let $c(A_1), c(A_2), \ldots c(A_n)$, be the costs of sending message to nodes $A_1, A_2, \ldots A_n$, respectively, from the previous node in the loop. According to the choice of neighbors in Fig. 4 it follows that $c(A_1) < c(A_{n-1})$ since the node A_n selects A_1, not A_{n-1}, to forward the message. Similarly $c(A_2) < c(A_n)$ since A_1 selects A_2 rather than A_n. Next, $c(A_3) < c(A_1)$, …, $c(A_n) < c(A_{n-2})$. By summing left and right sides we obtain $c(A_1)+c(A_2)+\ldots+c(A_n) < c(A_{n-1})+c(A_n)+\ldots+c(A_{n-2})$ which is a contradiction since both sides contain the same elements. Thus the algorithm is loop-free. The proof is valid for both formulas $c(A)=f(A)+ts/R$ and $c(A)=f(A)ts/R$. Note that the proof assumes that the cost of each node is not updated (that is, communicated to the neighbors) while the routing algorithm is in progress. It is possible to show that, on the other hand, if nodes inform their neighbors about new cost after every transmitted message, a loop (e.g. triangle) can be formed.

Theorem 4. Localized power-cost efficient algorithms are loop-free for the metrics *power-cost$(B,A)=\alpha u(r) + \beta f(A)$* (where α and β are arbitrary constants), and $pc(B,A)= power\text{-}cost(B,A) + v(s)t(A)$ (where $t(A)$ is determined by one of formulas i-iv).

Proof. The proof is again by contradiction, similar to the proof of previous theorems. Suppose that there exists a loop $A_1, A_2, \ldots A_n$ in the algorithm (see Fig. 4). Let $pc(A_n, A_1), pc(A_1, A_2), \ldots pc(A_{n-1}, A_n)$, be the power-costs of sending message to nodes $A_1, A_2, \ldots A_n$, respectively, from the previous node in the loop. According to the choice of neighbors in Fig. 4 it follows that $pc(A_n, A_1) < pc(A_n, A_{n-1})$ since the node A_n selects A_1, not A_{n-1}, to forward the message. Similarly $pc(A_1, A_2) < pc(A_1, A_n)$, $pc(A_2, A_3) < pc(A_2, A_1)$, …, $pc(A_{n-1}, A_n) < pc(A_{n-1}, A_{n-2})$. By summing left and right sides we obtain $pc(A_n, A_1) + pc(A_1, A_2) + pc(A_2, A_3) + \ldots+ pc(A_{n-1}, A_n) < pc(A_n, A_{n-1}) + pc(A_1, A_n) + \ldots + pc(A_{n-1}, A_{n-2})$. This inequality is equivalent to $[\alpha u(r_n) + \beta f(A_1)+ v(s_1)t(A_1)] + [\alpha u(r_1) + \beta f(A_2)+ v(s_2)t(A_2)] + \ldots + [\alpha u(r_{n-1}) + \beta f(A_n)+ v(s_n)t(A_n)] < [\alpha u(r_n) + \beta f(A_{n-1})+ v(s_{n-1})t(A_{n-1})] + [\alpha u(r_1) + \beta f(A_n)+ v(s_n)t(A_n)] + \ldots + [\alpha u(r_{n-1}) + \beta f(A_{n-2})+ v(s_{n-2})t(A_{n-2})]$ which is a contradiction since both sides contain the same elements. Thus the algorithm is loop-free. Note that the proof also assumes that the cost of each node is not updated (that is, communicated to the neighbors) while the routing algorithm is in progress. Note that this proof does not work for the formula *power-cost$(B,A)=f(A)u(r)$,* which does not mean that the corresponding power-cost routing algorithm is not loop-free.

7. Performance evaluation of power efficient routing algorithm

The experiments are carried using random unit graphs. Each of n nodes is chosen by selecting its x and y coordinates at random in the interval *[0,m)*. In order to control the average node degree k (that is, the average number of neighbors), we sort all $n(n-1)/2$ (potential) edges in the network by their length, in increasing order. The radius R that corresponds to chosen value of k is equal to the length of $nk/2$-th edge in the sorted order. Generated graphs which were disconnected are ignored. We have fixed the number of nodes to $n=100$, and average node degree k to 10. We have selected higher connectivity for our experiments in order to provide for better delivery rates and hop counts and concentrate our study on power conserving effects.

The comparison of *DIR* (compass routing), *MFR* and *GEDIR* methods in [SL] did not depend on the size *m* of square containing all the points. However, in case of power consumption, the actual distances greatly impact the behavior of algorithms. More precisely, the path selection (and the energy for routing) in our power saving algorithm depends on the actual size of the square. Although the path selection in *DIR, MFR* and *GEDIR* methods are independent on *m,* the energy needed for routing differs, and is not simply proportional to *m.* We compared all methods for squares of sizes *m=*10, 100, 200, 500, 1000, 2000, 5000 for both *HCB-* and *RM-models*. The results are averages over 20 graphs with 100 routing pairs in each chosen at random.

In our comparisons, the power consumption (cost, power-cost, respectively) in all compared methods was measured by assigning the appropriate weights to each edge. The shortest weighted path algorithm was used as a benchmark. Our comparison for the category of power (only) consumption involved the following GPS based distributed algorithms: *NFP,* random progress, *MFR, DIR*, *GEDIR*, *NC,* the proposed distributed power efficient routing algorithm (with *t=1*), and the benchmark shortest (weighted) path algorithm (*SP*). A node, currently holding the message, will stop forwarding the message if the best choice, by the method, is to return the message to the node message came from. Such nodes are called concave nodes [SL], and delivery fails at such nodes.
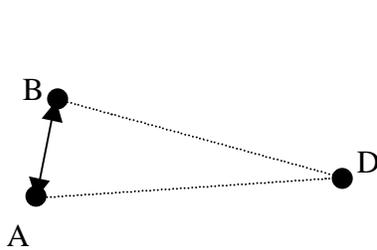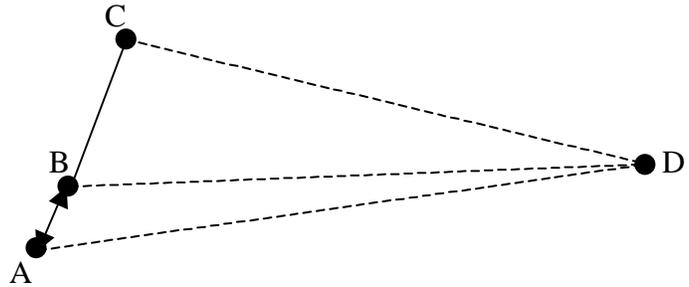

Figure 5. *NFP* method fails

Figure 6. *Power1* frequently fails

We have introduced a new routing method, called *NC* (nearest closer), in which node *A,* currently holding the message, forwards it to its nearest node among neighboring nodes which are closer to destination than *A*. This method is an alternative to the *NFP* method which was experimentally observed to have very low success rate (under 15% in our case). The reason for low success rate seems to be the existence of many acute triangles *ABD* (see Fig. 5) so that *A* and *B* are closest to each other, and therefore selected by *NFP* method which then fails at such nodes (*D* is the destination).

The proposed power efficient method, which will be referred as *power1* method, was also experimentally shown to have very low success rate for large *m.* The power efficient algorithm is therefore modified to increase its success rate. Only neighbors that are closer to destination than the current node are considered, and this variant will be called the *power* method. The success rates of *power* and *power1* methods are almost the same for *m≤200*. While the success rate of *power* method remains at 95% level, the success rate for *power1* drops to 59%, 11%, 4% and 2% only for remaining sizes of *m* (numbers refer to *HCB-model*, and are similar for the other model). Consider a scenario in which *power1* fails (see Fig. 6, where *|AD| < |BD| < |CD|*). Node *A* sends message to closest neighbor *B*. Since *A* is very close to *B* but *C* is not, power formula applied at *B* selects *A* to send message back, and a loop is created.

We included 2-hop *GEDIR, DIR, MFR* and *NC* methods in our experiments. 2-hop *NC* method is defined as follows. Each current node *C* finds the neighboring node *A* whose 1-hop nearest

(closer to destination *D*) neighbor *B* has the shortest distance (between *A* and *B*). If no such node exist (i.e. none of neighboring nodes of *C* has forward neighbor) then take the neighbor node *E* whose backward nearest neighbor *F* has smallest distance (between *E* and *F*).

The delivery rates for *1-GEDIR, 1-DIR,* and *1-MFR* methods in our experiments were about 97%, *1-NC* had 95% success, *2-GEDIR* (that is, *2-hop GEDIR*) and *2-MFR* about 99%, *2-DIR* about 91%, *2-NC* and random methods about 98%, and power method 95% success rate (for both *HCB-* and *RM-models).* While all other methods choose the same path independently on *m* and power formula applied, power method does not, and almost constant and good delivery rate for it is a very encouraging result. The hop counts for non-power based methods were 3.8, 4.2, 3.9, 8.0, 3.8, 3.9, 4.1, 5.2, and 6.4, respectively (in above order). Hop counts for power method were 3.8, 3.8, 3.8, 3.8, 6.3, 9.0 and 9.7 for *RM-model,* and 3.8, 3.8, 4.0, 6.6, 8.3, 9.1, 9.6 for *HCB-model,* in respective order of *m.* Clearly, with increased energy consumption per distance, power method reacted by choosing closer neighbors, resulting in higher hop counts.

Let us show the superiority of *GEDIR* method over *MFR* method and superiority of compass routing over random progress method. Let *A* and *B* be the nodes selected by the *GEDIR* and *MFR* methods, respectively, when packet is to be forwarded from node *S* (see Fig. 7). Suppose that *B* is different from *A* (otherwise the energy consumption at that step is the same). Therefore |*AD*|<|*BD*|. Node *B* cannot be selected within triangle *SAA'* where *A'* is the projection of node *A* on direction SD, since *B* has more progress than *A* (here we assume, for simplicity, that *A* and *B* are on the same side of line *SD*). However, the angle *SAB* is then obtuse, and |*SB*|>|*SA*|. From |*SB*| > |*SA*| and |*BD*| > |*AD*| and monotonicity of power consumption function, it follows that the packet requires more energy if forwarded to *B* instead of *A*. Note that the presented scenario is for an average case. It is possible to find a counterexample showing the opposite (or showing that a path via *B* exists while there is no path via *A*) in some very special cases.

Suppose now that *A* and *B* are selected neighbors in case of compass and random progress routing algorithms (we shall use the same Fig. 7). Since the lengths |*SA*| or |*SB*| are not considered when selecting the neighbors, on the average we may assume that |*SA*|=|*SB*|. However, the direction of *A* is closer to the direction of destination (that is, the angle *ASD* is smaller than the angle *BAD*) and thus *A* is closer to *D* than *B*. Therefore one can expect more energy efficient compass routing algorithms as compared to random progress one. Again, counterexamples may be easily found for particular cases.

Figure 7. GEDIR consumes less power than MFR
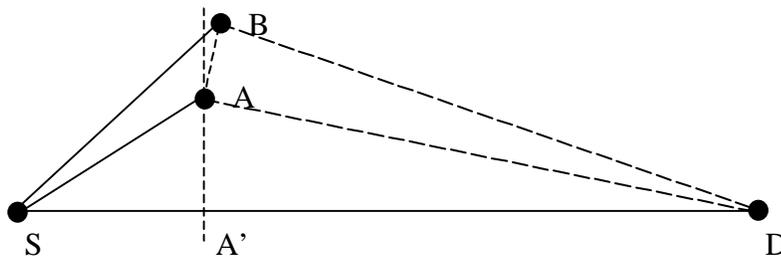
Table 1 shows average power assumption (rounded to nearest integers) per routing tasks that were successful by *all* methods, which occurs in about 85% of cases. It is calculated as the ratio of total power consumption (for each method) for these tasks over the total number of such deliveries. The quadratic *HCB-model* formula is used (the results for the *RM-model* were similar).

The power consumption for *GEDIR* algorithm is smaller than the one for *DIR* routing method for small values of square size *m.* The reason is that the smaller hop count is decisive when no retransmission is desirable. However, for larger *m,* compass routing performs better, since the greatest advance is not necessarily best choice, and the closer direction, possibly with smaller advance, is advantageous. The *NC* method is inferior to *GEDIR* or *DIR* for smaller values of *m,* because the greatest possible advance is the better choice for neighbor than the nearest node closer to destination. However, for larger values of *m, NC* outperforms significantly both, since it simulates retransmissions in the best possible way. 2-hop methods failed to produce power savings over corresponding 1-hop methods, and were eliminated in our further investigations.

method/size	10	100	200	500	1000	2000	5000
SP-Power	3577	4356	6772	20256	62972	229455	1404710
SP	3578	4452	7170	25561	92438	358094	2236727
Power	3619	4457	6951	21331	69187	261832	1647964
GEDIR	3619	4460	7076	24823	89120	344792	2152891
DIR	3928	4681	7046	23033	81001	311743	1942952
MFR	3644	4523	7264	25845	93150	361021	2254566
NC	7604	8271	10523	25465	80136	297580	1833993
Random	5962	7099	10626	34382	121002	465574	2896988
2-GEDIR	3587	4452	7148	25399	91570	354980	2216528
2-DIR	3937	4764	7386	25109	89371	344644	2148913
2-MFR	3603	4478	7208	25738	92816	359491	2248876
2-NC	4851	5824	8815	29125	102786	394951	2466065

Table 1. Power consumption of routing algorithms

As expected, the proposed distributed power efficient routing algorithm outperforms all known GPS based algorithms for all ranges of *m.* For small *m,* it is minor improvement over *GEDIR* or *DIR* algorithms. However, for large *m,* the difference becomes very significant, since nearest rather than furthest progress neighbors are preferred. For large *m,* the only competitor is *NC* algorithm. It is also observed that the proposed power efficient algorithm produces paths that are close to the optimal ones, obtained by *SP* (shortest weighted path) algorithm. More precisely, the overhead (percentage of additional energy per routing task) of power efficient algorithm with respect to SP one is 1.2%, 2.3%, 2.6%, 5.3%, 9.9%, 14.1% and 17.3% for the considered values of *m,* respectively. Therefore, localized power efficient routing algorithm, when successful, closely matches the performance of non-localized shortest weighted path algorithm.

8. Performance evaluation of cost and power-cost efficient routing algorithms

The performance evaluation in the previous section shows the superiority of power efficient method for the case of nodes with relatively equal and high battery powers (e.g. all batteries are refreshed). In this section, we shall assume that nodes have different remaining powers, and will involve two more routing methods, cost and power-cost, with their respective shortest weighted path algorithms.

The experiments that evaluate cost and power-cost routing algorithms are designed as follows. Random unit connected graphs are generated as in the previous section. An iteration is a routing task specified by the random choice of source and destination nodes. Any node currently holding the message will halt the routing task if it is a concave node for that task (method failure), or if the current node does not have enough power to transmit the message to the next node, decided by the given method (power failure). Instead of running fixed number of iterations, we decided to run iterations until the first power failure at a node occurs (at which point the corresponding method 'dies'). Each node is initially assigned an energy level at random in the interval [*minpow, maxpow*], where parameters are chosen differently for different values of *m*. After sending a message from node *A* to node *B,* the energy that remained at *A* (*B*) is reduced by the power needed to transmit (receive) the message, respectively. The experiment is performed on 20 graphs for each method, for each of *HCB-* and *RM-model* formulas.

Because of experience with success rate of power efficient method, we considered two variants of cost and power-cost algorithms, and in the main method we restrict the selection of neighbors to the nodes that are closer to destination than current node. The success rates for unrestricted versions (all neighbors considered) was again low in our experiments. For example, the success rate of *cost0* method drops from 64% to 55% with increasing *m,* while *power-cost0* method drops from 77% to 14% (data for other variants are similar; *HCB-model* is again used, while the other model had very similar data). Consequently, these methods were deemed not viable (the number of iterations before dying for them is not shown, because high failure rate certainly distorted their interpretation). The success rate for restricted versions (only closer neighbors considered) was in the range 92%-95% for all cost and power-cost methods discussed here, both models, and all sizes *m*. The number of iterations before each method dies, for *HCB-model*, is given in Table 2. *RM-method* gave similar results. The cost and power-cost methods from Table 2 are described in section 5.

method/trial count	10	100	200	500	1000	2000	5000
SP	289	713	1412	668	647	454	275
SP-Power	342	865	1710	983	1114	796	482
SP-Cost	674	1703	3540	1686	1590	1066	646
SP-PowerCost	674	1697	3530	1776	1838	1230	728
SPPowerCost1	647	1668	3495	1725	1688	1124	682
Power	379	954	1843	1009	1162	789	469
Cost0	624	1630	3255	1594	1479	988	601
Cost1	637	1616	3304	1651	1494	991	602
PowerCostP	671	1616	3127	1522	1522	1053	600
PowerCost0	662	1609	3118	1513	1528	1056	617
PowerCost1	660	1611	3180	1664	1757	1179	712
PowerCost2	631	1537	3211	1676	1716	1152	686
1-GEDIR	373	941	1814	832	849	548	318
1-DIR	345	921	1741	831	902	603	355
1-MFR	375	909	1775	800	797	525	316
1-NC	204	551	1268	809	931	668	414
Random	201	481	889	546	512	312	202

Table 2. Number of iterations before one of node in each method dies

The intervals *[minpow, maxpow]* were set as follows: [80K, 90K] for *m*=10, [200K, 300K] *m*=100, [500K, 1M] for *m*=200, [750K, 1.5M] for *m*=500, [3M, 4M] for *m*=1000, [8M, 10M] for *m*=2000, [30M, 40M] for *m*=5000, where K=1000 and M=1000000.

Our experiments confirmed the expectations on producing power savings in the network and/or extending nodes lifetime by applying our algorithms on randomly generated unit graphs. Both cost methods and all four power-cost methods gave very close trial numbers, and thus it is not possible to choose the best method based on trial number alone. However, all proposed cost and power-cost methods, which are localized, performed equally well as the corresponding non-localized shortest path cost and power-cost algorithms (the number of trials is sometimes even higher, due to occasional delivery failures which save power). It is also clear that cost and power-cost routing algorithms last for significantly more iterations than the power algorithm.

method/power	10	100	200	500	1000	2000	5000
SP-Cost	44381	133245	395592	618640	1857188	4819903	19238265
SP-PowerCost	44437	133591	396031	642748	2067025	5686092	23187052
SPPowerCost1	46338	136490	406887	646583	1972185	5252813	21081420
Cost0	43996	129608	410610	656349	2053190	5370162	21338314
Cost1	39831	120785	377549	619221	2022771	5335936	21233992
PowerCostP	30561	127819	421927	712958	2299590	6058424	24782129
PowerCost0	27434	126066	416889	712033	2286840	6030614	24419832
PowerCost1	27520	126201	409208	666907	2091211	5658144	22622947
PowerCost2	33563	131804	401174	652199	2078140	5684752	23136193

Table 3. Average remaining power level at each node

Table 3 shows the average remaining power at each node after the network dies, for the most competitive methods. Cost methods have more remaining power only for the smallest size *m*=10, when the power formula reduces to the constant function. For larger sizes of *m*, two better power-cost formulas leave about 15% more power at nodes than the cost method. Thus, power-cost method may outperform cost methods, since the network will continue to operate without the first node with power failure.

It is also of interest to check the average hop counts for the cost and power-cost routing algorithms. *SP-cost, cost0* and *cost1* methods have hop counts approximately 4.0, 4.5, and 4.9 for *HCB-model* and all values of *m.* Four power-cost methods have similar hop counts, 5.8, 4.7, 5.0, 6.7, 8.4, 9.1 and 9.6, respectively, for sizes of *m.* Two SP-power-cost methods do not similar hop counts. *SP-PowerCost* method has hop counts 4.0, 4.1, 4.3, 6.3, 7.8, 8.3, 8.7, while *SP-PowerCost1* method has hop counts between 4.0 and 4.6.

Conclusion

This paper described several localized routing algorithms that try to minimize the total energy per packet and/or lifetime of each node. The proposed routing algorithms are all demand-based and can be augmented with some of the proactive or reactive methods reported in literature, to produce the actual protocol. These methods use control messages to update positions of all nodes to maintain efficiency of routing algorithms. However, these control messages also consume power,

and the best trade-off for moving nodes is to be established. Therefore further research is needed to select the best protocols. Our primary interest in this paper was to examine power consumption in case of static networks and provide basis for further study. Our method was tested only on networks with high connectivity, and their performance on lower degree networks remains to be investigated. Based on experience with basic methods like GEDIR [SL], improvements in the power routing scheme to increase delivery rates, or even to guaranty delivery [BMSU, S] are necessary before experiments with moving nodes are justified. Power efficient methods tend to select well positioned neighboring nodes in forwarding the message, while cost efficient method favor nodes with more remaining power. The node movement, in this respect, will certainly assist power aspect of the formula since the movement will cause the change in relative node positioning. This will further emphasize the advantage of power-cost over power only or cost only methods.

The formulas for power, cost, and power-cost methods may also need some improvements, and our experiments do not give an ultimate answer on even the selection of approach that would give the most prolonged life to each node in the network. The presented distributed routing algorithms may be improved by multiplying the optimal power (or power-cost) for the remaining transmissions by a factor that will depend on the network conditions. Our energy consumption formulas are easily adjustable to different laws for received power in terms of transmitted power and distance. Also, the corresponding distributed algorithms may be improved by studying different ways of selecting neighbors (that is particularly the case in the power-cost routing algorithm for improving the judgement on whether to send packet directly to destination from the node currently holding it).

It is also interesting problem to consider power efficient broadcasting, in which a node wants to send a message to all other nodes in MANET. Previous studies concentrated on the time needed for broadcasting or number of retransmissions.

Finally, the reduction of CPU power, in addition to optimizing transmission power, deserves further investigation. For instance, the Dijkstra's shortest weighted path algorithm runs in $O(n^2)$ time (where n is the number of nodes in MANET), which can be improved to $O(n \log n)$ time using complicated data structures (therefore having high constants and potentially higher time complexity for smaller networks). Is linear time algorithm for computing optimal transmission power routing path possible? Galtier [G] argues that the CPU energy consumption is proportional to the square of the execution time, and therefore significant saving can be achieved by using parallel processing. The conversion of sequential to parallel distributed routing algorithms is straightforward, although addition and deletion of neighbors requires careful design. The design of a parallel shortest weighted path routing algorithm is also justified.

References

[BCSW] S. Basagni, I. Chlamtac, V.R. Syrotiuk, B.A. Woodward, A distance routing effect algorithm for mobility (DREAM), Proc. MOBICOM, 1998, 76-84.

[BMJHJ] J. Broch, D.A. Maltz, D.B. Johnson, Y.C. Hu, J. Jetcheva, A performance comparison of multi-hop wireless ad hoc network routing protocols, Proc. MOBICOM, 1998, 85-97.

[BMSU] P. Bose, P. Morin, I. Stojmenovic and J. Urrutia, Routing with guaranteed delivery in ad hoc wireless networks, 3rd Int. Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications, Seattle, August 20, 1999, 48-55.

[CC] C.J. Chang and J.F. Chang, Optimal design parameters in a multihop packet radio networks using random access techniques, in Proc. IEEE GLOBECOM, 1984, 493-497.

[CN] S. Chen and K. Nahrstedt, Distributed quality-of-service routing in ad hoc networks, IEEE J. Selected Areas in Communications, 17, 8, August 1999, 1488-1505.

[CZ] A. Chockalingam and M. Zorzi, Energy consumption performance of a class of access protocols for mobile data networks, Proc. Vehicular Technology Conference, Ottawa, Ontario, Canada, 1998.

[E] M. Ettus, System capacity, latency, and power consumption in multihop-routed SS-CDMA wireless networks, Proc. IEEE Radio and Wireless Conf., Colorado Springs, Aug. 1998.

[EGHK] D. Estrin, R. Govindan, J. Heidemann, S. Kumar, Next century challenges: Scalable coordination in sensor networks, Proc. MOBICOM, 1999, Seattle, 263-270.

[G] J. Galtier, Using parallel computing to reduce CPU power, Workshop on optimization methods for wireless computing, 32-34, Montreal, Canada, December 1998.

[GFM] J.J. Garcia-Luna-Aceves, C.L. Fullmer and E. Madruga, Wireless mobile internetworking, manuscript.

[GK] P. Gupta and P.R. Kumar, Critical power for asymptotic connectivity in wireless networks, in: Stochastic Analysis, Control, Optimization and Applications (W.M. McEneany, G. Yin, Q. Zhang, eds.), Birkhauser, Boston, 1998, 547-566.

[h] http://www.networks.digital.com/npb/html

[HCB] W.R. Heinzelman, A. Chandrakasan and H. Balakrishnan, Energy-efficient routing protocols for wireless microsensor networks, Proc. Hawaii Int. Conf. on System Sciences, Jan. 2000, to appear.

[HKB] W.R. Heinzelman, J. Kulik, H. Balakrishnan, Adaptive protocols for information dissemination in wireless sensor networks, Proc. MOBICOM, Seattle, 1999, 174-185.

[HL] T.C. Hou and V.O.K. Li, Transmission range control in multihop packet radio networks, IEEE Transactions on Communications, 34, 1, 1986, 38-44.

[IETF] IETF internet draft, http://www.ietf.org/html.charters/manet-charter.html, 1998.

[K] E.D. Kaplan (ed.) Understanding GPS: Principles and Applications, Artech House, Boston, MA, 1996.

[KKKP] L.M. Kirousis, E. Kranakis, D. Krizanc, A. Pelc, Power consumption in packet radio networks, STACS, Lecture Notes in Computer Science Vol. 1200, 1997, 363-374.

[KKP] J.M. Kahn, R.H. Katz, K.S.J. Pister, Next century challenges: Mobile networking for 'smart dust', Proc. MOBICOM, 1999, Seattle, 271-278.

[KSU] E. Kranakis, H. Singh and J. Urrutia, Compass routing on geometric networks, Proc. 11th Canadian Conference on Computational Geometry, Vancouver, August, 1999.

[KV] Y.B. Ko and N.H. Vaidya, Location-aided routing (LAR) in mobile ad hoc networks, Proc. MOBICOM, 1998, 66-75.

[LL] C. R. Lin and J.S. Liu, QoS routing in ad hoc wireless networks, IEEE J. Selected Areas in Communications, 17, 8, Aug. 1999, 1426-1438.

[MC] J.P. Macker and M.S. Corson, Mobile ad hoc networking and the IETF, Mobile Computing and Communications Review, 2, 1, 1998, 9-14.

[MG] W. Mangione-Smith and P.S. Ghang, A low power medium access control protocol for portable multi-media systems, 3rd int. Workshop on Mobile Multimedia Communications, Sept. 1996.

[N] NAVSTAR GPS operations, http://tycho.usno.navy.mil/gpsinfo.html, and Iowa State University GPS page, http://www.cnde.iastate.edu/gps.html .

[NI] J.C. Navas and T. Imielinski, GeoCast - geographic addressing and routing, Proc. MOBICOM, 1997, 66-76.

[NK] R. Nelson and L. Kleinrock, Tha spatial capacity of a slotted ALOHA multihop packet radio network with capture, IEEE Transactions on Communications, 32, 6, 1984, 684-694.

[PL] K. Pahlavan and A. Levesque, Wireless information networks, Wiley Interscience, 1995.

[R] T.S. Rappaport, Wireless communications: Principles and Practice, Prentice Hall, 1996.

[RB] J.M. Rulnick and N. Bambos, Mobile power management for wireless communication networks, Wireless Networks, 3, 1997, 3-14.

[RM] V. Rodoplu and T.H. Meng, Minimum energy mobile wireless networks, IEEE Journal on Selected Areas in Communications, Vol. 17, No. 8, August 1999, 1333-1344.

[RS] S. Ramanathan and M. Steenstrup, A survey of routing techniques for mobile communication networks, Mobile Networks and Applications, 1, 2, 1996, 89-104.

[RT] E.M. Royer and C.K. Toh, A review of current routing protocols for ad hoc mobile wireless networks, IEEE Personal Communications, April 1999, 46-55.

[SC] A.K. Salkintzis and C. Chamzas, An in-band power-saving protocol for mobile data networks, IEEE Transactions on Communications, 46, 9, 1998, 1194-1205.

[SR] S. Singh and C.S. Raghavendra, PAMAS – Power aware multi-access protocol with signaling for ad hoc networks, ACM Computer Communications Review, July 1998, 5-26.

[SSL] A.R. Shahani, D.K. Schaeffer, and T.H. Lee, A 12mW wide dynamic range CMOS front-end for a portable GPS receiver, Proc. IEEE Int. Solid-State Circuits Conf., vol. 40, Feb. 1997, 368-369.

[SWR] S. Singh, M. Woo, C.S. Raghavendra, Power-aware routing in mobile ad hoc networks, Proc. MOBICOM, 1998, 181-190.

[S] I. Stojmenovic, Power aware routing algorithms with guaranteed delivery in wireless networks.

[SL] I. Stojmenovic and X. Lin, GEDIR: Loop-free location based routing in wireless networks, IASTED Int. Conf. on Parallel and Distributed Computing and Systems, Nov. 3-6, 1999, Boston, MA, USA, 1025-1028.

[SV] I. Stojmenovic and B. Vukojevic, A routing strategy and quorum based location update scheme for ad hoc wireless networks, Computer Science, SITE, University of Ottawa, TR-99-09, September 1999.

[TK] H. Takagi and L. Kleinrock, Optimal transmission ranges for randomly distributed packet radio terminals, IEEE Transactions on Communications, 32, 3, 1984, 246-257.