Computer Networks 56 (2012) 1951–1967

Contents lists available at SciVerse ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Design principles and improvement of cost function based energy aware routing algorithms for wireless sensor networks

Anfeng Liu [a,b,*], Ju Ren [a], Xu Li [c], Zhigang Chen [a], Xuemin (Sherman) Shen [b]

[a] College of Information Science and Engineering, Central South University, Changsha 410083, China
[b] Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1
[c] INRIA Lille – Nord Europe, Univ. Lille Nord de France, USTL, CNRS, UMR 8022, LIFL, France

ARTICLE INFO

Article history:
Received 24 October 2011
Accepted 5 January 2012
Available online 21 February 2012

Keywords:
Wireless sensor networks
Energy hole avoidance
Adaptive routing
Network lifetime

ABSTRACT

Cost function based routing has been widely studied in wireless sensor networks for energy efficiency improvement and network lifetime elongation. However, due to the complexity of the problem, existing solutions have various limitations. In this paper, we analyze the inherent factors, design principles and evaluation methods for cost function based routing algorithms. Two energy aware cost based routing algorithms named Exponential and Sine Cost Function based Route (ESCFR) and Double Cost Function based Route (DCFR) have been proposed in this paper. For ESCFR, its cost function can map small changes in nodal remaining energy to large changes in the function value. For DCFR, its cost function takes into consideration the end-to-end energy consumption, nodal remaining energy, resulting in a more balanced and efficient energy usage among nodes. The performance of the cost function design is analyzed. Extensive simulations demonstrate the proposed algorithms have significantly better performance than existing competing algorithms.

1. Introduction

Wireless sensor networks (WSNs) are collections of low-cost battery-powered devices, called sensors, which have integrated sensing, computing, and wireless communication capabilities [1]. They are deployed for detecting events of a predetermined nature and transmitting sensed event data to the data sink or base station for further analysis [2,3]. It is recognized that WSNs have great potentials in many important applications such as military surveillance, environmental monitoring, infrastructure and facility diagnosis, and so on [4]. To reduce deployment budge, WSNs are expected to have minimized overall energy consumption and balanced energy usage among individual sensors. In WSNs, one of the main design challenges is to maximize network lifetime without scarifying network sensing performances (e.g., coverage and reliability).

The lifetime of a WSN can be defined as the time elapsed till the first sensor node in the network depletes its energy, since once a sensor node dies, the sensing capability of the network starts degrading [3]. To maximize network lifetime, an energy-efficient routing algorithm should be used for data communications. The algorithm needs to have the following three main features: (1) minimum total energy usage, (2) balanced energy consumption, and (3) distributed characteristics.

Cost function based routing has been studied extensively because of its distributed nature and good energy performance [5,6]. In such routing algorithms, a node currently having a packet to transmit decides locally which of its neighbors is the next hop based on a cost function. A well-designed cost function will lead to energy-efficient decisions and prolonged network lifetime. There are many cost functions proposed in literature. They were, however, designed merely according to designers' experience, which is suboptimal, and lacks of theoretical analysis on their performance.

* Corresponding author at: College of Information Science and Engineering, Central South University, Changsha 410083, China.
E-mail address: csuanfengliu@gmail.com (A. Liu).

1389-1286/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2012.01.023

In this paper, we analytically study existing cost function based routing algorithms and present the general principles and guidelines for cost functions construction. We propose a novel double cost function based routing (DCFR) algorithm, which is decentralized, adaptive, and outperforms existing cost function based solutions in terms of energy efficiency improvement and network lifetime elongation. Existing solutions consider only end-to-end energy consumption and nodal remaining energy, and achieve suboptimally balanced energy consumption. Unlike these schemes, DCFR additionally includes energy consumption rate in its cost function. The cost function has a rapidly increasing slope such that a small difference in energy consumption rate or available energy level can lead to a big difference in function values. Hence, DCFR has an excellent capability of balancing energy usage during routing. Each node chooses next hop to forward data according to the energy consumption rates as well as node remaining energy of its neighbors, so energy consumption is balanced in the entire network. We show the benefits of our proposed cost function design guideline. We evaluate the performance of our new routing algorithm DCFR through extensive simulation using various performance metrics. We compare DCFR with three well-known routing algorithms, i.e., DC (Direct Communication) [7,8], minimum transmission energy (MTE) [9], and distributed energy balanced routing (DEBR) [5]. Our simulation confirms that DCFR indeed has significantly better performance than these existing algorithms in network lifetime elongation and energy balancing.

2. Related work

To prolong the lifetime of a WSN, a number of routing algorithms have been proposed. They mainly aim to minimize total energy usage in the network. However, sensors along paths with minimized total energy cost are repeatedly used and deplete their energy quickly, resulting in short network lifetime. Thus, researchers found that routing algorithms should consider not only total energy consumption, but also the amount of remaining energy in each sensor. By giving preference to sensors with high remaining energy during route selection, per node energy usage peaks are flattened, and network lifetime is improved. Existing routing algorithms can be divided into distributed routing algorithms and centralized routing algorithms. In the distributed routing algorithms, the routing path selection depends only on local information, thus this kind of algorithms have good scalability for large-scale networks. Instead, the centralized routing algorithms generally need global information to assist route selection. This type of algorithms may find optimal routing paths, but they generate extra communication overhead and have poor scalability. Related research about these two types of routing algorithms is summarized as follows:

2.1. Centralized routing algorithms

In centralized routing algorithms, the network topology and energy consumption are known as a prior to sensor nodes. Wireless networks are modeled as graphs, in which, a vertex represents a wireless device and an edge between two vertices indicates that they are in direct communication range. The weight on a vertex indicates the node remaining energy and the weight on an edge (u, v) represents the energy consumption for node u (resp. v) to transmit one unit data to node v (resp. u). Therefore, graph theory is used in centralized routing algorithm to find paths which consume minimum energy (metrics 1) and paths which avoid energy hotspots nodes (metrics 2). In early research, Ettusand et al. proposed the so-called minimum transmission energy (MTE) routing scheme [9], which selects the route that uses minimum energy to transport one packet from the source to the destination. However, its network lifetime is not good due to the early death of energy efficient route. Zytoune et al. proposed an uniform balancing energy routing protocol (UBERP) [10]. In this protocol, the path selection for multi-hop transmission is done using the minimum energy transmission over network nodes which have a remaining energy greater than a threshold. The target is to balance the transmission energy consumption over the entire network so as to avoid energy depletion of the shortest routes nodes, therefore, this threshold should equal to the average network energy. UBERP protocol actually reflects the main idea of most centralized routing algorithm, i.e., nodes with lower remaining energy and edges with lower energy efficiency are excluded in the candidate set, and then use the Dijkstra's algorithm to calculate the optimal routing. However, global information is needed to exclude such nodes and edges, thus the system cost is increased and its extensibility is limited. Moreover, it is required to re-calculate the graph to make routing decisions. The complexity of the algorithm is $O((m + n)\log n)$.

Li et al. described the max–min zPmin algorithm [11]. The max–min zPmin algorithm attempts to balance metrics 1 and 2 by first calculating a path based on the remaining energy levels, and then rejecting any path whose total energy is z times of the minimum energy path. The quality of the solution provided by the max–min zPmin algorithm depends on the empirically generated parameter z, and this does not always provide an optimal solution. Chang and Tassiulas combined metrics 1 and 2 into a single metric and run Dijkstra's algorithm on this new metric [12]. This method does not actually optimize any metric and the performance of the approach closely depends on the empirical values assigned to the parameters.

Park and Sahni presented the online maximum lifetime (OML) heuristic [13], which is an enhancement of the CMAX algorithm presented by Kar et al. [14]. OML initially removes edges with low remaining energy from the graph. Then the edge weight is modified so that a high cost (and thus a heavy penalty) is associated with edges having low remaining energy or high communication cost. Dijkstra's algorithm runs on the modified graph so that the selected paths always use nodes with high energy levels and edges with low energy costs. They reported that OML gives the best network lifetime among all routing approaches in the current literature. Mohanoor et al. presented three polynomial time combinatorial techniques to provide a good tradeoff between metrics 1 and 2 [15]. The first technique, called the shortest widest path, first maximizes the concave metric (the remaining energy of a path) and then minimizes the additive metric (energy consumed along a path). The second technique, called the shortest width constrained path, finds paths with a suitably high remaining energy (may not be the maximum), and then minimizes the total energy consumed along such a path. Lastly, the third approach (shortest fixed width path) is similar to the second approach in the sense that it finds a minimum energy path among the paths that have a high remaining energy. However, it does not change the remaining energy in each route calculation and the remaining energy level is changed only when it becomes infeasible to find paths between the source and the destination at the current remaining energy level. They claimed that the proposed distributed techniques outperform the algorithm proposed by Park and Sahni [13].

2.2. Distributed routing algorithms

Distributed routing algorithms do not rely on any global information. In such algorithms, each node makes independent routing decisions merely based on its local knowledge. Thus, the system communication cost is greatly reduced, and the computing complexity is reduced from *nlogn* to a constant *k* so it is more promising. Distributed routing algorithms usually calculate the communication cost of each neighbor node using a predefined cost function and choose the node with the smallest cost as next hop.

Chang et al. presented an interesting approach named maximum lifetime energy routing [12]. The key idea of this approach is to maximize the network lifetime by defining link cost as a function of node remaining energy and the required transmission energy. The authors give two maximum remaining energy path algorithms. The two algorithms differ in the defining method of link costs and the incorporation of nodes remaining energy. Instead of using the consumed energy e_{ij}, when a packet is transmitted between node i and node j, the following link costs are used:

$$C_{ij} = \frac{1}{E_i - e_{ij}}, \quad C_{ij} = \frac{e_{ij}}{E_i} \tag{1}$$

where E_i is the remaining energy at node i. Using those algorithms, the average network lifetime can be extended in the best case, by a factor of 3.26 compared to the MTE [9]. DEBR algorithm [5] extended the above algorithm, DEBR defines the energy cost EC_{ij} for a transmission from node i and node j as

$$EC_{ij} = \frac{Required\ energy\ from\ node\ i\ to\ j}{Available\ energy\ at\ node\ i} = \frac{e_{ij}}{E_i^r} \tag{2}$$

The total energy cost (TEC_{ik}) of a neighboring node k at sensor i is the sum of the energy costs from node i to k and from node k to the base station

$$TEC_{ik} = EC_{ik} + EC_{k,BS} \tag{3}$$

Based on this definition, sensor i can select the best candidate, node K

$$K = Arg \min_{j \in N_i + \{i\}} (TEC_{ij}) \tag{4}$$

Compared with Ref. [12], this algorithm takes the energy cost of the entire routing path into consideration. They claimed this algorithm can establish energy sufficiency as well as efficiency. Since the cost function EC_{ij} is transmission energy cost relative to available energy, its value is low in the case that the required transmission energy is low and available energy is high. Rahme et al. proposed a representative cost function [6], which includes four cost functions. The main difference between these four cost functions and previous research is that they take into consideration the current node energy consumption, the remaining energy of nodes within one hop and two hops, and the energy consumption due to signal interference within two hops. In addition, Ok et al. also proposed a routing algorithm named MaxEW [16]. It adopts the social welfare function from social sciences to compute energy welfare as a measure for energy populations. When each sensor tries to maximize energy welfare of its local society, it collectively leads to globally efficient energy-balancing.

From the above discussion, nodes can only make routing decisions based on the information of itself and its neighbors, leading to only a few limited parameters to form the cost function. This paper attempts to give the general principles of cost function construction, and proposes an energy aware routing strategy based on the cost function, with better performance to improve the network lifetime.

3. Sensor network model

3.1. Network topology

We adopt the same network model as [5,16]. The network is composed of n homogenous sensors randomly and uniformly distributed over a target area. Events occur uniformly such that every sensor has one packet to report periodically. The neighboring distance is defined as the maximal reachable distance of radio frequency with the maximum transmission power. Each sensor can be aware of the current energy level of its neighbors and the energy required to transmit data from each of its neighboring sensors to the base station [17]. We assume a perfect transmission model, i.e., a sensor's neighbors can receive all the messages that the sensor transmits. When a sensor transmits a message to one of its neighbors or the base station, the sensor attaches the information of its remaining energy to the message so that all of the neighbors can update its energy level. This updating process guarantees that information of neighboring sensors for the routing decision is available for sensors. The lifetime of the network is the time elapsed till the first sensor node in the network uses up its energy. The goal is to maximize the network lifetime by designing an energy-efficient routing algorithm.

3.2. Energy consumption model

Sensors consume energy when they are sensing, receiving and transmitting [18]. The amount of energy consumed for sensing is not related to routing [19]. Therefore, we

Table 1
Network parameters.

Parameter	Value
Threshold distance (d_0) (m)	87
Sensing range r_s (m)	15
E_{elec} (nJ/bit)	50
e_{fs} (pJ/bit/m^2)	10
e_{amp} (pJ/bit/m^4)	0.0013
Initial energy (J)	0.5

consider only the energy usage for transmitting and receiving messages. According to the radio model used in [7], energy consumption for transmitting is given by Eq. (5)

$$\begin{cases} E_{member} = lE_{elec} + l\varepsilon_{fs}d^2 & \text{if } d < d_0 \\ E_{member} = lE_{elec} + l\varepsilon_{amp}d^4 & \text{if } d > d_0 \end{cases} \tag{5}$$

where E_{elec} is transmitting circuit loss. Both the free space (d^2 power loss) and the multi-path fading (d^4 power loss) channel models are considered in the model, depending on the distance between transmitter and receiver. ε_{fs} and ε_{amp} are the energy required by power amplification in these two models, respectively. The energy spent for receiving a l-bit packet is

$$E_R(l) = lE_{ele} \tag{6}$$

The above parameter settings are given in Table 1 [7].

4. Analysis and design principles of cost function

4.1. A case study of existing cost function based routing

We use an example to illustrate that existing cost function based routing strategies can be farther improved. In Fig. 1, there are two obstacles in a rectangular WSN. Assume that each sensor initially has the same amount of energy enough for transmitting and receiving $m = 10{,}000$ data packets. The area covered by the network is $s = L \times W(115 \times 75)\,\mathrm{m}^2$; node density is $\rho = 0.06/\mathrm{m}^2$; no sensor is deployed in the obstacle areas. There are $n = 400$ sensors in the network. The number of sensors in the red area is $n_1 = 280$. In each round of data collection, each sensor generates one data packet.

We first look at the minimum transmission energy (MTE) routing algorithm proposed by Ettus and Shepard [9,20]. This algorithm selects the route that uses the least amount of energy to transport one packet from source to destination. Data in the red area can be sent to the sink via Route-1 and Route-2. Route-1 consumes less energy than Route-2 since it is shorter, and therefore is selected by the MTE algorithm. The path segment $B - B'$ in Route-1 can also be used by many other routes, thus $B - B'$ is called hotspots, which determines the lifetime of the entire network. Assume that there are $k = 3$ nodes in the hotspots, they have to forward data from $n_1 = 280$ nodes in the red region to the sink. The network lifetime by MTE can be calculated as

$$l_1 = \frac{m}{n_1/k} = \frac{km}{n_1} = \frac{10000}{280/3} = 107 \quad (\text{rounds}).$$

We examine the distributed energy balanced routing (DEBR) algorithm proposed by Chang recently [5]. Eqs. (2)–(4) are used in DEBR. In Fig. 1, assume that the number of hops in the hotspot part $B - B'$ in Route-1 is $\frac{1}{x}$ of that in the $A - A'$ portion in Route-2. We have results about the energy consumption and network lifetime in Lemma 1.

Lemma 1. *In Fig. 1, when the TEC (see the definition in Eq. (3)) of Route-1 equals to the TEC of Route-2, i.e., when the route is balanced (i.e., the probabilities that nodes transmit data via different routes are the same), the remaining energy of Route-1 in the hotspot is just $\frac{1}{x}$ of that of Route-2.*

Proof. Let

$$TEC_{route-1} = \sum_{i,j \in route-1} EC_{ij}, TEC_{route-2} = \sum_{i,j \in route-2} EC_{ij}.$$

When the network begins to operate, nodes have the same amount of initial energy. As Route-1 is shorter than Route-2, we have $TEC_{route-1} < TEC_{route-2}$. Therefore, data in the red zone is routed through Route-1, and $TEC_{route-1}$ increases rapidly. This results in $TEC_{route-1} = TEC_{route-2}$ after a period of time. We have the following analysis about the energy consumption of Route-1 and Route-2.

Route-1 and Route-2 can be divided in the following way. Route-1: $S \to B \to B' \to$ Sink; Route-2: $S \to A \to A' \to$ Sink. We can consider that traffic load is basically the same for all sensors. So the difference of TEC between Route-1 and Route-2 is due to the different energy consumptions in hotspot $B \to B'$ and $A \to A'$. The number of hops in $B \to B'$ is n, and that in $A \to A'$ is $x \times n$. To make $TEC_{route-1} = TEC_{route-2}$, we should have

$$\sum_{i,j \in B \to B'} EC_{ij} = \sum_{i,j \in A \to A'} EC_{ij} \tag{7}$$

In $EC_{ij} = e_{ij}/E_i^r$, e_{ij} is the energy consumed for transmitting one packet, which can be considered the same for both routes. For simplicity, let us assume that the remaining energy E_i^r is the same for nodes in $B \to B'$ and nodes in $A \to A'$. Eq. (7) can be transformed as follows

$$n\frac{e_{ij}}{E^r_{B-B'}} = xn * \frac{e_{ij}}{E^r_{A-A'}} \Rightarrow E^r_{B-B'} = \frac{1}{x}E^r_{A-A'} \qquad \square$$

From Lemma 1, in DEBR, the network remaining energy is not balanced in different routes. For example, only when the remaining energy of $B - B'$ in Route-1 is $1/x$ of Route-2,

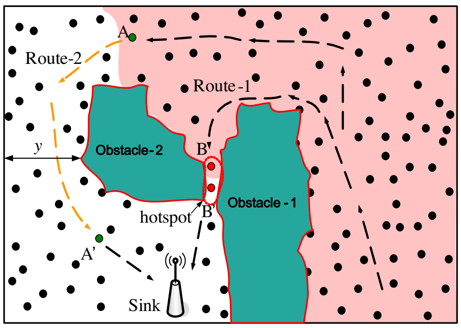


Fig. 1. Network topology 1.

the data in red area can be routed through Route-2. Even worse, the energy consumption is more imbalanced if the path length of Route-2 is longer than its currently path length. In Fig. 1, there may be *y-1* paths longer than Route-2 in the left side of Route-2 (i.e., the size of obstacles-2 in Fig. 1 and size of the narrowest region in network on the left are *y* times of the hotspot). We have the following Lemma 2.

Lemma 2. *The ratio of DEBR network lifetime to the optimal network lifetime is* $\varphi_d = 1 - \frac{(x-1)y}{l_o}$.

Proof. When hotspot nodes can transmit data for one round of collection in the red area, Route-2 can transmit $(x-1)$ rounds. According to the routing principles of DEBR, Route-1 will be chosen. Then nodes in the hotspots area die fast. At the time of their death, Route-2 can still transmit for $(x-1)$ rounds, and there are *y* paths similar to Route-2. Therefore, after the death of nodes in hotspots, the network can carry out more than $(x-1)y$ rounds of data transmission before the entire network dies. In other words, the network using DEBR dies $(x-1)y$ rounds earlier than that of the optimal algorithm. Denote the optimal lifetime of the network by l_0. The network life using DEBR algorithm is $l_d = l_o - (x-1)y$. We have

$$\varphi_d = \frac{l_o - (x-1)y}{l_o} = 1 - \frac{(x-1)y}{l_o} \qquad \square$$

If energy consumption can be balanced, data collection will be performed along both route-1 and route-2. Then, there are *y + 1* paths that can be used to transmit data from 320 nodes (see Fig. 1). The optimal network lifetime can be estimated as

$$l_o = \frac{10000}{320/30} = 937 \quad \text{(rounds)}.$$

In Fig. 1, let $x = 12$ and $y = 9$, then the network lifetime by DEBR is $l_d = 838$. Fig. 2 shows the performance comparison of network lifetime of these algorithms. We can see that the shortest path algorithm has the minimum lifetime, and DEBR can be improved by 11% compared to the optimal algorithm. Meanwhile, the larger *x* and *y*, the farther the lifetime is from the optimal lifetime.

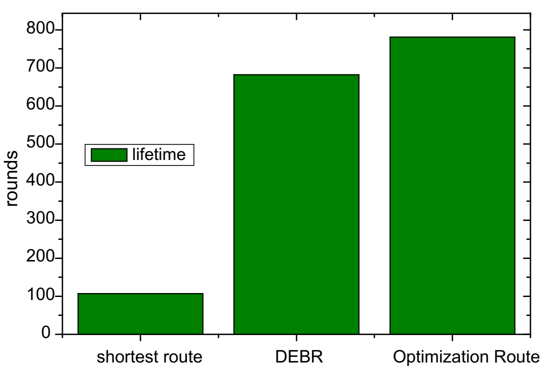


Fig. 2. The lifetime under different routing algorithms.

4.2. Design principles of cost function

Most of existing cost function based routing algorithms are designed according to designer's experience. It is difficult to justify the rationality of their introduced cost functions. Below we present an analysis of a few existing routing algorithms and derive several cost function design principles.

In general, the cost function can be represented as $f(x_i) = k_i/x_i$, where x_i is the nodal remaining energy or a constant such as 1, k_i a constant such as 1 or e_{ij} (energy consumption for sending unit data from *i* to *j*). The total energy cost is calculated as $TEC = \sum_{i \in path} f(x_i)$. We analyze two typical cost functions, one is $f(A) = 1/E_i^r$, where E_i^r is the nodal remaining energy (i.e., $k_i = 1,\ x_i = E_i^r$), the other is $f(B) = e_{ij}$ (i.e., $k_i = e_{ij}$, $x_i = 1$). The state of the network can be divided into two stages.

4.2.1. Expansion stage of non-balanced energy consumption

At this stage, the nodal remaining energy between Route-1 and Route-2 is increasingly different. Figs. 3 and 4 show the total energy cost (TEC) and remaining energy of Route-1 and Route-2 under cost functions *f*(*A*) and *f*(*B*). The evolution of the network state under *f*(*A*) is as follows. At the beginning, since Route-1 is much shorter than Route-2, its TEC is low, and data in the red area are routed through Route-1. Thus, its energy consumption rate is very high. Fig. 4 shows that the remaining energy of Route-1 decreases quickly. Since few data are routed through Route-2, the energy consumption of Route-2 is low (the remaining energy of Route-2 goes down slowly, as shown in Fig. 4). However, since the remaining energy of Route-1 decreases very fast, its TEC increases fast; whereas, the remaining energy of Route-2 changes slightly, and its TEC increases slowly. When the network passes 102 data collection rounds, it reaches a turning point *A* where $TEC_{route-1} = TEC_{route-2}$. At this point, the gap between the remaining energy of Route-1 and Route-2 is the largest. Under the effect of the cost function *f*(*B*), TEC gap between Route-1 and

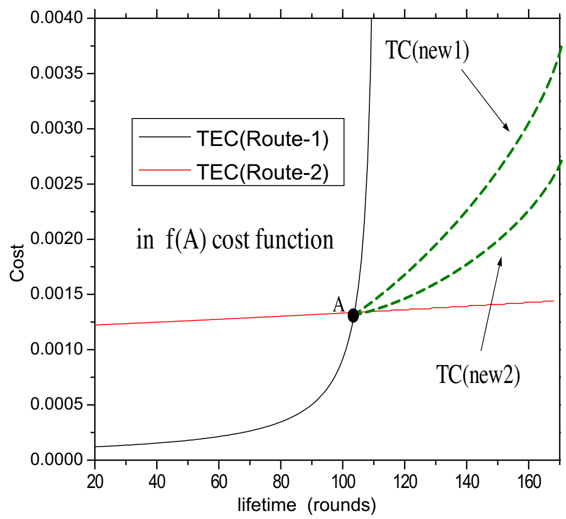


Fig. 3. The total cost of routes 1 and 2.

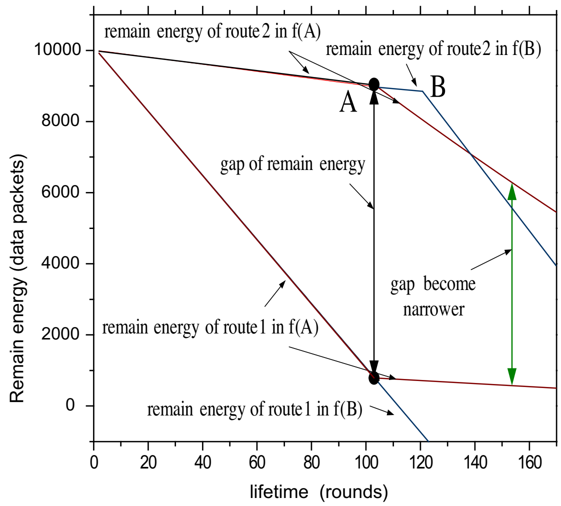


Fig. 4. The remaining energy of routes 1 and 2.

Route-2 exists when the algorithm starts, and it remains until nodes in Route-1 die. After that, it comes to a turning point *B*, from which the remaining energy gap keeps growing.

4.2.2. Energy consumption gap reduction

At this stage, routing paths with different distances to the sink have relatively stable cost difference, and the remaining energy gap no longer increases, as shown in Figs. 3 and 4. Under cost function *f*(*A*), when the network reaches a turning point *A* where $TEC_{route-1} = TEC_{route-2}$, the data in the red region are also routed through Route-2, which makes the TEC curve of Route-2 steeper and the TEC curve of Route-1 slower. Equilibrium is eventually reached. Thus, after the turning point *A*, $TEC_{route-1}$ and $TEC_{route-2}$ converge to the same point. Which of TC (new1) and TC (new2) is the total cost depends on the parameter *y*, as depicted in Fig. 3. A large *y* means that there are many possible routing paths available for selection, leading to a moderate TEC (thus TC (new2) in Fig. 3). A small *y* will on the contrary lead to a steeper TEC curve (i.e., TC (new1) in Fig. 3). The remaining energy at this stage is shown in Fig. 4. After the turning point *A*, the energy consumption rate of Route-1 slows down (the remaining energy of Route-1 curve is flatter), and the energy consumption rate of Route-2 increases fast (the remaining energy of Route-2 curve is steeper). There is a gap between the remaining energy levels of Route-1 and Route-2. This gap becomes small as the algorithm runs. When the remaining energy is low, the cost function can map a relative small difference into a bigger cost difference. Therefore, the nodes in Route-1 die first, and then nodes in Route-2.

From the above analysis we can see that cost function design must take into account the remaining energy variable. The cost function *f*(*B*) does not include this variable and ends up with poor performance. The cost function *f*(*A*) contains the remaining energy variable, and therefore has relatively better performance. However, cost functions that consider nodal remaining energy are not necessarily the optimal ones. An optimal cost function balances the energy consumption among nodes and maximizes network lifetime. Hence, the general principle of cost function design can be summarized as follows:

Principle 1. A cost function should have the following properties: small changes in nodal remaining energy can lead to large changes in the function value. □

Such a function can increase sharply the cost of a path whose nodal remaining energy is small, forcing nodes to select the path with more remaining energy and balancing energy usage. Among the cost functions in line with this principle, those in which small energy change can result in a large change in function value is more favorable. In the following, we propose two energy aware routing algorithm.

4.3. Exponential and sine cost function based routing (ESCFR)

The cost function of DEBR is essentially equivalent to function $f(x) = k/x$, where $x = E_i^r$(remaining energy), $k = e_{ij}$. The function of MTE is $f(x) = k$, where $x = 1$, $k = e_{ij}$. According to principle 1, we can design some functions with better performance than $f(x) = k/x$. Below is an illustrating example. It also shows the efficiency of principle 1 to simplify cost function design and enhance performance.

Exponential and sine functions are the kind of functions where small changes in variables can cause large changes in function values. We put these two types of functions together and construct an exponential function, as represented by Eq. (8). The following illustrates the correctness of *principle* 1 through the exponential and sine function.

$$f(x) = \exp(1/\sin(x)) \tag{8}$$

Exponential and sine function are functions with period π, and the cost function only needs to be a function from $\pi/2$ to π. Therefore, a cost function is needed to map nodal remaining energy to $[\pi/2, \pi]$. Assume the nodal initial energy is E_0, the current remaining energy is E_i^r. The mapping function can be given by

$$s = f(E_i^r) = \pi - \frac{\pi}{2}\frac{E_i^r}{E_0} \tag{9}$$

Then its cost function is

$$C_{ij} = e_{ij}\exp(1/\sin(s)) = e_{ij}\exp\left(1/\sin\left(\pi - \frac{\pi}{2}\frac{E_i^r}{E_0}\right)\right) \tag{10}$$

$$TC_{ik} = C_{ik} + C_{k,BS} \tag{11}$$

Based on this metric, a sensor *i* can select the best candidate neighbor *K* as next hop

$$K = Arg \min_{j \in N_i + \{i\}} (TC_{ij}) \tag{12}$$

We check whether the cost function C_{ij} in Eq. (10) is more effective than EC_{ij} of DEBR in Eq. (2). Figs. 5 and 6 show the total cost and the remaining energy of Route-1 and Route-2, respectively. In Fig. 5, the turning point *A* is obtained according to EC_{ij}. When the network starts to operate, Route-1 has a smaller TEC due to its shorter

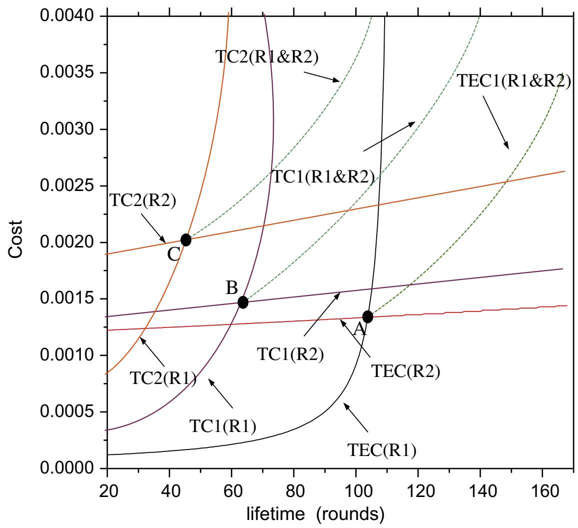


Fig. 5. The total cost of routes 1 and 2.

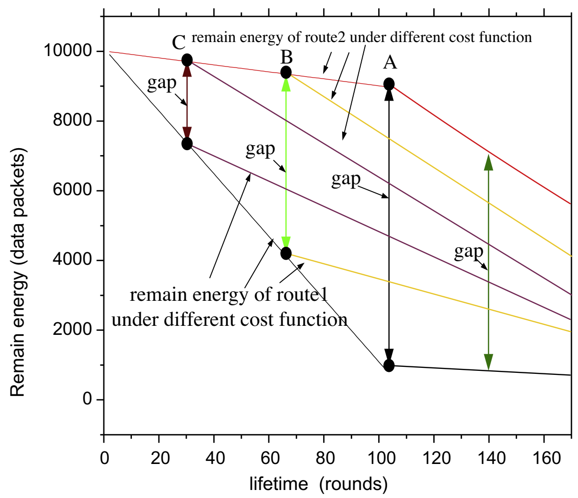


Fig. 6. The remaining energy of routes 1 and 2.

length, and thus the data in red region (Fig. 1) is routed through Route-1. This make Rout-1 have higher energy consumption than Route-2. Since the TEC gap between C_{ij} and EC_{ij} is large under the same energy gap, the cost function using exponential and sine function reaches a turning point B earlier. The remaining energy of nodes in Route-1 and Route-2 is shown in Fig. 6. We can see that, when the cost function in Eq. (2) is deployed, the remaining-energy gap expands from the beginning till turning point A, while with the cost function based on exponential and sine function, the remaining-energy difference begins to shrink from point B. It is clear that the energy consumption is more balanced for cost function based on exponential and sine function, the routing performance is better and the network lifetime is longer. Meanwhile, the exponential and sine function can be further improved. Eq. (8) maps remaining energy to $[\pi/2, \pi]$. In interval $[\pi/2, 3\pi/4]$, the exponential and sine function does not increase as fast as the independent variable, but it increases faster than the independent variable in $[3\pi/4, \pi]$. Thus, the nodal remaining energy can be instead mapped into $[3\pi/4, \pi]$, and Eqs. (9) and (10) can be replaced by the following Eqs. (13) and (14), respectively. If Eqs. (11) and (12) do not change, we can get the turning point C in Figs. 5 and 6, then the energy consumption is more balanced, and the network lifetime is further improved.

$$s = f(E_i^r) = \pi - \frac{\pi}{4}\frac{E_i^r}{E_0} \tag{13}$$

Then the cost function is

$$\begin{aligned} C_{ij} &= e_{ij}\exp(1/\sin(s)) \\ &= e_{ij}\exp\left(1\Big/\sin\left(\pi - \frac{\pi}{4}\frac{E_i^r}{E_0}\right)\right) \end{aligned} \tag{14}$$

Lemma 3. *The energy-balancing performance of exponential and sine cost function based routing (ESCFR) is no worse than DEBR.*

Proof. Suppose that there are two different routes Route-a and Route-b in the network. When the calculation results of the cost function are the same for DEBR and ESCFR, nodes can select the two paths with the same probability. The network is stable and the energy consumption rate of the routing paths is the same. If the remaining energy of these two paths is the same, then ESCFR and DEBR will clearly have the same energy-balancing performance. Since the routing paths are not the same, for the two cost functions to produce equal result, the remaining energy should not be equal. In this case, we assume that the remaining-energy difference between these two paths is Δe_1 under ESCFR, and Δe_2 under DEBR. We have $\Delta e_1 < \Delta e_2$ as ESCFR maps a small energy different into a larger cost function. Because the cost function values are equal, ESCFR only needs a small energy consumption difference to trigger energy balancing. Therefore, the energy-balancing performance of ESCFR is not inferior to that of DEBR. □

4.4. Double cost function based routing (DCFR)

According to the above analysis, ESCFR is not necessarily able to eliminate energy imbalance between established routing paths. We have the following Lemma 4.

Lemma 4. *Routing strategy based on mapping remaining energy into cost function cannot completely eliminate the energy consumption imbalance between routing paths with different hop counts.*

Proof. No matter which routing strategy (based on mapping remaining energy into cost function) is selected, as for two routing paths with length difference Δx, their energy consumption must not be balanced. This is because if they are balanced, there must be $TC(short) < TC(long)$(TC is the total cost of routing path), then the routing algorithm must select the shorter path, making the shorter path have to consume Δe more energy to guarantee $TC(short) = TC(long)$. That is, the shorter path must consume additional Δe energy to make up for the cost function difference Δx, i.e., $f(\Delta e) = TC(\Delta x)$. □

In a distributed routing algorithm based on cost function, although a better cost function can result in more balanced energy consumption, the algorithm can only reduce the difference in energy consumption to some limited extent. Note that the energy consumption rate of nodes in hotpots is surely high. If this factor is included in the cost function, we can then further improve the energy-balancing performance of the routing algorithm. The energy consumption rate of a node can be defined as follows

$$es_i = \frac{E^r_{t_i} - E^r_{t_j}}{t_j - t_i} \tag{15}$$

We need to map es_i into $[\pi/2, \pi]$. Assume the maximum es_i is $R_{\max}$. The mapping function can be easily obtained as follows

$$rs_i = f(es_i) = \frac{\pi}{2} + \frac{\pi}{2}\frac{es_i}{R_{\max}} \tag{16}$$

Then the new cost function is

$$\begin{aligned} RC_i &= e_{ij}\exp(1/\sin(rs)) \\ &= e_{ij}\exp\left(1/\sin\left(\frac{\pi}{2} + \frac{\pi}{2}\frac{es_i}{R_{\max}}\right)\right) \end{aligned} \tag{17}$$

$$TRC_{ij} = RC_i + \sum_{k \in path|j \ldots base} RC_k \tag{18}$$

Based on this metric, sensor i can select the best candidate, neighbor K as the next hop

$$K = Argmin_{j \in N_i + \{i\}}(TC_{ij} + TRC_{ij}) \tag{19}$$

Below we will analyze how the cost function in Eq. (19) can further improve network performance. Figs. 7 and 8 show the cost and energy consumption changes during network lifetime with the new cost function. In Fig. 7, the calculation of TC cost is the same as in Fig. 5, while the cost calculated in Eq. (18), which is according to energy consumption rate, is added to the total cost. TRC($R1$) and TRC($R2$) respectively represent the cost calculated according to energy consumption rate in Route-1 and Route-2.

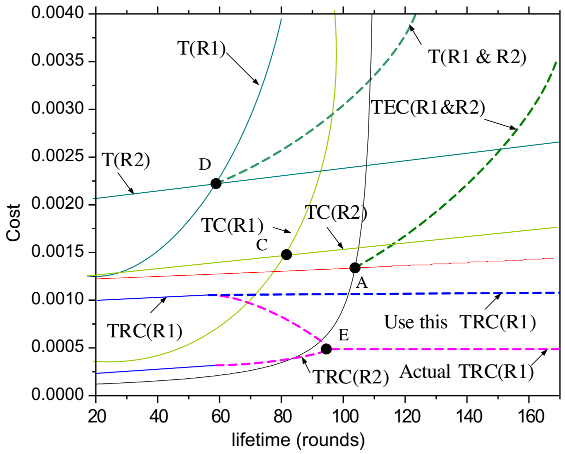


Fig. 7. The total cost of route 1 and 2.

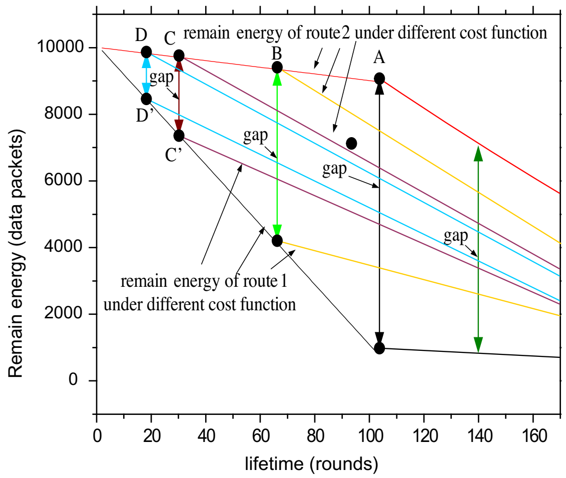


Fig. 8. The remaining energyof route 1 and 2.

When the network starts to operate, the total cost of Route-1 $T(R1) < T(R2)$, all data in the red region is routed through Route-1, thus the energy consumption rate of nodes in Route-1 is high, but its energy consumption rate remains at a fixed level, so TRC($R1$) at this stage maintains at a basic level, but higher than TRC($R2$) (due to cost function sensitive to the independent variables). Since $T(R1) = \mathrm{TC}(R1) + \mathrm{TRC}(R1)$ and $T(R2) = \mathrm{TC}(R2) + \mathrm{TRC}(R2)$, it is clear that no matter how cost TC is computed, the new cost function shifts the turning point (where $T(R1) = T(R2)$) from C to D. As shown in Fig. 8, the remaining energy of the two routing paths is reduced from the gap-line $C - C'$ to $D - D'$.

The new cost function makes the network process the energy balance in advance. After reaching the equilibrium point D, energy consumption rate of Route-1 declines, and that of Route-2 increases. At the end, these two paths have the same energy consumption rate (the energy consumption rate equals at point E in Fig. 7). As ESCFR, this new algorithm only considers remaining energy difference. Therefore, we improve it by changing the cost function as Rule 1:

Rule 1: We calculate the cost caused by energy consumption rate in an increasing mode instead of declining mode, i.e., no matter the energy consumption rate of current node is increasing or declining, the cost of energy consumption rate is calculated as the highest energy consumption rate so far. □

From Fig. 7, Route-1 has non-declining TRC($R1$) (the blue line), although the actual energy consumption rate declines after the turning point E (the brown line). While TRC($R2$) increases as the energy consumption rate increases after the turning point E, it is always smaller than TRC($R1$). It is because the two paths undertake the same amount of data, and as the number of routing paths increases, the amount of data undertaken per path decreases and so does the energy consumption rate. With *Rule 1* being applied, whether the energy consumption rates of the paths are equal or not, the cost of energy consumption rate is higher for the paths with more remaining energy, which may further reduce remaining-energy difference and achieve more balanced energy usage.

We demonstrate that DCFR's cost function is effective and able to adapt to the complex network through an example shown in Fig. 9. The cost changes of the three routing paths are shown in Fig. 10. Initially, since the node remaining energy is the same, the cost calculation of TC can be calculated according to the path length, thus $TC(R1) < TC(R2) < TC(3)$, so the data amount undertaken by each path meets $R1 > R2 > R3$ and as a result we have $TRC(R1) > TRC(R2) > TRC(R3)$. After a period of time, the energy of $R1$ goes down quickly, and TC ($R1$) increases fast, resulting in $T(R1)$ and $T(R2)$ meet at the equilibrium point C. The data amount undertaken by $R1$ declines, and that by $R2$ increases; whereas, according to *Rule 1*, $TRC(R1)$ does not decrease, and $TRC(R2)$ increases. These three paths achieve a balance at time of D, i.e., $T(R1) = T(R2) = T(R3)$. After point D, the data amount of $R1$ and $R2$ continues to decline, and that of $R3$ keeps increasing. Since there will be some delay to balance these three paths, the data amount balance points B and A occur later than the cost balance points C and D. Therefore, the energy consumption along these three paths is balanced after D. It is clear that DCFR reaches energy consumption balance status earlier than the algorithms considering only remaining energy.

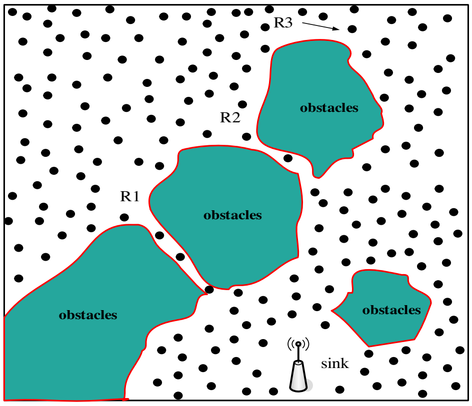


Fig. 9. Network topology 2.

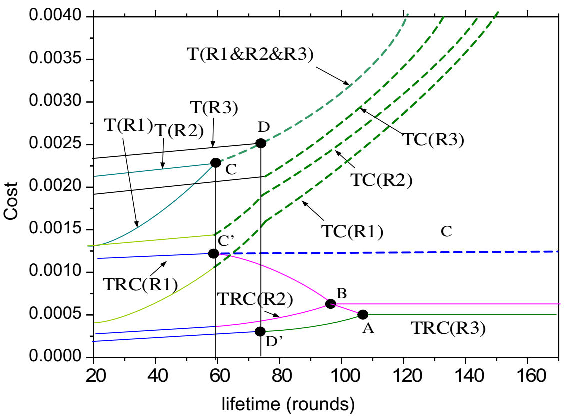


Fig. 10. The cost of route 1 and route 2 and route 3.

Lemma 5. *DCFR is not inferior to ESCFR in energy balancing.*

Proof. Suppose that there exist two paths Route-a and Route-b to select. If Route-a and Route-b have the same energy consumption rate, then DCFR and ESCFR have the same performance. Without loss of generality, we examine the case that the energy consumption rate of Route-a is greater than that of Route-b at some time. Assume that the rate difference between the two paths is $\Delta s > 0$. The cost in Route-a is small, so more energy needs to be consumed to make its total cost increase faster than Route-b and achieve balanced energy consumption rate. In DCFR, Route-a can have its total cost to be equal to Route-b only by relying on energy consumption difference, while in ESCFR the cost of energy consumption rate difference is TRS $(\Delta s) > 0$. Thus, Route-a may consume less energy to balance energy along the two paths, namely, the energy consumption difference needed is not so big as in ESCFRR. Therefore, the performance of DCFR is not inferior to ESCFR. □

The pseudo code of DCFR is given as follows:

(1) Discovery of neighbors and establishment of the initial routing table.
Each sensor broadcasts a setup message to neighboring nodes using a pre-set transmission power. This setup message includes node identifier (NI), minimum total cost (MTC) to the base station, remaining energy (RE), current energy consumption rate (CES). If a node is connected directly to the sink, it sets MTC = 0. After a source node S receives the information of each neighbor, it constructs a neighbor node table as shown in Table 2, where RE, CES, MTC is based on the message broadcasted by neighbor nodes, e_{ij} is the optimal transmit power based on signal power of S, and max energy speed (MES) the node largest energy consumption rate so far.

(2) *Calculation*: For each neighboring nodes $j \in Ns$, calculate the cost to the sink according to Eq. (19). As for each row in Table 2, calculate C_{sj} according to Eqs. (9) and (10), compare CES and MES, and make MES the largest value. Then map MES into RC_i according to Eq. (17); Calculate C_{ij} according to Eq. (10) or Eq. (14), put the calculation result of $RC_i + C_{ij}$ + MTC into the last column, indicting the cost of routing through this node to sink.After the above calculation, select

Table 2
Neighbor node table of node S.

Node id	RE	CES	MES	e_{ij}	MTC	Cost
S						
N_1	50000	75	98	170	300	410
...	...	...	...	...	...	...
N_k	80000	88	97	213	320	530

node N_i that has minimum cost as the next hop and update the row of node S as follows: put the ID number of N_i in the cost column and the cost of N_i in the MTC column, to show the minimum cost for S to route to the sink. At last, update the RE, CES columns.

(3) *Broadcast*: In each fixed period, S broadcasts its N_i, MTC, RE, CES to neighbors.
(4) For each message received from a neighbor, S node does the following:
  (a) if it is the new neighbor, then add the information (NI, MTC, RE, CES) as a new line to the neighbor information table, and make MES = CES, and calculate e_{ij}.
  (b) Otherwise, means the node is already in the neighbor table, then replace the original items (NI, MTC, RE, CES) with corresponding items received, and MES in this line is updated to the bigger one of new CES and MES.
  (c) No matter what the situation is, recalculate MTC of the updated line according to step 2. Then decide whether to update corresponding value of Cost and MTC according to the result.
(5) As for each packet received, node S select N_i from the cost column as the next hop to forward.
(6) End.

5. Experimental results

In this section, we provide experimental results to validate the effectiveness of exponential and sine cost function based routing (ESCFR) and double cost function based routing (DCFR) algorithm. We compare them with three existing algorithms discussed in [7,8]: direct communication (DC), minimum transmission energy (MTE), and distributed energy balanced routing (DEBR). In DC, every sensor simply transmits data directly to the base station without considering any energy efficiency. MTE considers multi-hop routing to save sensor power, but always chooses the path with the least total energy cost of transporting a packet to the sink. DEBR, ESCFR, DCFR all select the next hop based on the cost function value. We measure the network lifetime by the time when the number of dead nodes reaches 1% of the total nodes. We use this approximation to avoid the experimental randomness brought by premature death in random deployment of nodes. We consider that all nodes die when the number of dead nodes or nodes that cannot send data to the sink reaches 99% of the total nodes.

We carry out experimental verification using OMNET++ [21]. The main simulation parameters are given in Table 1. We use a simplified MAC layer where message losses, collisions, and duplications are not considered, as in [22,24]. Four different network scenarios are considered (see Figs. 11–14). The first is a network of 457 random-uniformly-deployed nodes in a circular area whose radius is 300 m with the sink located at (0,0). The second has the same topology as the first one, but it does not have the sink at the center, but at (0,200). The other two networks are rectangular networks, (0,0) is located at the upper-left corner, x direction is on the right and y direction

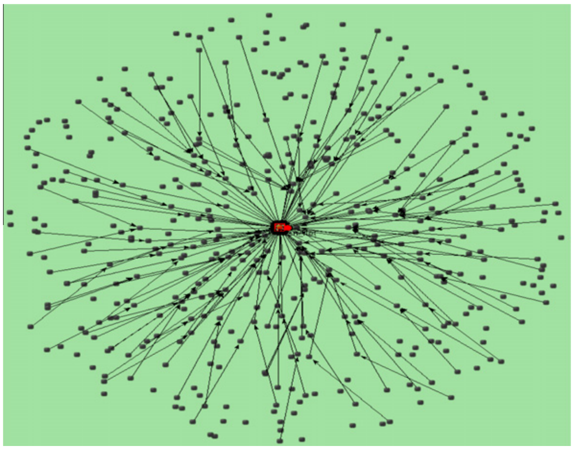


Fig. 11. Scene-1: Circular network with sink located at (0,0).

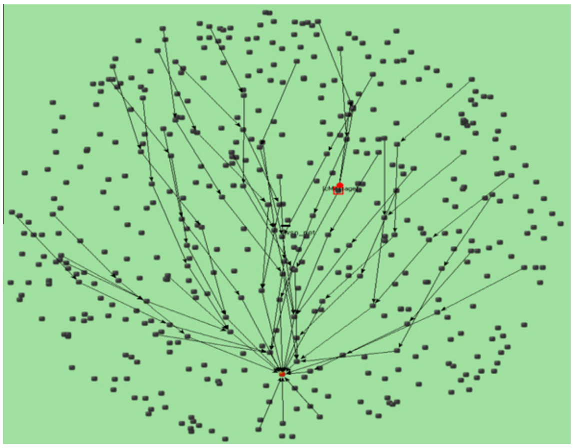


Fig. 12. Scene-2: Circular network with sink located at (0,200).

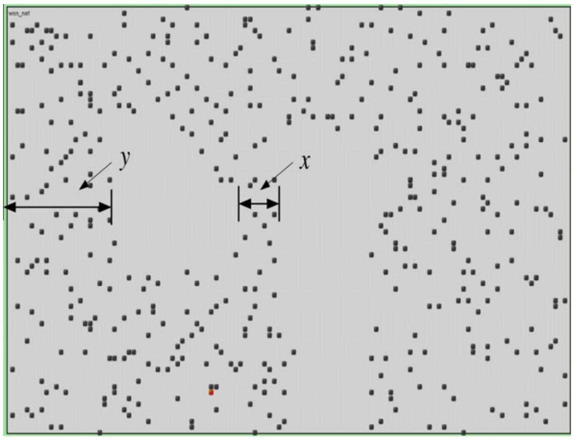


Fig. 13. Scene-3: Network with two obstacles.

is straight down. The third network is deployed in an 805×525 m area with the sink being located at (294,469). The fourth is a network whose area is

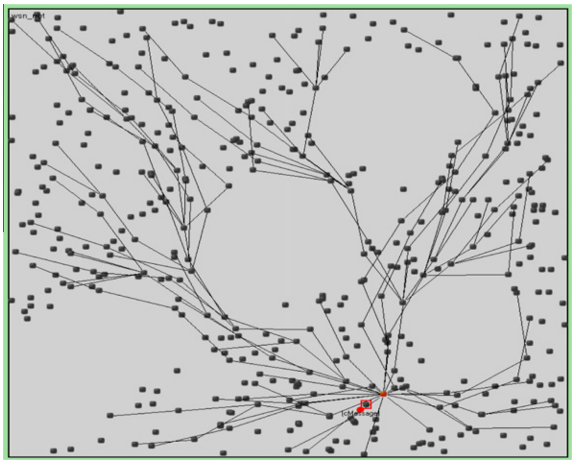


Fig. 14. Scene-4: Network with four obstacles.

550×550 m with the sink located at $(370, 475)$. 457 nodes are randomly uniformly deployed in these two networks, and in these two networks, there are obstacles where nodes cannot be deployed.

5.1. Lifetime of sensor network

This experimentation evaluates the performance of the algorithms with $r = 85$ m (r is the maximum reachable distance for nodal transmission power, called node transmission radius) in the above four network scenarios. Figs. 15–18 plots the number of living sensors against the number of rounds for each algorithm. In each data communication round, every sensor sends a single data packet to the base station.

We can have the following conclusions from experimental results in Figs. 15–18. (1). In all the scenes, ESCFR and DCFR both have longer lifetime than the other three algorithms. In Scene-1, the lifetime difference between

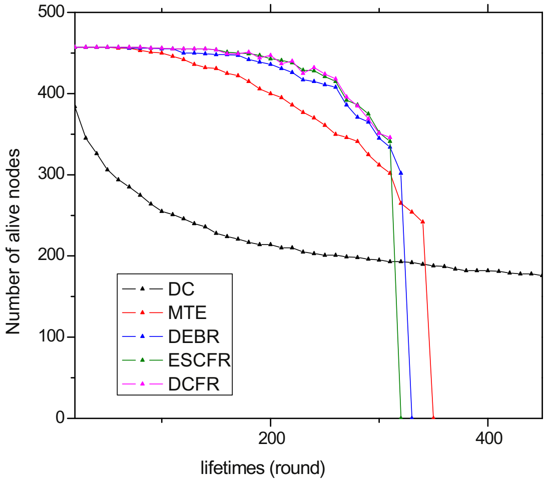


Fig. 16. The number of alive sensors over time under scene 2.

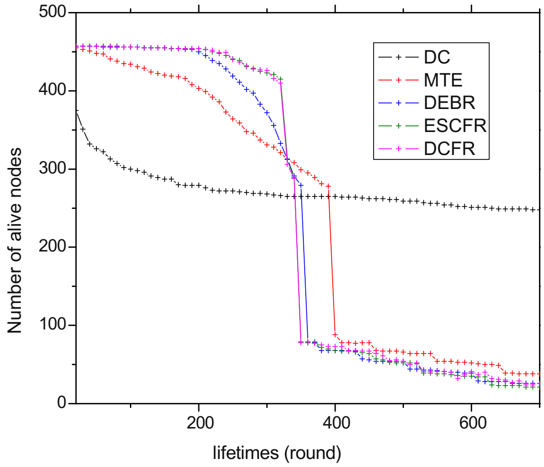


Fig. 17. The number of alive sensors over time under scene 3.

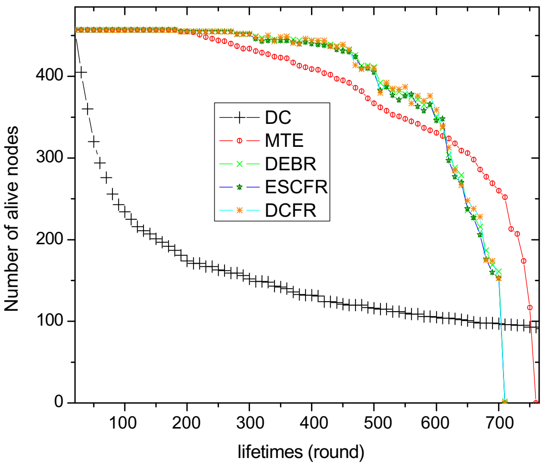


Fig. 15. The number of alive sensors over time under scene 1.

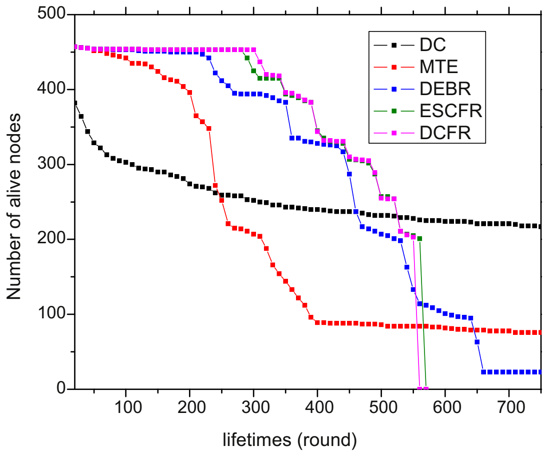
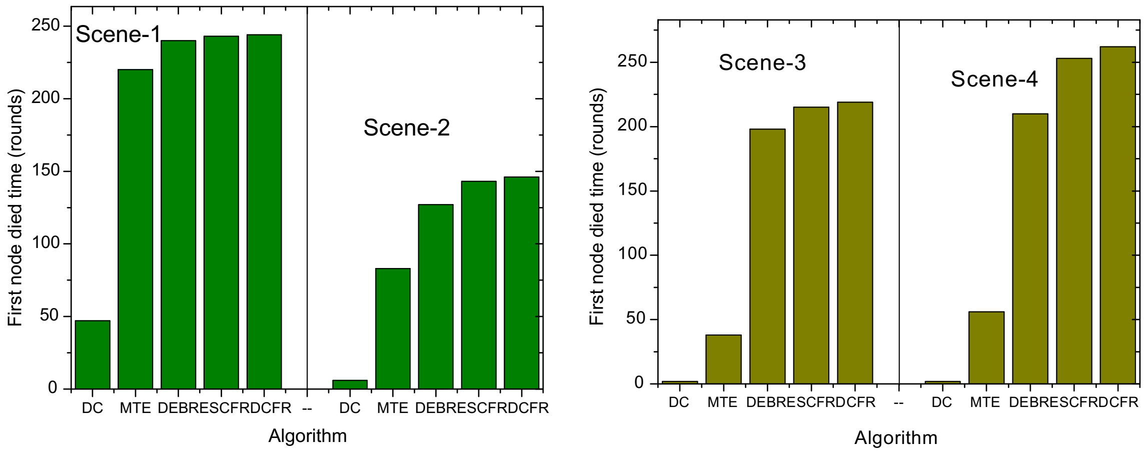


Fig. 18. The number of alive sensors over time under scene 4.



(a) Lifetime in Scene-1 and Scene-2 (b) Lifetime in Scene-3 and Scene-4

Fig. 19. The network lifetime under different scenes.

ESCFR, DCFR, MTE and DEBR is small. This is because the sink is located at the center, and the routing paths by these algorithms are basically the same. Because ESCFR, DCFR, DEBR consider the energy efficiency of the next hop (through $\frac{e_{ij}}{E_i^r}$), theses lead to a network lifetime slightly longer than MTE. In the other three scenes, energy balancing plays an important role in the network lifetime. In an energy-balanced network, the routing algorithm mainly improves the network lifetime through energy saving, while in an energy-imbalanced network (for example, different routing paths have the same routing length and their probabilities to be chosen as routing path by nodes are the same), mainly through energy balancing to improve network lifetime. (2) In different networks, nodes die in different ways. In networks whose lifetime is relatively short, nodes die very early. After the death of some nodes, the network load is decreased, and the data amount undertaken by remaining nodes decreases, and these nodes last long and die slow as a result. While in networks whose lifetime is relatively high and energy balancing is better, few nodes die at the beginning, but most nodes die at the same time in the later period. It is worth noting that in this experiment, dead nodes including nodes disconnected from the sink for having no paths to the sink. In Scene-1 and Scene-2, when nodes around the sink all die and the range of the dead zone around the sink is larger than r, there are no path from outside nodes to the sink, they are considered to be dead nodes. For this reason, in the experimental results of Scene-1 and Scene-2, we can see that lots of nodes die at the same time at last in the energy consumption balanced routing paths. In Scene-3 and Scene-4, when nodes die, nodes away from the sink are separated from the sink, resulting in a sharp increase of death rate of nodes.

Fig. 19 shows the network lifetime of different algorithms in these four scenes. The network lifetime by DC is very short, and ESCFR, DCFR, DEBR have an improved network lifetime than MTE. Further, ESCFR, DCFR can improve the network lifetime more than DEBR.

5.2. Energy balancing

Figs. 20 and 21 show the load of Route-1 and Route-2 in Scene-3 with different algorithms (vertical axis refers to the number of packets in 10 rounds). Since DC has no concept of routing, therefore, we simply compare the other three routing algorithms. As can be seen from the results, MTE only focuses on energy saving, its Route-1 bears a great amount of data at the beginning stage, therefore, Route-1 dies soon. After the death of Route-1, data amount and energy consumption rate of Route-2 rise rapidly, and Route-2 dies soon, and dead zones spread outward quickly. DEBR, ESCFR, DCFR have similar characteristics, that is, at the beginning, Route-1 has less cost, and therefore Route-1 has bigger load and consumes more energy (similar to MTE at this time). However, as the network operates, the cost of Route-1 increases rapidly, therefore, Route-1 and Route-2 achieve a routing balance (meaning, having equal routing cost). DCFR reaches such a balance first, then ESCFR, the last is DEBR. After achieving the routing balance, Route-1 and Route-2 share the network load, and thus

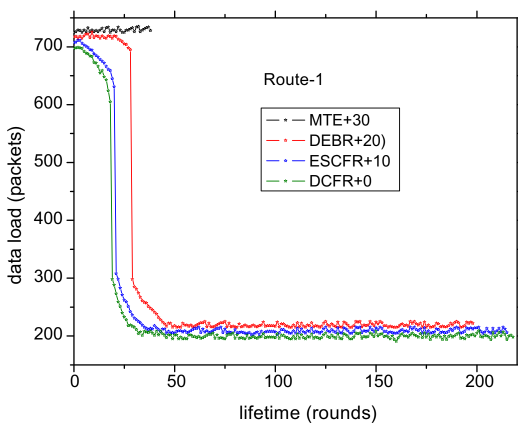


Fig. 20. Data packets by nodes of Route-1 in ten rounds in Scene-3.

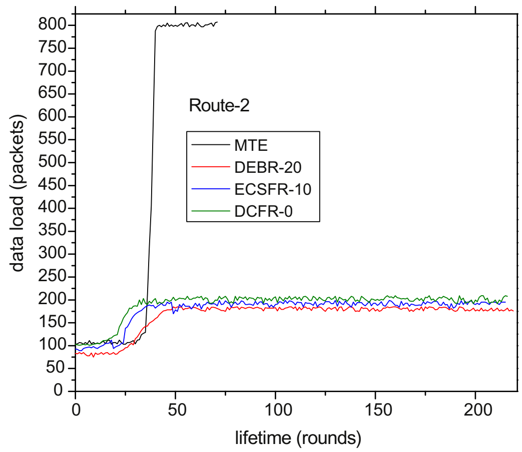


Fig. 21. Data packets by nodes of Route-2 in ten rounds in Scene-3.

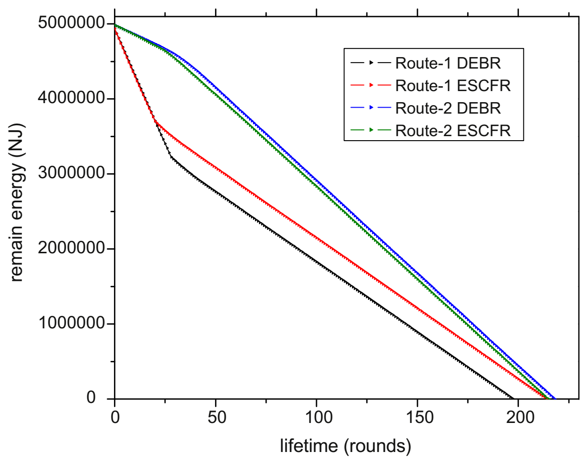


Fig. 22. Remaining energy with different algorithms in Scene-3.

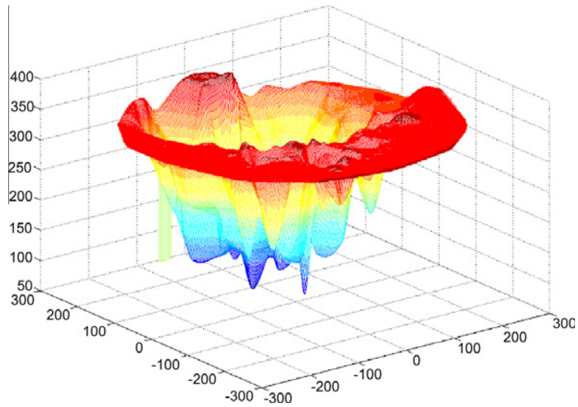


Fig. 23. Network remaining energy in Scene-3 (all nodes die under MTE).

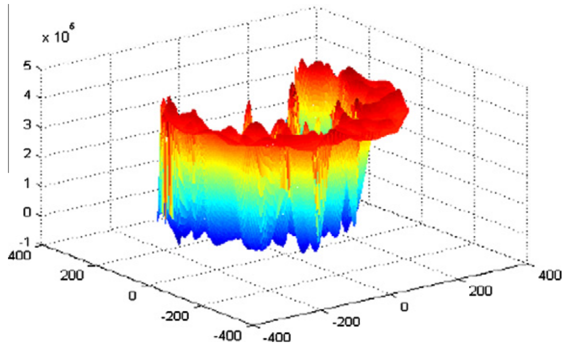


Fig. 24. Network remaining energy in Scene-3 (all nodes die under ESCFR).

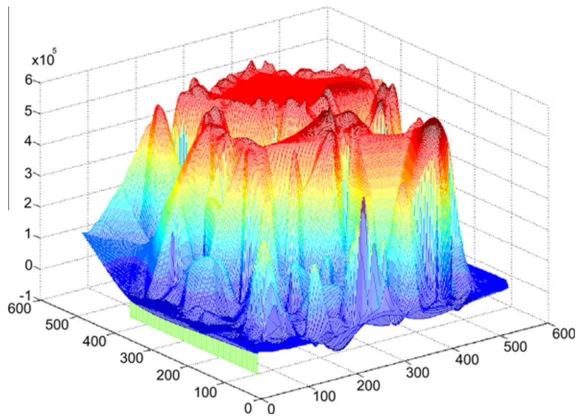


Fig. 25. Network remaining energy in Scene-4 (all nodes die under ESCFR).

make the nodal load to the average level. Since the total amount of data in the network is fixed, algorithms that achieve balanced routing first can balance the nodal data load well, and they have the longest network lifetime. Therefore, the network lifetime by these tested four algorithms is DCFR > ESCFR > DEBR > MTE.

We can calculate nodal remaining energy according to the load condition of different routing paths. Figs. 20–22 show the remaining energy of Route-1 and Route-2 by ESCFR and DEBR in Scene-3. It can be seen that our theoretical analysis results are consistent with the results given in Fig. 6, namely, our proposed algorithm not only reduces the uneven energy consumption of the two routing paths earlier, but also make the energy consumption difference among nodes in the two routing paths smaller.

Figs. 23 and 24 compare the nodal residual energy of MTE and ESCFR after all nodes die in Scene-3. In MTE, each node sends data to the sink along the shortest path, since the sink is located at (0, 200), nodes on the side near the sink consume a little energy and still have a lot of remaining energy when nodes on the other side dies early. In ESCFR data will be routed to the other side such that, when the network dies, nodes near the sink consume all the energy and only nodes far away the sink have lots of remaining energy. This confirms the energy balancing performance of our proposed algorithm. Fig. 25 shows the network remaining energy in Scene-4 under ESCFR, and Fig. 26 shows nodal data amount in Scene-2 under MTE when all nodes die. Due to space limitation, other experimental

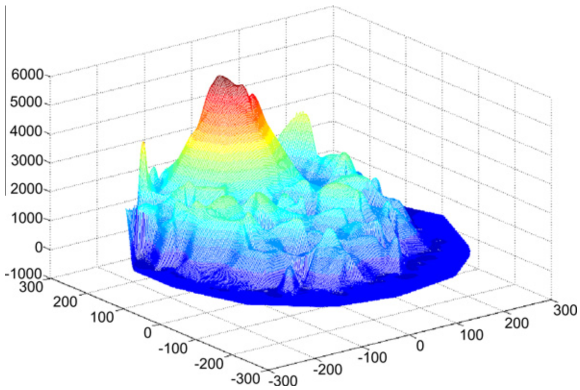


Fig. 26. Node data amount in Scene-2 (all nodes die under MTE).

results are not reported, but they are consistent with our previous analysis.

5.3. The effects of network parameters on performance

5.3.1. Varying the network scale

First we evaluate how the network shape in Scene-3 affects the cost function. We expand the width of y in Fig. 13 to test its influence. Fig. 27 shows the data amount undertaken by nodes of Route-1 and Route-2 with ESCFR under different y. It can be seen, after doubling y, the time to reach balance point is almost the same. But after the balance point, since there are more paths (the larger y, the more paths) to take data, nodes in Route-1and Route-2 undertake a declining amount of data. Fig. 28 shows the remaining energy of Route-1 and Route-2 before and after the expansion of y. It can be seen that after increasing y, the remaining energy of Route-1 and Route-2 declines more moderately, which means the time needed for remaining energy to drop to 0 is longer.

Fig. 29 shows the network lifetime under different network scale. The circular network's radius was varied between (200, 300, 400, 500), and the sink located at (0, 200). To maintain the same node density, the number of nodes deployed is respectively (203, 457, 812, 1269). It can be seen that the network lifetime has declined for all these algorithms. But ESCFR and DCFR are still higher than the other algorithms, and with the network expansion, they have an increased network lifetime.

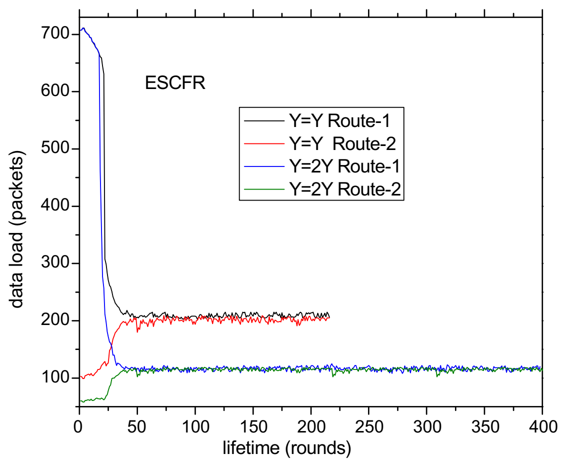


Fig. 27. The data amount when y is doubled.

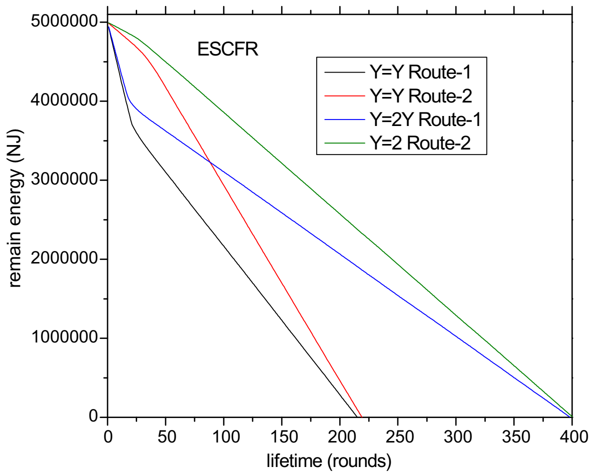


Fig. 28. The remaining energy when y is doubled.

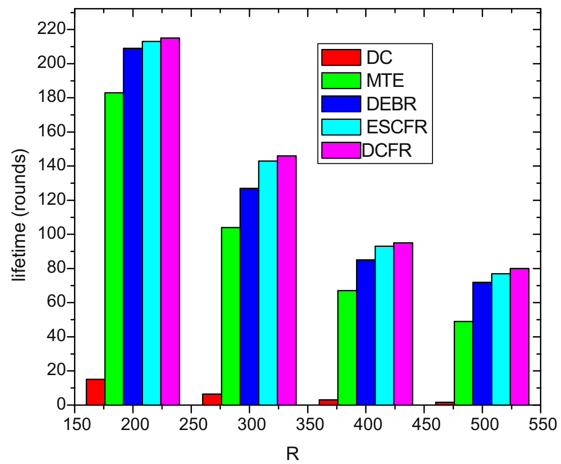


Fig. 29. The comparison of network lifetime under different network scale.

5.3.2. Varying the sink location and transmission radius r

Fig. 30 shows the network lifetime in Scene-3 when the sink is located at (0, 0), (0, 50), (0, 100), (0, 150), (0, 200), (0, 250), (0, 300). When the sink is located at the circle center, the network has highest lifetime. The farther the sink is away from the circle center, the shorter the network lifetime. Similarly, the performance of DCFR and ESCFR is still better than the other algorithms. The underlying causes are: the farther the sink is off the circle center, the more variation in the length of the routing paths to the sink, and therefore the more obvious the effect of these algorithms on energy balancing.

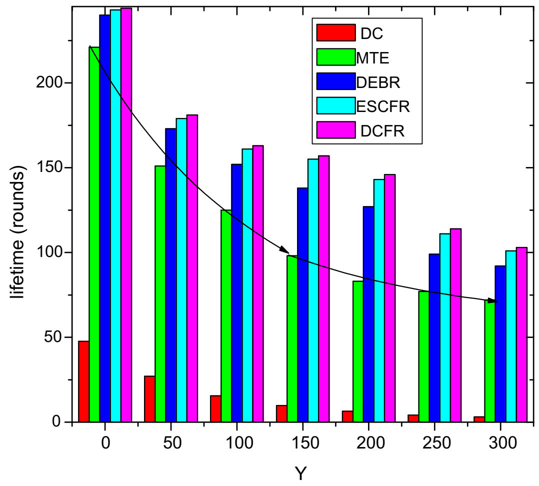


Fig. 30. Lifetime comparison under different location of the sink.

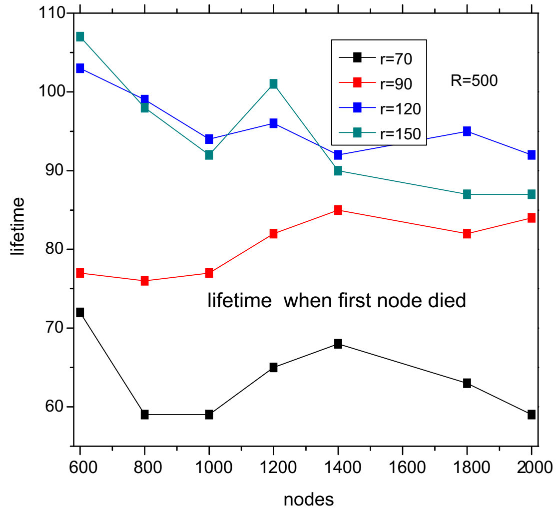


Fig. 32. Network lifetime under different node density.

Fig. 31 shows the network lifetime under different transmission radius r when the sink is located at the centre of a circular network whose radius is 500. Since the nodal energy consumption is 2 or 4 power of the transmission distance (d^2,d^4), different transmission radius r can lead to different network lifetime. By carefully selecting of node transmission power (namely transmission radius r), we can optimize the network lifetime.

5.3.3. Varying the number of sensor nodes

The network discussed in this section is a circular network with a radius of 500 m, and the sink is located at the circle center. The number of randomly deployed nodes are (600, 800, 100, 1200, 1400, 1600, 1800, 2000). Fig. 32 shows that although the number of nodes in the network varies greatly, the network lifetime difference is not large under the same transmission radius, indicating that the node density has little effect on the network lifetime. This result is consistent with the results in Ref. [23–25]. As node density goes up, the total amount of data transmitted grows, and the number of routing paths increases. In this case, with uniform node distribution, the work load per node changes barely on average, and as a result, the network lifetime remains roughly unchanged.

5.3.4. Routing system overhead

System overhead is defined as the data packets to the total packets (including system messages and data packets. System messages include four segments (NI, MTC, AE, CNS)). System messages are assumed to have the following length in our simulation: 1 data packet long, 1/2 data packet long, 1/4 data packet long. Each node in one data collection round broadcasts each of these messages (NI, MTC, AE, CNS) only once. So if each node forwards only one data packet in one round, the ratio of useful message load is 1/2 = 50%, 1/1.5 = 66.67%, 1/1.25 = 80%.

Fig. 33 represents the number of nodes under different system useful ratio in a circle network with radius $R = 300$ m, $R = 400$ m, the sink is located in (0, 200). It can be seen from Fig. 33, a large part of nodes only send a

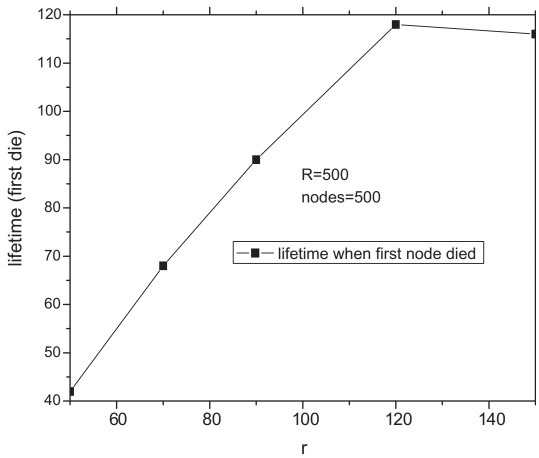


Fig. 31. Network lifetime under different transmission radius.

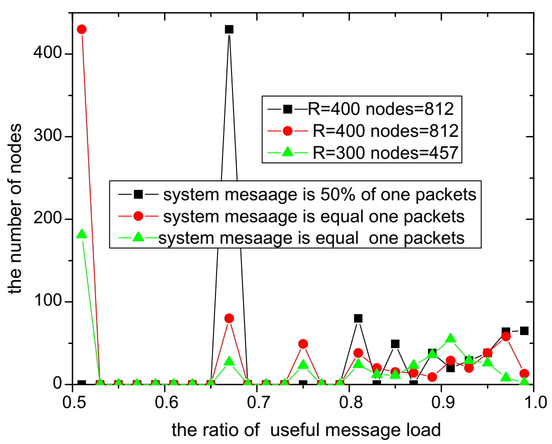


Fig. 33. The number of nodes under different system useful ratio.

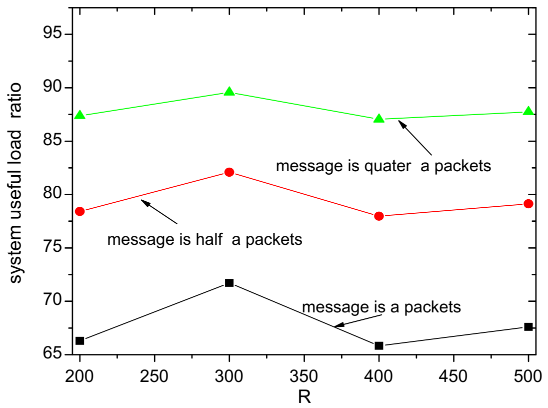


Fig. 34. The comparison of system useful load ratio.

single packet in one data collection cycle, the number of nodes are 192,423 (if the length of system message and data packet equals, then the node whose useful load ratio is 50% must only send a single packet in one data collection cycle, and the number of such nodes is respectively 192, 423). This is because for nodes who are within the distance $r = 85$ m from the network border, they do not need to forward data from other regions. These nodes constitute a large portion of the network. Since there are a fixed number of system messages transmitted by each node in a data collection round, the more data packets a node forwards, the higher the useful ratio. Experimental scene in Fig. 34 is to count the useful ratio of the entire network (i.e., the average useful ratio) in the network with radius (200, 300, 400, 500) and the sink being located at (0, 200).

6. Conclusion

In this paper, we have studied cost function based energy-aware routing. We proposed the general principles of cost function design and evaluation criteria. Further, we presented two novel energy aware cost based routing algorithms, named exponential and sine cost function-based routing (ESCFR) and double cost function based routing (DCFR). These two algorithms aim at maximizing the lifetime of the network by means of power consumption equalization. Comprehensive simulation results demonstrate that the algorithm can significantly improve network lifetime comparing with the best solution known in the literature, such as DC, MTE, and DEBR.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (61073104, 61073186), China Postdoctoral Science Foundation (20100471789), Specialized Research Fund for the Doctoral Program of Higher Education of China (20090162120074), Hunan Provincial Natural Science Foundation of China (09JJ6095), and Ontario Research Fund Canada.

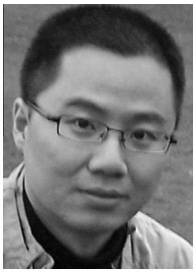
References

[1] G. Anastasi, C. Marco, M. Di-Francesco, P. Andrea, Energy conservation in wireless sensor networks: a survey, Ad Hoc Netw. 7 (3) (2009) 537–568.
[2] O. Powell, P. Leone, J. Rolim, Energy optimal data propagation in wireless sensor networks, J. Parall. Distrib. Comput. 67 (3) (2007) 302–317.
[3] K. Matrouk, B. Landfeldt, RETT-gen: a globally efficient routing protocol for wireless sensor networks by equalising sensor energy and avoiding energy holes, Ad Hoc Netw. 7 (3) (2009) 514–536.
[4] I.F. Akyildiz, W.L. Su, Y. Sankarasubramaniam, E. Cayirci, A survey on sensor networks, IEEE Commun. Mag. 38 (8) (2002) 393–422.
[5] C.-S. Ok, S. Lee, P. Mitra, S. Kumara, Distributed energy balanced routing for wireless sensor networks, Comput. Ind. Eng. 57 (1) (2009) 125–135.
[6] J. Rahme, A. Carneiro Viana, K. Al Agha, Looking for network functionalities extension by avoiding energy-compromised hotspots in wireless sensor networks, Ann. Telecommun. 63 (2008) 487–500.
[7] W.R. Heinzelman, A. Ch, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, in: Proceedings of the 33rd Hawaii International Conference on System Sciences, 2000.
[8] A. Rogers, E. David, N.R. Jennings, Self-organized routing for wireless micro sensor networks, IEEE Trans. Syst. Man Cybern. A 35 (3) (2005) 349–359.
[9] M. Ettus, System capacity, latency, and power consumption in multihop-routed SS-CDMA wireless networks, in: Proceedings of Radio and Wireless Conference (RAWCON), Colorado Springs, CO, 1998, pp. 55–58.
[10] O. Zytoune, M. El-aroussi, D. Aboutajdine, A uniform balancing energy routing protocol for wireless sensor networks, Wireless Pers. Commun. 55 (2) (2010) 147–161.
[11] Q. Li, J. Aslam, D. Rus, Online power-aware routing in wireless Ad hoc networks, in: Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom), 2001, pp. 97–107.
[12] J.H. Chang, L. Tassiulas, Maximum lifetime routing in wireless sensor networks, in: Proceedings of the Advanced Telecommunications and Information Distribution Research Program (ATIRP'2000), vol. 12, no. 4, College Park, MD, 2000, pp. 609–619.
[13] J. Park, S. Sahni, An online heuristic for maximum lifetime routing in wireless sensor networks, IEEE Trans. Comput. 55 (8) (2006) 1048–1056.
[14] K. Kar, M. Kodialam, T.V. Lakshman, L. Tassiulas, Routing for network capacity maximization in energy-constrained ad-hoc networks, in: International Conference on Computer Communications (INFOCOM), 2003, pp. 673–681.
[15] A.B. Mohanoor, S. Radhakrishnan, V. Sarangan, Online energy aware routing in wireless networks, Ad Hoc Netw. 7 (5) (2009) 918–931.
[16] C-S. Ok, S. Lee, P. Mitra, S. Kumara, Distributed routing in wireless sensor networks using energy welfare metric, Inform. Sci. 180 (2010) 1656–1670.
[17] IEEE.802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, IEEE std. 802.11-1999 ed., 1999.
[18] Q. Wang, W. Yang, Energy consumption model for power management in wireless sensor networks, in: 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Network (SECON'2007), 2007, pp. 142–151.
[19] M. Stemm, R.H. Katz, Measuring and reducing energy consumption of network interface in hand-held devices, IEICE Trans. Commun. E80-B (8) (1997) 1125–1131.
[20] T. Shepard, Decentralized channel management in scalable multihop spread spectrum packet radio networks[R], Technical report, MIT/LCS/TR-670, Massachusetts Institute of Technology Laboratory for Computer Science, 1995.
[21] A. Varga, The OMNET++ Discrete Event Simulation System, version 3.3. <http://www.omnetpp.org>, 2009.
[22] S. Kwon, NB. Shroff, Energy-efficient interference based routing for multi-hop wireless networks, in: IEEE INFOCOM, Barcelona, 2006, pp. 28–29.
[23] J. Li, P. Mohapatra, Analytical modeling and mitigation techniques for the energy hole problem in sensor networks, Pervasive Mobile Comput. 3 (3) (2007) 233–235.
[24] X. Li, H. Frey, N. Santoro, I. Stojmenovic, Strictly localized sensor self-deployment for optimal focused coverage, IEEE Trans. Mobile Comput. 10 (11) (2011) 1520–1533.

[25] S. He, J. Chen, Y. Sun, et al., On optimal information capture by energy-constrained mobile sensors, IEEE Trans. Veh. Technol. 59 (5) (2010) 2472–2484.

Anfeng Liu is an Associate Professor of School of Information Science and Engineering of Central South University. He is also a Member (E200012141M) of China Computer Federation (CCF). He received the M.Sc. and Ph.D degrees from Central South University, China, 2002 and 2005, both in computer science. Currently he is a Visiting Scholars in University of Waterloo, Canada. His major research interest is wireless sensor network.

Ju Ren received B.Sc. on 2009. Currently he is a master in School of Information Science and Engineering of Central South University. His research interest is in wireless sensor network.



Xu Li received his Ph.D. degree from Carleton University, Canada (2008), his M.C.S. degree from the University of Ottawa, Canada, 2005, and his B.Sc. degree from Jilin University, China, 1998. In 2004, he held a visiting researcher position at National Research Council Canada. He is currently a postdoctoral fellow at SITE, University of Ottawa and at CNRS/INRIA, France. His current research interests are wireless ad hoc, sensor, and actuator networks, mobile robots, distributed and localized algorithms, and wireless security. He was/is involved in many scholarly activities including ACM FOWANC'09, IMAGINE'09, AdHoc-Now'08& 09, LOCALGOS'09, WWASN'09, IFIP WSAN'08, IEEE MASS'07, etc.

Zhigang Chen received B.Sc. the M.Sc. and Ph.D degrees from Central South University, China, 1984, 1987 and 1998, He is a Ph.D. Supervisor and his research interests are in network computing and distributed processing.

Xuemin (Sherman) Shen (M'97-SM'02-F'09) received the B.Sc. (1982) degree from Dalian Maritime University (China) and the M.Sc. (1987) and Ph.D. degrees (1990) from Rutgers University, New Jersey (USA), all in electrical engineering.

He is a Professor and University Research Chair, Department of Electrical and Computer Engineering, University of Waterloo, Canada. Dr. Shen's research focuses on resource management in interconnected wireless/wired networks, UWB wireless communications networks, wireless network security, wireless body area networks and vehicular ad hoc and sensor networks. He is a coauthor of three books, and has published more than 400 papers and book chapters in wireless communications and networks, control and filtering. Dr. Shen served as the Technical Program Committee Chair for *IEEE VTC'10*, the Symposia Chair for *IEEE ICC'10*, the Tutorial Chair for *IEEE ICC'08*, the Technical Program Committee Chair for *IEEE Globecom'07*, the General Co-Chair for *Chinacom'07* and *QShine'06*, the Founding Chair for *IEEE Communications Society Technical Committee on P2P Communications and Networking*. He also served as a Founding Area Editor for *IEEE Transactions on Wireless Communications*; Editor-in-Chief for *Peer-to-Peer Networking and Application*; Associate Editor for *IEEE Transactions on Vehicular Technology*; *Computer Networks*; and *ACM/Wireless Networks*, etc., and the Guest Editor for *IEEE JSAC*, *IEEE Wireless Communications*, *IEEE Communications Magazine*, and *ACM Mobile Networks and Applications*, etc. Dr. Shen received the Excellent Graduate Supervision Award in 2006, and the Outstanding Performance Award in 2004 and 2008 from the University of Waterloo, the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada, and the Distinguished Performance Award in 2002 and 2007 from the Faculty of Engineering, University of Waterloo. Dr. Shen is a registered Professional Engineer of Ontario, Canada, an IEEE Fellow, an Engineering Institute of Canada Fellow, and a Distinguished Lecturer of IEEE Communications Society.