PAPER

Node Aggregation Degree-Aware Random Routing for Non-uniform Wireless Sensor Networks*

Xiaoming WANG[†a)], ***Nonmember***, **Xiaohong JIANG**[††b)], ***Member***, **Tao YANG**[†], **Qiaoliang LI**[†††], ***and* Yingshu LI**[††††], ***Nonmembers***

SUMMARY Routing is still a challenging issue for wireless sensor networks (WSNs), in particular for WSNs with a non-uniform deployment of nodes. This paper introduces a Node Aggregation Degree-aware Random Routing (NADRR) algorithm for non-uniform WSNs with the help of two new concepts, namely the Local Vertical Aggregation Degree (LVAD) and Local Horizontal Aggregation Degree (LHAD). Our basic idea is to first apply the LVAD and LHAD to determine one size-proper forwarding region (rather than a fixed-size one as in uniform node deployment case) for each node participating in routing, then select the next hop node from the size-proper forwarding region in a probabilistic way, considering both the residual energy and distribution of nodes. In this way, a good adaptability to the non-uniform deployment of nodes can be guaranteed by the new routing algorithm. Extensive simulation results show that in comparison with other classical geographic position based routing algorithms, such as GPSR, TPGF and CR, the proposed NADRR algorithm can result in lower node energy consumption, better balance of node energy consumption, higher routing success rate and longer network lifetime.

key words: *non-uniform WSN, LHAD, LVAD, probability selection mechanism, routing algorithm*

1. Introduction

Wireless sensor networks (WSNs) have many potential applications [1]–[3], like the smart home, environment monitoring, disaster surveillance, target tracking, etc. A WSN typically consists of a large number of wireless sensors, where each wireless sensor may be regarded as a node with limited wireless communication range, computing capability and energy. In a WSN, two nodes are called neighbor nodes if they can communicate with each other, a node generating packets is usually called a source node, and a node responsible for finally collecting and processing packets is a sink node, while a node participating in packet forwarding is an intermediate node. In fact, source nodes are the same as intermediate nodes. When a node generates packets and needs to forward these packets, the node is regarded as a source node, while the node is regarded as an intermediate node when it just forwards packets coming from other nodes. Usually, the sink node periodically broadcasts its geographic position coordinates to all the nodes, and each node stores the geographic position coordinates of the sink node into its buffer. When there are multiple sink nodes, the geographic position coordinates of each sink node are stored into the buffer of each node. Which sink node a certain packet will goes to is related to the practical application. Generally, the geographic position coordinates of the sink node which a certain packet will finally arrive at is enclosed in the head of the packet, and the intermediate nodes participating in packet forwarding can learn the geographic position of the sink node by checking the head of the received packet. This mechanism is useful in the multiple destinations routing in a WSM.

In WSNs, one of the challenging issues is to find one or several optimum routing paths from a source node to a certain sink node for packets transmitting, while some constraints, such as limited node energy, transmission latency and reliability, should be satisfied. This is the so-called routing issue on network layer [2]–[4].

So far, many routing algorithms for WSNs have been proposed, and these algorithms can be generally classified as network topology based algorithms and node geographic position based algorithms. Network topology based routing algorithms transmit packets relying on link information, while node geographic position based routing algorithms transmit packets according to physical geographic positions of nodes participating in routing, such as geographic position coordinates and distances between nodes. The basic idea of node geographic position based routing algorithms is to use the geographic position information of nodes to identify a certain fixed direction and fixed size region (namely forwarding region) for the node to transmit a packet, so that only the nodes inside this forwarding region may participate in packet forwarding, resulting in a reduction of unnecessary overhead from flooding in the entire network. In addition, most node geographic position based routing algorithms only need the local node position information to make a routing decision, thus they have a better adaptability to frequent network topology changes [2], [3]. Therefore, node geographic position based routing algorithms have been regarded as a promising routing solution for large scale WSNs [6]–[12].

Manuscript received October 30, 2009.
Manuscript revised August 9, 2010.

[†]The author is with the Shaaxi Normal University, China.
[††]The author is now with Future University Hakodate, Japan. He is also a Visiting Professor of the State Key Laboratory for Novel Software Technology, Nanjing University, China.
[†††]The author is with the Hunan Normal University, China.
[††††]The author is with the Georgia State University, USA.

[*]This work is supported by the NSF of China (NSFC) with grant No.60773224, 60970054, the Scientific Research Foundation for the Returned Overseas Scholars, State Education Ministry, China, and also the State Key Laboratory for Novel Software Technology, Nanjing University, China.

a) E-mail: wangxmsnnu@hotmail.com
b) E-mail: jiang@fun.ac.jp

DOI: 10.1587/transcom.E94.B.97

To the best of our knowledge, most node geographic position based routing algorithms, such as the LAR algorithm [5], the GPSR algorithm [6], the TPGF algorithm [7], and the CR algorithm [8], were designed based on the assumption that the deployment of nodes in a WSN is always uniform. In many practical WSN applications, however, the deployment of nodes is usually not uniform [9]. On one hand, some applications even require a WSN with non-uniform node deployment. For example, in a WSN based wild animal habitat monitoring system, many nodes are usually deployed in the regions frequently visited by the wild animals, whereas only a few number of nodes are sparsely deployed in other regions seldom visited by the wild animals. Similarly, in a WSN based forest fires monitoring system, more nodes are deployed in the regions where forest fires are more likely to occur. On the other hand, to prolong the network lifetime, more nodes are deployed in the regions closer to the sink node, while a small account of nodes are deployed in those far away from the sink node [9]. We say that a WSN with non-uniform node deployment is a non-uniform WSN. One of the features of a non-uniform WSN is that the number of the neighbor nodes for nodes in different regions may be very different. Thus it may be impossible to directly apply the existing node geographic position based routing algorithms in a non-uniform WSN. In a densely deployed region, a node usually has many neighbor nodes in its fixed forwarding region, thus it may spend too much time and energy to select a neighbor node as the next hop node for packet forwarding. Whereas in a sparsely deployed region, a node usually has very few neighbor nodes, and there may be no node in its fixed forwarding region, causing a failure of packet forwarding. In addition, a node void may be formed by a cluster of nodes whose energy is used up or whose components are physically destroyed, or some obstacles present. Theoretically, the non-uniform node deployment can effectively reduce the probability of node void occurrence, but it cannot completely avoid the occurrence of node voids in a WSN [9]. When a packet is forwarded to a node facing a node void, the packet may have to be discarded by the node. As a consequence, the packet cannot finally arrive at the sink node. Thus it is very important to bypass node voids in the routing process.

To enhance the performance of node geographic position based routing algorithms, in this paper, we introduce two new concepts: Local Horizontal Aggregation Degree (LHAD) and Local Vertical Aggregation Degree (LVAD), and propose a node aggregation degree-aware random routing (NADRR) algorithm for non-uniform WSNs based on these two novel concepts. Our basic idea is to first identify one size-proper forwarding region for each node participating in routing based on LHAD and LVAD of the node, then select the next hop node from the size-proper forwarding region in a probabilistic way, where the selection probability is determined based on both residual energy and distribution of nodes participating in packet forwarding. To bypass node voids in the routing process, we further propose a backward routing policy, where the routing is allowed to return to the previous-hop node to search for a new routing path again. The simulation results indicate that the NADRR algorithm outperforms the available algorithms of GPSR, TPGF and CR in terms of node energy consumption, energy consumption balance, packet transmission latency, network lifetime, and transmission success rate. Moreover, the proposed routing algorithm has a very good adaptability to the dynamic and frequent topology change of a non-uniform WSN.

The remainder of the paper is organized as follows. In Sect. 2, the related work is introduced. The NADRR algorithm and its main building blocks are discussed in Sect. 3, and the simulation results are presented in Sect. 4. Section 5 concludes this paper.

2. Related Work

Thanks to the availability of cheap geographic positioning instruments such as GPS receivers, some geographic position based routing algorithms have been proposed for Mobile Ad Hoc Networks (MANETs). In [6], the authors proposed a distributed routing algorithm for MANETs, namely the greedy parameter stateless routing (GPSR) algorithm. This algorithm uses two different packet forwarding strategies: greedy forwarding and perimeter forwarding. The greedy forwarding means that a node always selects from all its neighbors the one that is the closest to the sink node as its next hop node for packet forwarding when it receives a packet to forward. When a node cannot find a neighbor closer to the sink than itself, we say, the node encounters a void. At this moment, the GPSR algorithm initiates a bypassing void routing mechanism based on the right (left) hand rule, i.e., the perimeter forwarding. In the perimeter forwarding, the received packet will move around the right (left) boundary nodes of the void until arriving at a node closer to the sink node than the node which initiated the perimeter forwarding. After that the perimeter forwarding is over, and the greedy forwarding works again. Since it can adapt to the dynamic change of network topology, the GPSR algorithm is broadly applied in WSNs. However, the GPSR algorithm only uses the knowledge of the shortest distances between nodes to select the next hop nodes, thus it always builds the same path to forward packets from a certain source node to the sink node. As a result, the energy of nodes on the routing path is quickly used up, and more or bigger voids may be formed. In [5], the authors proposed a geographic position-aided routing (LAR) algorithm for MANETs. The LAR algorithm applies a limited flooding policy to forward packets, and uses an on-demand routing mechanism to reduce the total network overhead. Like the GPSR algorithm, the LAR algorithm also adapts to the dynamic change of network topology, thus the LAR algorithm is also being widely used in WSNs. In [8], the authors proposed a compass routing (CR) algorithm based on the geometric graph to find the shortest path from a node to another for generalized geometric networks. In the CR algorithm, each node may be represented by a vertex, the direct neighboring relationship between two nodes may be modeled by an edge connecting these two nodes, and each vertex learns the geographic position coordinates of the sink node and itself. As a result, a geometric graph is formed. When a node needs to send a packet toward the sink node, the node will choose the edge closest to the straight line segment connecting the node and the sink node. Although the CR algorithm can also adapt to the dynamic change of network topology, the CR algorithm cannot effectively balance the energy consumption of nodes.

Since the algorithms of GPSR and LAR are originally designed for MANETs and the CR algorithm is originally designed for generalized geometric networks, they do not take into account the important energy conservation issue of WSNs. Recently, the authors in [7] explored the geographic position based routing in Wireless Multimedia Sensor Networks (WMSNs) and proposed the TPGF algorithm. The TPGF algorithm uses the same greedy forwarding policy for packet transmission as one of the GPSR algorithm, and focuses on looking for the maximum number of node-disjoint routing paths from a source node to the sink node to increase the throughput of multimedia streams during a short network lifetime. The TPGF algorithm can be repeatedly executed to look for multiple node-disjoint routing paths. When only generating one routing path to forward packets, the TPGF algorithm is basically the same as the GPSR algorithm, thus the TGPS algorithm may be regarded as an improved GPSR algorithm for WMSNs. However, the TPGF algorithm does not take into consideration the balance of energy consumption of nodes for searching a routing path, thus it cannot effectively prolong network lifetime.

In the above routing algorithms, each node can learn its own geographic position coordinates through a geographic position device, and also can learn the geographic position coordinates of its neighbor nodes through periodically broadcasting the beacon information and that of the sink node by reading the heads of received packets. Thus, the distances between nodes may be calculated. Based on vertex position information, the routing problem can be mapped to the problem of finding an optimum path from a given vertex to another in a weighted graph, and many approximate algorithms have been proposed to solve this problem (e.g., the Dijkstra's algorithm). However, these routing algorithms are either energy-inefficient or time latency-high. Especially, when the deployment of nodes is not uniform in a WSN, these algorithms are not efficient in reducing or balancing the energy consumption of nodes or increasing the routing success rate, or prolonging the network lifetime.

3. The Proposed Routing Scheme

In this section, we first introduce the network model and assumptions, and then define two new concepts, LHAD and LVAD. Based on these two concepts, we further discuss how to determine a size-proper forwarding region for a node and propose a probabilistic mechanism to select the next hop node considering both the residual energy and geographic position information of nodes. Finally, we design the NADRR algorithm for non-uniform WSNs.

3.1 Network Model and Assumptions

Suppose that a non-uniform WSN consists of m nodes randomly deployed inside a rectangle area with length Y and width X. To simplify the discussion, we assume there is only one sink node. A non-uniform WSN may be represented by an undirected graph $G(V, E)$, where V is the set of nodes, E is the set of edges, and $E \subseteq V \times V$. At time t, for any two nodes $n_i, n_j \in V$, where $i \neq j$, suppose that the geographic position coordinates of n_i and n_j may be detected by a certain geographic position service system (like GPS), denoted by coordinates (x_i, y_i) and (x_j, y_j), respectively. The distance between n_i and n_j, denoted by $d_{i,j}$, is defined by

$$d_{i,j} = \sqrt{(x_i - x_i)^2 + (y_i - y_j)^2}. \tag{1}$$

The communication range of each node is a circle with radius r. For any two nodes $n_i, n_j \in V$, if $d_{i,j} \leq r$, we say that n_i and n_j are two neighboring nodes and they can directly communicate with each other. If nodes n_i and n_j are two neighbor nodes, there is an undirected edge $(n_i, n_j) \in E$. At time t, the residual energy of node n_i is denoted as $e_i(t)$. Suppose that all the nodes are static once they are non-uniformly deployed in an area, and each node knows the geographic positions of its neighbors and its own. A source node generating packets knows the geographic position of the sink node towards which the generated packets will be transmitted. For instance, in the event driving routing, each node generating packets learns the geographic position of the sink node towards which the packets will be transmitted.

The problem to address is to find optimum paths from a source node to the sink node, with the aims of balancing and minimizing the total energy consumptions of the nodes, and the time delay of packet transmission is also minimized as much as possible. Unfortunately, when considering multiple constraints, the optimal routing problem for large scale WSNs has been proved to be NP-complete [11], [13]–[16].

3.2 LHAD and LVAD

For any node n_i, its communication range, denoted as O_i, is a cycle with radius r. For simplicity, the center of O_i is denoted by I. The communication range of the sink node n_s is denoted as O_s, and the center of O_s is S. Straight line segment IS represents the distance from I to S, and the length of IS is $d_{i,s}$. As shown in Fig. 1, BC is a diameter of O_i, BC is vertical to IS and they intersect at I, that is, $\angle BIS = \angle CIS = \pi/2$. Thus O_i is cut into two equal parts. The part closer to S is denoted by Ω_i. In this paper, all the nodes in Ω_i are regarded as the forwarding neighbors of n_i except the ones on BC. The set of the forwarding neighbors of n_i is denoted by N_i and formally defined as:

$$N_i = \{n_k | n_k \in \Omega_i\}. \tag{2}$$

We represent the number of all the elements in N_i as $|N_i|$.

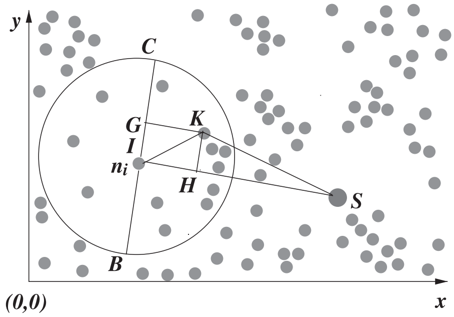

Fig. 1 A non-uniform WSN.

Then the following equation is true:

$$|N_i| \geq 0. \tag{3}$$

For a node $n_k \in N_i$ (say K in Fig. 1), the distance from node n_k to the diameter BC, denoted as $L_{i,k}$, is the length of KG. Similarly, the distance from node n_k to IS is the length of KH, denoted as $T_{i,k}$. Thus we have $\angle IHK = \angle IGK = \pi/2$. Moreover, according to Eq. (1) we have

$$d_{k,s} = \sqrt{(x_k - x_s)^2 + (y_k - y_s)^2}, \tag{4}$$

and

$$d_{i,k} = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2}. \tag{5}$$

In the following, the symbol '$\cdot$' represents the digital product operator. In $\triangle KIS$, we have

$$\cos\angle KIS = \frac{d_{i,k}^2 + d_{i,s}^2 - d_{k,s}^2}{2 \cdot d_{i,k} \cdot d_{i,s}}. \tag{6}$$

Thus

$$\angle KIS = \arccos\frac{d_{i,k}^2 + d_{i,s}^2 - d_{k,s}^2}{2 \cdot d_{i,k} \cdot d_{i,s}}. \tag{7}$$

In $\triangle IHK$, since $\angle IHK = \pi/2$, we have

$$T_{i,k} = d_{i,k} \cdot \sin\left(\arccos\frac{d_{i,k}^2 + d_{i,s}^2 - d_{k,s}^2}{2 \cdot d_{i,k} \cdot d_{i,s}}\right). \tag{8}$$

In $\triangle IGK$, since $\angle IGK = \pi/2$ and $\angle GIK = (\pi/2) - \angle KIS$, we have

$$L_{i,k} = d_{i,k} \cdot \sin\left(\frac{\pi}{2} - \arccos\frac{d_{i,k}^2 + d_{i,s}^2 - d_{k,s}^2}{2 \cdot d_{i,k} \cdot d_{i,s}}\right). \tag{9}$$

Therefore, we have

$$L_{i,k} = d_{i,k} \cdot \cos\left(\arccos\frac{d_{i,k}^2 + d_{i,s}^2 - d_{k,s}^2}{2 \cdot d_{i,k} \cdot d_{i,s}}\right). \tag{10}$$

Similarly, we can calculate the distances from other nodes in N_i to BC and IS, respectively. The average value of the distances from each node in N_i to BC is denoted as L_i. When $|N_i| > 0$, we have

$$L_i = \frac{\sum_{n_k \in N_i} L_{i,k}}{|N_i|}. \tag{11}$$

We name L_i the Local Horizontal Aggregation Degree (LHAD) of the forwarding neighbors of n_i. L_i can be used to describe the horizontal distribution of the forwarding neighbors of n_i. The larger the L_i, the further the forwarding neighbors of n_i away from BC. Conversely, the smaller the L_i, the closer the forwarding neighbors of n_i to BC. Similarly, the average value of the distances from each node in N_i to IS is denoted as T_i. When $|N_i| > 0$, we have

$$T_i = \frac{\sum_{n_k \in N_i} T_{i,k}}{|N_i|}. \tag{12}$$

T_i is regarded as the Local Vertical Aggregation Degree (LVAD) of the forwarding neighbors of n_i, and it may be used to describe the vertical distribution of the forwarding neighbors of n_i. A larger T_i implies more forwarding neighbors of n_i are far away from IS. Conversely, a smaller T_i means more forwarding neighbors of n_i are close to IS.

3.3 Forwarding Region Control for a Non-uniform WSN

Before node n_i forwards a packet to the sink node, we identify a size-proper forwarding region for this node based on its LHAD and LVAD to efficiently limit the routing search scope of n_i. The objective is to accelerate the process of selecting the next hop node of n_i, and save energy. Informally, our basic idea is that if node n_i has a lot of neighbor nodes, we identify a smaller forwarding region for node n_i. Otherwise, we identify a lager forwarding region for node n_i to include a proper number of forwarding neighbors. That is, the size of the forwarding region of n_i is not fixed and it is decided by the geographic distribution of the forwarding neighbors of n_i. Thus, this routing scheme has good adaptability to the non-uniform deployment of nodes in a non-uniform WSN.

To determine a proper forwarding region for node n_i, we first define the following function:

$$f(T_i, L_i, r, N_i) = \frac{\pi}{3}\left(1 + \frac{T_i}{r} - \frac{L_i}{r} + \log^{r}_{(1+|N_i|+r)} - \log^{|N_i|}_{(1+|N_i|+r)}\right), \tag{13}$$

where r is the communication radius of n_i, L_i is the LHAD of n_i, and T_i is the LVAD of n_i. The function f increases as the value of T_i or r increases, whereas f decreases as the value of L_i or $|N_i|$ increases. When $|N_i|$ is very large, f may be less than 0. At this moment, most forwarding neighbors of n_i are on the line IS or very close to the line IS. Considering the meanings of L_i and T_i in Eq. (11) and Eq. (12), we may use f to determine a size-proper forwarding region for node n_i. As shown in Fig. 2, for any node n_i in a non-uniform WSN, we regarded $\angle EIF$ as the forwarding angle of n_i, denoted by α_i. Based on the function $f(T_i, L_i, r, N_i)$, α_i is defined as

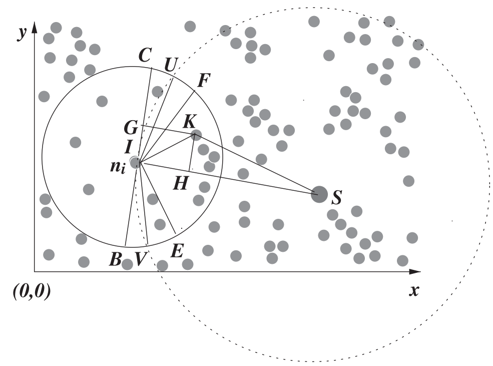

Fig. 2 The forwarding angle $\angle EIF$ and the maximum forwarding angle $\angle UIV$ of n_i.

$$\alpha_i = \begin{cases} f(T_i, L_i, r, N_i), & f(T_i, L_i, r, N_i) > 0 \\ 0, & f(T_i, L_i, r, N_i) \leq 0 \end{cases}. \tag{14}$$

When $\alpha_i = 0$, only n_i's neighbor nodes on the line *IS* are the candidate next hop nodes of n_i. In this case, if there is no n_i's neighbor node on the line *IS*, the routing may return to the previous-hop node of n_i, and search a new routing path again (refer to Sect. 3.5). As for the property of forwarding angle α_i, we have the following proposition (refer to Appendix for proof).

Propasition 1: For any node n_i in a non-uniformity WSN, its forwarding angle α_i satisfies $0 \leq \alpha_i < \pi$.

To avoid a loop route, we create a disk O_s centered at *S* with radius $d_{i,s}$, as shown in Fig. 2. Then O_s and O_i must have two intersection points, denoted by *U* and *V*, respectively. We can easily see that $d_{i,u} = d_{i,v} = r$, $\angle UIS = \angle VIS$ and $d_{u,s} = d_{v,s} = d_{i,s}$. We name the angle $\angle UIV$ the maximal forwarding angle of n_i, denoted by β_i. From the mathematics, the following equation is necessarily true:

$$\beta_i = 2 \cdot \arccos\left(\frac{r^2 + d_{i,s}^2 - d_{u,s}^2}{2 \cdot r \cdot d_{i,s}}\right). \tag{15}$$

Since $d_{u,s} = d_{i,s}$, from Eq. (15) we have

$$\beta_i = 2 \cdot \arccos\left(\frac{r}{2 \cdot d_{i,s}}\right). \tag{16}$$

We name an angle α_i' an effective forwarding angle of n_i, and α_i' is defined as:

$$\alpha_i' = \begin{cases} \alpha_i & if \quad \alpha_i \leq \beta_i \\ \beta_i & otherwise \end{cases}. \tag{17}$$

When $\alpha_i' = \alpha_i$, the fan-shaped region IEFI is called the forwarding region of n_i. When $\alpha_i' = \beta_i$, the fan-shaped region IUVI is called the maximal forwarding region of n_i. For simplicity, we denote the forwarding region of n_i by Ω_i'. Except node n_i, the nodes included in Ω_i' are called the candidate next hop nodes of n_i. The set of all the candidate next hop nodes of n_i is denoted by CN_i, and the set CN_i is defined as:

$$CN_i = \{n_k | n_k \in \Omega_i'\} - \{n_i\}. \tag{18}$$

We limit the selection of next hop node of n_i within the set CN_i. For instance, for node n_i in Fig. 2, the forwarding region of n_i is the fan-shaped region IEFI. There are six candidate next hop nodes of n_i, and one of the six nodes may be selected as the next hop node of n_i to forward n_i's packets toward the sink node n_s. When the set of the candidate next hop nodes of a node is empty, a backward routing mechanism is used, which will be described in Sect. 3.5 in details.

3.4 Probabilistic Mechanism for Selecting a Next Hop Node

Suppose node n_i is the node with the routing detection packet and node n_s is the sink node. For each node n_k in CN_i, we now determine the probability $p_{i,k}$ of selecting node n_k as the next hop node of n_i. To balance the energy consumptions of nodes and minimize the transmission time latency of packets, we combine the residual energy and geographic position information of nodes in calculating the selection probability $p_{i,k}$. For this purpose, we introduce weight parameters w_1 and w_2, where $0 \leq w_1, w_2 \leq 1$ and $w_1 + w_2 = 1$. w_1 is used to describe user's preference of balancing the energy consumption of nodes, while w_2 is used to indicate user's preference of reducing the transmission time latency of packets. Intuitively, a larger w_1 means that the candidate next hop node with higher residual energy is selected as the next hop node of the current node. As a result, the energy consumptions of nodes can be better balanced. A larger w_2 means that the candidate next hop node with a longer distance to the current node is selected as the next hop node of the current node. As a consequence, a shorter routing path from the source node to the sink node is built. Based on w_1 and w_2, the probability $p_{i,k}$ is defined by

$$p_{i,k} = \frac{w_1 \cdot \log_{10}^{(1+\varepsilon_{i,k})} + w_2 \cdot \log_{10}^{(1+\ell_{i,k})}}{\sum\limits_{n_j \in CN_i} (w_1 \cdot \log_{10}^{(1+\varepsilon_{i,k})} + w_2 \cdot \log_{10}^{(1+\ell_{i,k})})}. \tag{19}$$

where $\varepsilon_{i,k}$ is the measure of the relative residual energy of neighbor node n_k of n_i, and when $|CN_i| > 0$, $\varepsilon_{i,k}$ is given by

$$\varepsilon_{i,k} = \frac{e_k}{\sum\limits_{n_j \in CN_i} e_j}. \tag{20}$$

In Eq. (20), e_k is the current residual energy of n_k, and the meaning of e_j is similar to that of e_k. In Eq. (19), $\ell_{i,k}$ is the measure of the relative distance from n_k to n_s, and $\ell_{i,k}$ is defined by

$$\ell_{i,k} = D_{i,k} \cdot R_{i,k}, \tag{21}$$

where, $D_{i,k}$ is the measure evaluation of the distance of n_k to *IS*, and $D_{i,k}$ is defined by

$$D_{i,k} = \frac{d_{i,s}}{d_{i,k} + d_{k,s}}. \tag{22}$$

Clearly, $0 < D_{i,k} \leq 1$. The smaller the $D_{i,k}$, the smaller the distance from n_k to *IS*. To a certain extent, this describes the

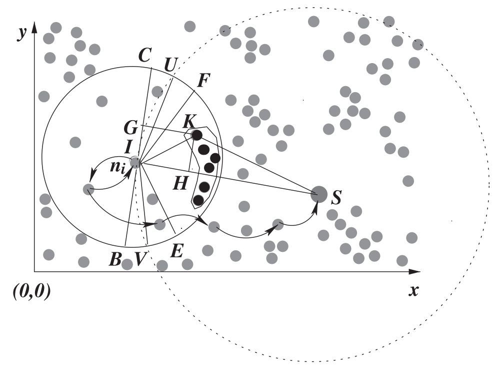

Fig. 3 Node void formed by a cluster of dead nodes (black nodes).

distance feature from n_k to the sink node n_s. When node n_k is on *IS*, $D_{i,k}$ is 1, and this means that the distance of n_i to n_s through n_k is minimized. In Eq. (21), $R_{i,k}$ is the measure evaluation of the relative distance from n_k to n_i, and when $|CN_i| > 0$, $R_{i,k}$ is defined as

$$R_{i,k} = \frac{d_{i,k}}{\sum\limits_{n_j \in CN_i} d_{i,j}}. \tag{23}$$

Obviously, $0 \leq R_{i,k} \leq 1$. The larger the $R_{i,k}$, the larger the distance from node n_k to node n_i. To some extent, $R_{i,k}$ describes the distance feature from n_i to n_k. When $R_{i,k}$ is 0, n_k and n_i are the same node. When $R_{i,k}$ is 1, there is a single candidate next node for node n_i. The function $\log_{10}^{(\cdot)}$ in Eq. (19) is used to smooth $\varepsilon_{i,k}$ and $\ell_{i,k}$ to avoid the fluctuation of $p_{i,k}$.

3.5 Routing Policy for Bypassing Node Voids

A node void may be formed by a cluster of nodes whose energy is depleted, or whose components are physically destroyed. In addition, a node void may also be generated by some obstacles of radio signals, or blocks where no node exists in the forwarding region of a node. For instance, the black nodes illustrated in Fig. 3 are some dead nodes, and they form a node void.

In general, owing to the significant non-uniform deployment of nodes in a non-uniform WSN, the number of node voids is much more than that in a uniform WSN, and the size of each node void is also larger than that in a uniform WSN. How to bypass node voids is very important for increasing the success rate of finding routing paths, reducing the length of routing paths, or avoiding a cycle routing path in non-uniform WSNs. Although using the right (left) hand rule in the GPSR algorithm can effectively bypass most node voids in uniform WSNs, it is usually inefficient in bypassing node voids in non-uniform WSNs. Because a node void in a non-uniform WSN is generally larger than the one in a uniform WSN due to the non-uniform deployment of nodes, the single side bypassing void mechanism of the right (left) hand rule may result in a cycle path or a long path. To address this issue, we propose a backward routing policy, and it is similar to the one in the TPGF algorithm, but they are not completely common. Suppose the current node is n_j, and due to the node void, the next hop node of n_j cannot be found. At this moment, we first return to the previous-hop node of n_j, say node n_k in Fig. 3, and mark n_j as 'unavailable node.' Then, another next hop node of n_k is selected from the set $CN_k - \{n_j\}$. If there is no forwarding neighbor node of n_k to select, the routing recursively returns to the previous-hop node of n_k, and so on. When this backward routing process finally reaches the source node and there is no other forwarding neighbor node of the source node to select, the routing fails. The node marking mechanism can effectively prevent other nodes to select node n_j as their next hop node. In fact, our bypassing node void policy combines the greedy forward exploring policy and the greedy backward routing policy, and there is no single side routing. Moreover, the greedy forward exploring mechanism is used prior to the greedy backward routing mechanism in each node in our bypassing node void routing, thus our routing policy cannot result in a cycle path.

Algorithm 1 PS

```
Input: Probabilities for selecting individual candidate next hop node as
    the next hop node of $n_j$: $p_1, p_2, \cdots, p_m$, where $m$ is the number of the
    candidate next hop nodes of $n_j$, i.e., $m = |CN_j|$.
Output: The next hop node of $n_j$
Step 1: Divide the interval [0,1] into $m$ subintervals by $(m+1)$ variables
        $q_0, \cdots, q_m$, such that $q_0 = 0$ and $q_i = q_{i-1} + p_i$.
Step 2: Produce a random number $\xi$, such that $0 \leq \xi \leq 1$.
Step 3: If $q_{i-1} \leq \xi \leq q_i$, then node $n_i$ is selected as the next hop node of
        $n_j$, where $n_i$ is the candidate next node with the selection probability $p_i$.
```

3.6 The NADRR Algorithm

Based on the probability selection mechanism in Sect. 3.4, the probability algorithm for selecting the next hop node of the current node can be summarized in Algorithm 1.

In the following, we discuss the termination and time complexity of Algorithm 1.

Theorem 1: Algorithm 1 is terminable and its time complexity is $O(m)$, where m is the number of the candidate next hop nodes of the current node n_j, that is, $m = |CN_j|$.

Proof. In Algorithm 1, the time complexity of Step 1 is $O(m)$, the time complexity of Step 2 is usually a constant, that is, $O(1)$. The time complexity of Step 3 is $O(m)$. Since m is finite, Algorithm 1 is terminable and its time complexity is $O(m)$.

Next, we discuss the optimization problem of a routing path. Suppose that pa is a routing path from the source node n_i to the sink node n_s. We represent the path pa by an ordered set of nodes $pa = \{n_i, n_k, n_h, \cdots, n_s\}$. The objective to optimize pa is to reduce the number of nodes included in pa. Our idea is to remove node n_k from the routing path pa if n_i and n_k are two neighbor nodes, n_k and n_h are two neighbor nodes, and n_i and n_h are also two neighbor nodes. As a result, pa becomes $pa' = \{n_i, n_h, \cdots, n_s\}$. Repeat this procedure until there are no three mutually neighboring nodes

Algorithm 2 RPO

```
Input: A given routing path $pa = \{n_i, n_k, n_h, \cdots, n_s\}$
Output: The optimized routing path $pa$
Step 1: Map $pa$ to an ordered set $pa' = \{u_1, u_2, u_3, \cdots, u_{|pa|}\}$, such that
        $n_i \mapsto u_1, n_k \mapsto u_2, n_h \mapsto u_3, \cdots, n_s \mapsto u_{|pa|}$, and then let $pa = \emptyset$.
Step 2: For $pa' = \{u_1, u_2, u_3, \cdots, u_{|pa|}\}$, if $u_i$ is the neighbor of $u_{i+1}$, $u_{i+1}$
        is the neighbor of $u_{i+2}$ and $u_{i+2}$ is the neighbor of $u_i$, then $u_{i+1}$ is
        removed from $pa'$, where $1 \leq i \leq |pa'|$.
Step 3: Repeat Step 2 until there are not three directly neighboring nodes
        in the routing path $pa'$.
Step 4: Map $pa'$ to $pa$ in terms of the converse mapping relationships of
        the mapping relationships in Step 1, then the new path $pa$ is the
        optimized routing path from $n_i$ to $n_s$, and output the routing path
        $pa$.
```

Algorithm 3 NADRR

```
Input: A source node $n_i$, the sink node $n_s$, communication radius of
nodes $r$, weights $w_1$ and $w_2$
Output: An optimum routing path from node $n_i$ to node $n_s$
Step 1: Denote the current node as $n_j$, then calculate the distance from
        $n_j$ to $n_s$, that is, $d_{j,s}$,by using Eq. (1).
Step 2: If $d_{j,s} \leq r$, select node $n_s$ as the next hop node of $n_j$, halt;
        otherwise, go to Step 3.
Step 3: Calculate the forwarding angle $\alpha_j$ of $n_j$ by Eqs. (1)–(17).
Step 4: Calculate the set $CN_j$ of the candidate next hop nodes of $n_j$ by
        Eq. (18), go to Step 5.
Step 5: If $CN_j = \emptyset$, that is, there is no candidate next hop node of $n_j$,
        return to the last-hop node of $n_j$. Suppose the last-hop node of
        $n_j$ is $n_{j-1}$. Mark node $n_j$ as "unavailable node," and make node
        $n_{j-1}$ become the current node $n_j$, go to Step 1. Otherwise, go to
        Step 6.
Step 6: For each node $n_k$ in $CN_j$, calculate the selected probability $p_{j,k}$
        by Eqs. (19)–(23), go to Step 7.
Step 7: Execute Algorithm 1 to select the next hop node of $n_j$, and make
        the next hop node of $n_j$ become the current node, go to Step 8.
Step 8: Repeat Steps 1–7 until a routing path $pa$ is obtained or routing
        fails, and then go to Step 9.
Step 9: If the routing search is successful, execute Algorithm 2 to opti-
        mize the routing path $pa$. Otherwise, the routing fails.
```

included in the routing path. The algorithm optimizing a routing path can be summarized in Algorithm 2.

As for Algorithm 2, we have the following conclusion.

Theorem 2: Algorithm 2 is terminable and its time complexity is $O(m)$, where m is the number of nodes included in a routing path pa to be optimized, that is, $m = |pa|$.

Proof. Since m is finite, it is obvious that Algorithm 2 is terminable. The time complexity of Step 1 is $O(m)$, the time complexity of Steps 2 and 3 is $O(m)$, and the time complexity of Step 4 is also $O(m)$. Thus the time complexity of Algorithm 2 is $O(m)$.

By combining the bypassing node void policy, the probability selection algorithm and the routing optimization algorithm together, the node aggregation degree-aware random routing (NADRR) algorithm for non-uniform WSNs can be summarized in Algorithm 3.

We have the following conclusion derived from Theorems 1 and 2.

Theorem 3: Algorithm 3 is terminable and its time complexity is $O(m)$, where m is the number of nodes in a non-uniform WSN.

4. Simulation and Analysis

We conduct simulations to validate the performance of the NADRR algorithm and compare it with the classical node geographic position based routing algorithms, such as GPSR, TPGF and CR. The performance metrics to be considered include energy consumption of nodes, balance of energy consumptions of nodes, average time latency of packet transmission, average time of searching a routing path, network lifetime, and packet transmission success rate.

4.1 Simulation Environment and Assumptions

The six non-uniform WSNs' topologies used in our simulations are shown in Fig. 4, respectively. For simplicity, we

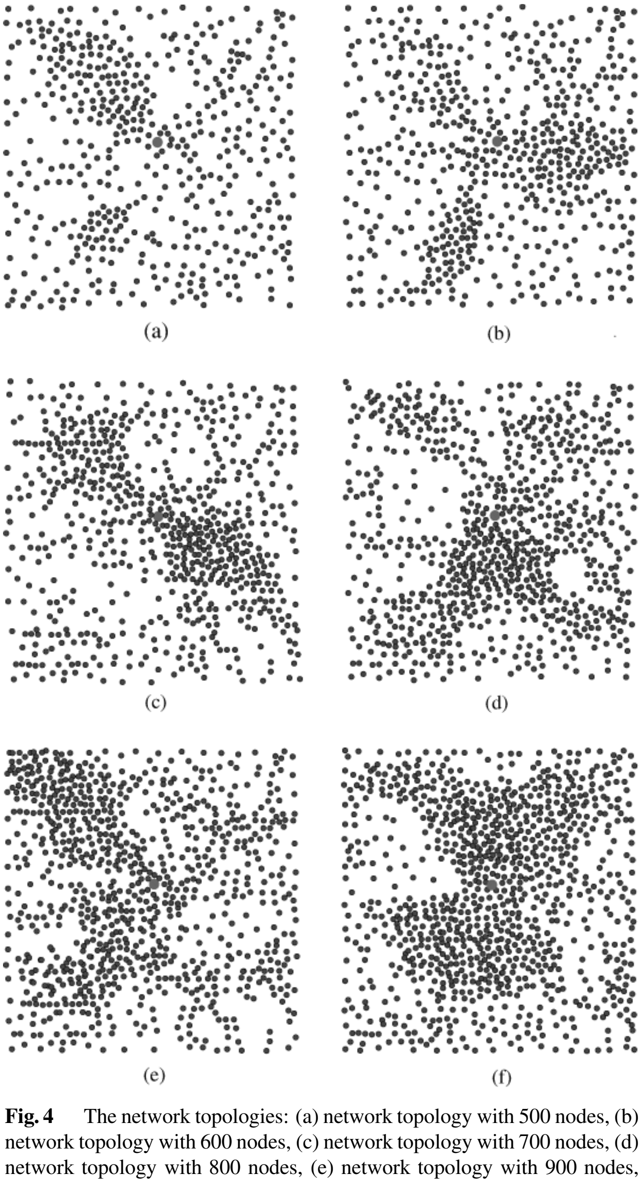

Fig. 4 The network topologies: (a) network topology with 500 nodes, (b) network topology with 600 nodes, (c) network topology with 700 nodes, (d) network topology with 800 nodes, (e) network topology with 900 nodes, and (f) network topology with 1000 nodes.

assume there is only one sink node n_s, denoted as a larger node in Fig. 4. We also assume that each non-unifor mWSN has an area of 600×600 (m^2), in which 500–1000 nodes are randomly and non-unoformly deployed, respectively. The wireless communication radius of each node is set to 50 (m), the initial energy of each node is set to 10 (J), and the size of each packet is 5000 (bit). The geographic position coordinates of each node are randomly generated, but the geographic position of the sink node is always chosen as the central point of the network.

We run each algorithm 200 times so as to get the average results. The simulation platform is the NetTopo [7], an open source simulation tool for the routing of WSNs. In the following simulations, we suppose, for simplicity, that there is no interference when several source nodes send packets to the sink node simultaneously. In fact, Parissidis (2008) and Qiu (2007) studied the problem of radio interferences in wireless multihop networks, and proposed some models. However, those models are too complex to adapt to the situations in non-uniform WSNs.

4.2 Energy Consumption of Nodes

Energy consumption of nodes refers to the energy consumed by the nodes in the packet transmission process. In our simulations, when a node n_i transmits a packet of η bits to another node n_k, the consumed energy of n_i, denoted as $E_S(n_i)$, is defined by

$$E_S(n_i) = E_{elec} \cdot \eta + \rho_{amp} \cdot \eta \cdot d_{i,k}^{\mu}. \tag{24}$$

where, E_{elec} is the consumed energy for node n_i to send each bit of information, and $E_{elec} = 50 \times 10^{-9}$ J/bit; ρ_{amp} is the consumed energy that the amplifier of n_i amplifies each bit of information, and $\rho_{amp} = 100 \times 10^{-12}$ J/(bit/m^2); $d_{i,k}$ is the distance between nodes n_i and n_k, and $d_{i,k}$ is calculated by Eq. (1); μ is the factor of signal attenuation, and we usually have $2 \leq \mu \leq 5$. When the communication environment is better, the value of μ is smaller, such as $\mu = 2$. Otherwise, the value of μ is larger, such as $3 \leq \mu \leq 5$. In our simulations, $\mu = 3$. η is the length of each packet, and its unit is bit.

Similarly, when a node n_k receives a packet of η bits, the consumed energy of n_k, denoted as $E_R(n_k)$, is defined as

$$E_R(n_k) = E_{elec} \cdot \eta. \tag{25}$$

Unlike a sender, a receiver does not need to amplify signals, thus it consumes less energy. For different network topologies, we run each algorithm 200 times, and 200 source nodes are randomly generated in one simulation. Each source node generates four packets per second, and the size of each packet is 5000 (bit). To be fare in comparing the energy consumption of nodes when using the algorithms of GPSR, TPGF, CR and NADRR, we suppose $w_1 = 0.5$ and $w_2 = 0.5$. This means that there is no partiality to saving the energy of nodes and reducing the time delay of packet transmission when using the NADRR algorithm. The simulation duration is 30 (min). For each routing algorithm, the average energy consumption of nodes is shown in Fig. 5. As can be seen in Fig. 5, the energy consumption of GPSR and TPGF is almost the same, whereas the energy consumption of NADRR is less than both of them. The energy consumption of CR is the least. The main reason is that the neighbor node farthest to the current node is always selected as the next hop node of the current node when using GPSR and TPGF. However, the neighbor node farther to the current node is always selected as the next hop node of the current node when using NADRR. In CR, for each node, the routing policy is to select the neighbor node closest to the straight line segment connecting the source node and the sink node as the next hop node of the node. As a result, the length of the routing path found by CR is close to the length of the straight line segment from the source node to the sink node. From Eq. (24), we know that the energy consumption of a sender significantly increases as the distance between a sender and a receiver increases. Therefore, the simulation results are consistent with the theoretical analysis.

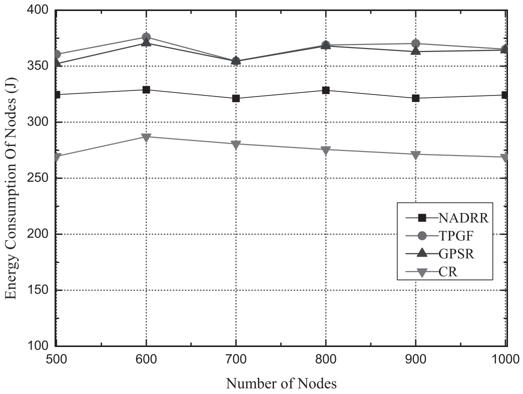

Fig. 5 The energy consumption of nodes.

4.3 Balance of Energy Consumptions of Nodes

The balance of energy consumption of nodes refers to the difference degree of the residual energy of all the nodes. Usually, the standard deviation of the residual energy of nodes may be used to measure the balance of energy consumption of nodes in a non-uniform WSN. The more balanced the residual energy of nodes, the longer the network lifetime. The standard deviation of the residual energy of nodes, denoted as $\hat{E}$, is defined by

$$\hat{E} = \sqrt{\sum_{i=1}^{N} (e_i - \bar{E})^2}, \tag{26}$$

where e_i is the current residual energy of n_i, N is the total number of nodes, $\bar{E}$ is the average value of the residual energy of all active nodes in a non-uniform WSN, and is defined by

$$\bar{E} = \sum_{i=1}^{N} e_i / \ N. \tag{27}$$

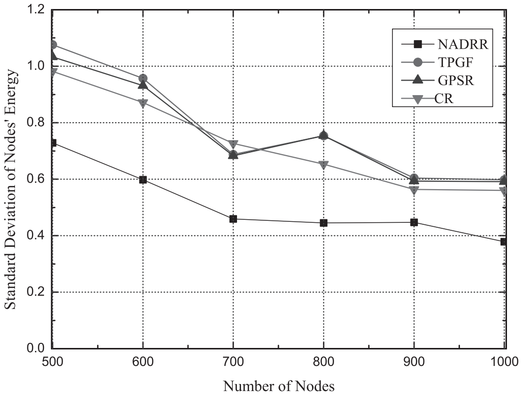

Fig. 6 Standard deviation of nodes' residual energy.

A larger $\hat{E}$ means that the energy distribution of nodes is not uniform. Conversely, a smaller $\hat{E}$ implies that the energy distribution of nodes is more uniform. Generally, node voids will not likely to occur when the energy distribution of nodes is uniform. For example, consider the six different network topologies in Fig. 4, the standard deviation of the residual energy of nodes is shown in Fig. 6 after the simulations are conducted for thirty minutes. To be fare in comparing the balance of energy consumption of nodes when using NADRR, we let $w_1 = 0.5$ and $w_2 = 0.5$. Figure 6 reveals that the standard deviation of the residual energy of nodes is the smallest when using NADRR, and the other three algorithms have similar standard deviations of the residual energy of nodes. The simulation results indicate that NADRR can achieve a better balance of energy consumption of nodes.

4.4 The Average Time Latency of Packet Transmission

The time latency of packet transmission refers to the spent time to transmit a packet from a given source node to the sink node, that is, end-to-end time delay. Suppose that the spent time for each node to receive, process and send a packet is nearly uniform. Since the time latency when a packet is in a wireless channel is much less than the time for a node to receive, process and send this packet, the time latency when the packet is transmitted in a wireless channel may be neglected. Furthermore, we can use the number of nodes included in a routing path to estimate the time latency of packet transmission. For the six non-uniform WSNs with different network topologies in Fig. 4, we run each algorithm 200 times, and the average numbers of nodes included in a routing path found by GPSR, TPGF, CR and NADRR are shown in Fig. 7, respectively. From Fig. 7, we notice that the average number of nodes included in a routing path found by GPSR is nearly equal to the average number of nodes included in a routing path found by TPGF. The average numbers of nodes included in a routing paths found by NADRR or CR are about one node more than that of a routing path found by GPSR or TPGF.

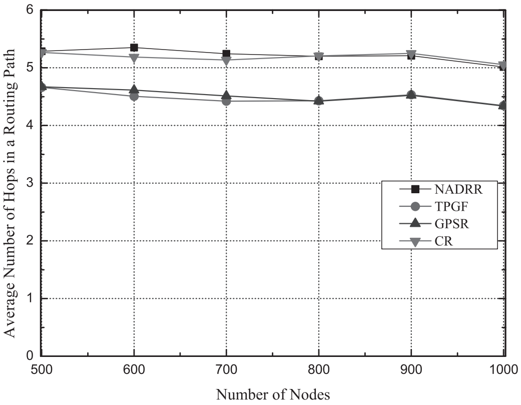

Fig. 7 The average number of nodes included in a routing path.

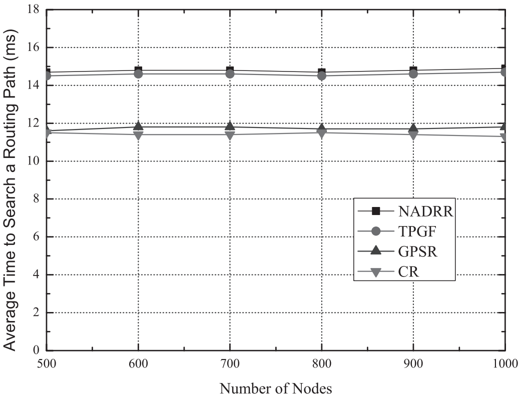

Fig. 8 The average time of searching a routing path.

4.5 The Average Time of Searching a Routing Path

To evaluate the real-time performance of a routing algorithm for non-uniform WSNs, we need to consider the time needed for searching a routing path from a source node to the sink node and the time needed for transmitting packets along a routing path. The later is measured by the number of nodes included in a routing path. To measure the former, we finished 4800 simulations with different network topologies by using the four routing algorithms, respectively. The average time needed for searching a routing path is shown in Fig. 8. We notice that the average time for NADRR or TPGF to find a routing path is about between 15 and 16 milliseconds, and the average time for GPSR or CR to find a routing path is about between 11 and 12 milliseconds. In contrast, the average time for NADRR or TPGF is slightly larger than that of GPSR or CR. The main reason, we believe, is that both NADRR and TPGF have a higher computation complexity than GPSR or CR. Moreover, consider the situation in Fig. 7, we notice that the average number of nodes included in a routing path found by NADRR or CR is slightly larger than that of GPSR or TPGF. Thus, these four routing algorithms are almost appropriate in the real-time performance. However, it is worth mentioning that NADRR focuses on

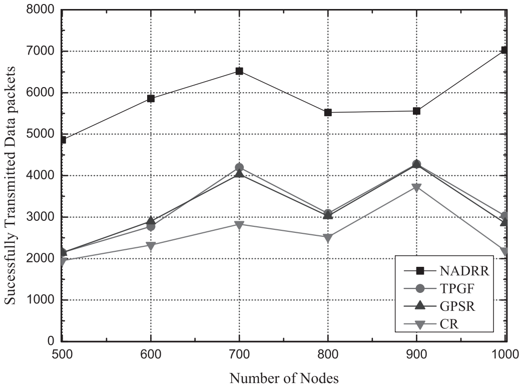

Fig. 9 The network lifetime.

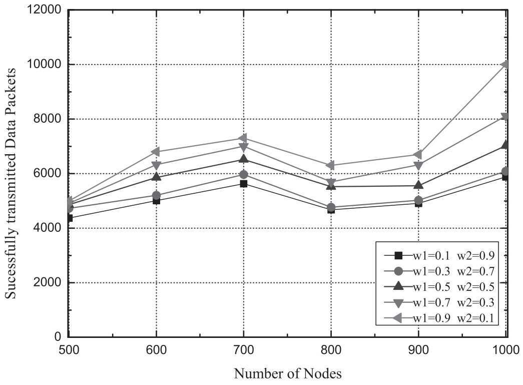

Fig. 10 Sensitivity of network lifetime to residual energy weight.

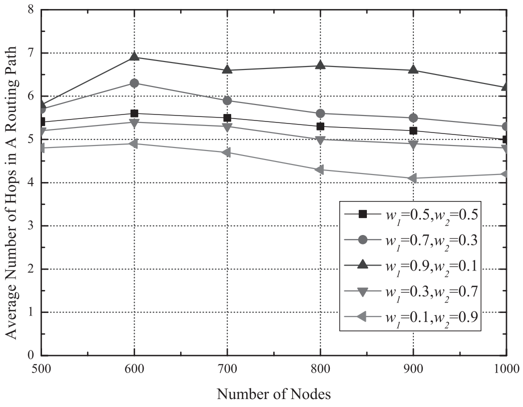

Fig. 11 Sensitivity of average hops of a routing path to distance weight.

balancing the energy consumption of nodes, prolonging network lifetime, and increasing the packet transmission rate in non-uniform WSNs.

4.6 The Network Lifetime

For different WSN applications, the definition of network lifetime is different [17], [18]. Unlike other definitions of network lifetime, in our simulations we regard the number of packets successfully transmitted from source nodes to the sink node as the lifetime of a WSN before the first dead node appears. Our basic idea is to randomly select a source node, and make the selected node generate a packet of 5000 bits, then send this packet toward the sink node. Randomly select such source nodes to send packets toward the sink node until the first dead node appears. At this time, the total number of packets received by the sink node is regarded as the network lifetime. Due to the randomness of selecting source nodes, the more packets received by the sink node, the longer the network lifetime. So, the above definition of network lifetime is reasonable. When using GPSR, TPGF, NADRR and CR, the maximum numbers of successfully transmitted packets are shown in Fig. 9, respectively. In the simulations, we still let $w_1 = 0.5$ and $w_2 = 0.5$. Figure 9 shows that NADRR does a better job in prolonging network lifetime than GPSR, TPGF and CR. The main reason is that NADRR can better balance the energy consumption of nodes.

4.7 Sensitivity of the NADRR Algorithm to Weights

In Eq. (19), w_1 is the weight of balancing the energy consumption of nodes, and w_2 is the weight of reducing the time latency of packet transmission. We use network lifetime to reflect the sensitivity of NADRR to weight w_1, and use the average hops of a routing path to measure the sensitivity of NADRR to weight w_2. Theoretically, when $w_1 > w_2$, NADRR cares more for balancing the energy conservation of nodes, thus the network should have a longer lifetime. In contrast, when $w_2 > w_1$, NADRR prefers more to reducing the time latency of packet transmission, thus each routing path should have fewer number of hops. We run NADRR with five groups of w_1 and w_2 on the network topologies shown in Fig. 4, and the results are shown in Fig. 10 and Fig. 11. From Fig. 10, we notice that as w_1 increases or w_2 decreases, the network lifetime increases. Figure 11 shows that the average number of nodes involved in a routing path decreases as w_1 decreases or w_2 increases. The simulation results are consistent with the theoretical analysis.

4.8 Packet Transmission Success Rate

The packet transmission success rate refers to the ratio of the number of packets received by the sink node to the total number of packets sent by source nodes. In a certain period of time, suppose that the source nodes send Q packets, and the sink node successfully receives W packets. The packet transmission success rate, denoted by σ, is defined as

$$\sigma = \frac{W}{Q}. \tag{28}$$

Obviously, a larger σ means more packets successfully reach the sink node from source nodes, and this just is what users or applications hope. For each network topology in Fig. 4, we randomly select 100 source nodes, each source node generates 200 packets, and 20000 packets are transmitted toward the sink node in each simulation. To objectively estimate the value of σ, for an individual algorithm we conducted 200 simulations to obtain the average packet transmission success rate, and the simulation results are shown

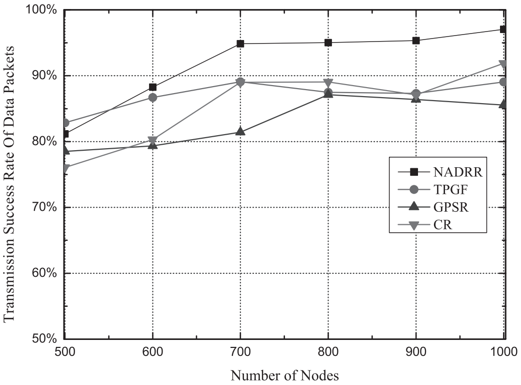

Fig. 12 Packet transmission success rate.

in Fig. 12. Figure 12 shows that NADRR has a higher transmission success rate than TPGF, GPSR and CR. For a high density non-uniform WSN, the packet transmission success rate of NADRR is nearly 100 percent. The main reasons are that the energy consumption of nodes is better balanced and the back routing policy is used in NADRR. Furthermore, NADRR can significantly decrease the occurrence rate of node voids, and efficiently bypass node voids which have occurred.

5. Conclusions

The routing for WSNs has been a popular research issue for a long time. However, limited attentions have been paid to WSNs with non-uniform deployment of nodes. In this paper, we proposed a novel routing scheme for non-uniform WSNs based on the concepts of Local Vertical Aggregation Degree (LVAD) and Local Horizontal Aggregation Degree (LHAD). In the proposed routing scheme, the search scope for a node to select its next-hop node is efficiently limited to a size-proper forwarding region based on the LVAD and the LHAD of the node. On this basis, we take into consideration both the residual energy and the geographic position information of nodes to design a node aggregation degree-aware random routing (NADRR) algorithm for non-uniform WSNs. Simulation results show that NADRR outperforms GPSR, TPGF and CR in terms of energy consumption of nodes, balance of energy consumption of nodes, network lifetime and packet transmission success rate. It is an interesting topic to improve the NADRR algorithm to adapt to the non-uniform WSN routing considering wireless communication interferences.

Acknowledgments

The authors would like to thank the anonymous reviewers and the editors of this paper for their valuable revision suggestions.

References

[1] J. Yick, B. Mukherjee, and D. Ghosal, “Wireless sensor network survey,” Comput. Netw., vol.52, no.12, pp.2292–2330, 2008.

[2] M. Mauve, J. Widmer, and H. Hartenstein, “A survey on geographic position-based routing in mobile ad hoc networks,” IEEE Netw., vol.15, no.6, pp.30–39, 2001.

[3] T.F. Shih and H.C. Yen, “Geographic position-aware routing algorithm with dynamic adaptation of request zone for mobile ad hoc networks,” Wirel. Netw., vol.14, no.3, pp.321–333, 2008.

[4] R. Xiao and G. Wu, “A survey on routing in wireless sensor networks,” Progress in Natural Science, vol.17, no.3, pp.261–269, 2007.

[5] Y.B. Ko and N.H. Vaidya, “Location-aided routing in mobile ad hoc networks,” Wirel. Netw., vol.6, no.4, pp.307–321, 2000.

[6] B. Karp and H. Kung, “Greedy perimeter stateless routing for wireless networks,” MOBICOM 2000: Proc. 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking, pp.243–254, ACM, New York, NY, USA, 2000.

[7] L. Shu, Z. Zhou, M. Hauswirth, and G. Hynes, “Transmitting and gathering streaming data in wireless multimedia sensor networks within expected network lifetime,” Mobile Netw. Appl., vol.13, no.3-4, pp.306–322, 2008.

[8] E. Kranakis, H. Singh, and J. Urrutia, “Compass routing on geometric networks,” Proc. 11th Canadian Conference on Computational Geometry, pp.51–54, Vancouver, Canada, Aug. 1999.

[9] J. Lian, K. Naik, and G. Agnew, “Data capacity improvement of wireless sensor networks using non-uniform sensor distribution,” Internal Journal of Distributed Sensor Networks, vol.2, no.2, pp.121–145, 2006.

[10] D. Chena and P.K. Varshney, “On-demand geographic forwarding for Data delivery in wireless sensor networks,” Comput. Commun., vol.30, no.14-15, pp.2954–2967, 2007.

[11] J.N. Al-Karaki and A.E. Kamal, “Routing techniques in wireless sensor networks: A survey,” IEEE Wirel. Commun., vol.11, no.6, pp.6–28, 2004.

[12] M. Bandai, T. Mioki, and T. Watanabe, “A routing protocol with stepwise interest retransmission for wireless sensor networks,” IEICE Trans. Commun., vol.E91-B, no.5, pp.1446–1453, May 2008.

[13] K. Matrouk and B. Landfeldt, “RETT-gen: A globally efficient routing protocol for wireless sensor networks by equalising sensor energy and avoiding energy holes,” Ad Hoc Networks, vol.7, no.3, pp.514–536, 2009.

[14] F. Yu, S. Park, E. Lee, Y. Choi, and S.H. Kim, “QSLS: Efficient quorum based sink location service for geographic routing in irregular wireless sensor networks,” IEICE Trans. Commun., vol.E92-B, no.12, pp.3935–3938, Dec. 2009.

[15] W. Qiu, A.E. Skafidas, and P. Haoa, “Enhanced tree routing for wireless sensor networks,” Ad Hoc Networks, vol.7, no.3, pp.638–650, 2009.

[16] H. Shokrzadeh, A.T. Haghighat, and A. Nayebi, “New routing framework based on rumor routing in wireless sensor networks,” Comput. Commun., vol.32, no.1, pp.86–93, 2009.

[17] G. Anastasi, M. Conti, M.D. Francesco, and A. Passarella, “Energy conservation in wireless sensor networks: A survey,” Ad Hoc Networks, vol.7, no.3, pp.537–568, 2009.

[18] I. Dietrich and F. Dressler, “On the lifetime of wireless sensor networks,” ACM Trans. Sensor Networks, vol.5, no.1, pp.1–38, 2009.

Appendix: Proof of Proposition 1

From Eq. (11) and Eq. (12), we have $0 \le L_i \le r$ and $0 \le T_i \le r$. Thus, we have

$$-1 \le \frac{T_i}{r} - \frac{L_i}{r} < 1. \qquad \text{(A· 1)}$$

From Eq. (A· 1), we have

$$0 \le 1 + \frac{T_i}{r} - \frac{L_i}{r} < 2. \qquad (\mathrm{A}\cdot 2)$$

Then

$$\log^{r}_{(1+|N_i|+r)} - \log^{|N_i|}_{(1+|N_i|+r)} = \log^{\frac{r}{|N_i|}}_{(1+|N_i|+r)}. \qquad (\mathrm{A}\cdot 3)$$

From Eq. (3), we have $|N_i| \ge 1$. If $|N_i| = 1$, $\log^{\frac{r}{|N_i|}}_{(1+|N_i|+r)} = \log^{r}_{(2+r)}$. In a WSN, usually $r \ge 1$. Since $r < 2 + r$ and $1 < 2 + r$, we have

$$0 \le \log^{r}_{(2+r)} < 1 \qquad (\mathrm{A}\cdot 4)$$

Next, we need to prove the following equation holds for $|N_i| > 1$.

$$\log^{\frac{r}{|N_i|}}_{(1+|N_i|+r)} < 1. \qquad (\mathrm{A}\cdot 5)$$

Since $(1+|N_i|+|r|) > 1$, we only need to prove the following Equation always holds.

$$\frac{r}{|N_i|} < 1 + |N_i| + r. \qquad (\mathrm{A}\cdot 6)$$

Since

$$\begin{aligned}(1 + |N_i| + r) - \frac{r}{N_i} &= \frac{|N_i| + |N_i|^2 + (|N_i| - 1)\cdot r}{|N_i|} \\ &= \frac{|N_i|\cdot(|N_i| - 1) + 2\cdot|N_i| + (|N_i| - 1)\cdot r}{|N_i|} \\ &= \frac{(|N_i| - 1)\cdot(|N_i| + |r|) + 2\cdot|N_i|}{|N_i|}, \qquad (\mathrm{A}\cdot 7)\end{aligned}$$

when $|N_i| > 1$, we have

$$\frac{(|N_i| - 1)\cdot(|N_i| + |r|) + 2\cdot|N_i|}{|N_i|} > 0. \qquad (\mathrm{A}\cdot 8)$$

Since $|N_i| > 1$, Eq. (A· 8) always holds. Therefore, Eq. (A· 5) holds. From Eq. (A· 2) and Eq. (A· 5), we have

$$1 + \frac{T_i}{r} - \frac{L_i}{r} + \log^{\frac{r}{|N_i|}}_{(1+|N_i|+r)} < 3. \qquad (\mathrm{A}\cdot 9)$$

Therefore,

$$\frac{\pi}{3}\left(1 + \frac{T_i}{r} - \frac{L_i}{r} + \log^{\frac{r}{|N_i|}}_{(1+|N_i|+r)}\right) < \pi. \qquad (\mathrm{A}\cdot 10)$$

From Eq. (14), Eq. (A· 4) and Eq. (A· 10), we have $0 \le \alpha_i < \pi$.

Xiaoming Wang received the B.S. and the Ph.D. degrees in computer theory and software from Shaanxi Normal University, and Northwest University, China, in 1989 and 2005. He is a professor of the School of Computer Science at Shaanxi Normal University. Dr. Wang was a research professor at Georgia State University, USA, from 2007 to 2008. His research interests include wireless sensor network, and pervasive computing. He is a senior member of CCF.

Xiaohong Jiang received the B.S., the M.S. and the Ph.D. degrees from Xidian University, China, in 1989, 1992 and 1999. He is a full Professor at Future University, Japan. Before joining Future University, He was an associate professor at Tohoku University, an assistant professor at the Japan Advanced Institute of Science and Technology. His research interests include computer communications networks, and network coding. He is a Senior Member of IEEE.

TaoYang received the B.S., and the M.S. degrees in computer theory and software from East China Institute of Technology, and Shaanxi Normal University, China, in 2004 and 2010, respectively. His research interests include wireless sensor networks, and pervasive computing.

Qiaoliang Li received the B.S. and the M.S. degrees in Application Mathematics from Nankai University, China, in 1987 and 1990, and received the Ph.D. degree from University of Science and Technology of China, in 1997. He is a professor of the School of Computer Science at Hunan Normal University. Dr. Li was a research professor at Georgia State University, USA, in 2009. His research interests include wireless sensor network, and information security.

Yingshu Li received the B.S. degree in computer science from Beijing Institute of Technology, China, in 2001, the M.S. and the Ph.D. degrees in computer science from University of Minnesota at Twin Cities, USA, in 2003 and 2005, respectively. She is an associate professor of the Department of Computer Science at Georgia State University, USA. Her research interests include wireless networks,and optimization theory. She is a member of IEEE.