Energy-Balanced and Depth-Controlled Routing Protocol for Underwater Wireless Sensor Networks

Hao Qin, Zhiyong Zhang, Rui Wang, Xiaojun Cai, and Zhiping Jia(✉)

School of Computer Science and Technology, Shandong University, Jinan, China
`jzp@sdu.edu.cn`

Abstract. As the ocean exploration becomes more and more popular, the Underwater Wireless Sensor Network (UWSN) has recently received extensively attentions. In UWSN, a large number of nodes are deployed at different depths, which means once deployed, it will be difficult to replace or recharge due to the complex underwater environment. Therefore, improving the lifetime of the UWSN network is one of critical issue to be studied. Since the sensor nodes are distributed at different depths, the energy of the nodes near to the horizontal plane which have more data to forward will be exhausted more quickly. The unbalanced energy consumption leads to a decline in network lifetime. To address this problem, we propose an Energy-Balanced and Depth-Controlled Routing Protocol for Underwater Wireless Sensor Networks in this paper. The proposed protocol replaces the low-energy nodes with the high-energy nodes by adjusting their depths to achieve balanced energy consumption among the whole network. Experimental results show our scheme effectively improves the lifetime of the whole network.

1 Introduction

In recent years, UWSN has gained increasing popularity in both academia and industry area because people are interested in exploring the vast underwater environment. UWSN has a large number of applications such as tactical surveillance, seismic monitoring, assisted navigations, pollution monitoring and many more scientific based applications [1,2,17].

Although there are many routing protocols that are proposed for terrestrial Wireless Sensor Networks (WSNs), these existing routing protocols may not be suitable for underwater environment [3,18]. Radio signals have rapid attenuation in the water, which means UWSN has to use acoustic channels for communication. In the harsh underwater environment, the acoustic signals have unique characteristics such as long propagation delay (five orders of magnitudes slower than radio), high signal to noise ratio, low bandwidth etc. So designing routing protocol for UWSNs is very challenging [4].

In UWSN, the sensor nodes have limited battery power and replacing the batteries of all the nodes is very expensive and difficult. Hence, improving the network lifetime is one of the most important issues. In UWSN, the sensor nodes are distributed at different depths and have different number of data to forward.

S. Ibrahim et al. (Eds.): ICA3PP 2017, LNCS 10393, pp. 115–131, 2017.
DOI: 10.1007/978-3-319-65482-9_8

There is a problem that each node has different energy consumption. Due to the unbalanced energy consumption, some nodes with high load die earlier than other nodes, which influences the lifetime of UWSN. Therefore, the energy consumption balance among the sensor nodes is one major method to improve the network lifetime.

Some protocols are proposed to address this issue (e.g. EEDBR [6]). Although these protocols consider balancing the energy consumption and prolonging the network lifetime, they don't consider the energy balance between the nodes near the sink nodes and the nodes far from the sink nodes. The nodes close to the sink nodes have unbalanced load and forward more packets than the nodes far from the sink nodes. This results that the nodes close to the sink nodes die earlier. Due to the unbalanced energy consumption, the nodes far from the sink still have energy to work when the overall network is dead.

To address this problem, we propose Energy-Balanced and Depth-Controlled Routing Protocol (EBDCR) to improve the network lifetime. In EBDCR, we replace the low-energy nodes near the sink nodes by the high-energy nodes far from the sink nodes through depth adjustment. We decline the low-energy nodes and float up the high-energy nodes to make the high-energy nodes closer to the sink nodes. Doing like this, we ensure the nodes now near the sink nodes have more energy to forward data. Hence, the lifetime of the network can be prolonged.

The major contributions of this paper can be summarized as follows:

- We propose a strategy to prolong the lifetime of the network through depth adjustment. We decline the low-energy nodes and float up the high-energy nodes to achieve the balance of the energy consumption between the nodes near the sink nodes and the nodes far from the sink nodes.
- We put forward an algorithm to identify the time when the nodes should be replaced. And then we determine a strategy to ensure the data transmission thereby improving the network lifetime.

The remainder of this paper is organized as follows. Section 2 describes the related work. The proposed protocol is presented in Sect. 3. In Sect. 4, we present simulations results that we have conducted in order to evaluate the proposed protocol. Finally, Sect. 5 concludes this paper.

2 Related Work

Recently, a number of routing protocols have been proposed for UWSNs. In this section, we present some related routing protocols as follows.

In [7], the authors propose the VBF routing protocol which uses the distance between the node and the routing vector to determine whether it should forwards the data packet. The forwarding process of VBF can be seen as to build a routing pipe between the source node and the destination node so that packets are delivered through the nodes in the pipe. Moreover, VBF uses a self-adaption algorithm in order to reduce the number of forwarding nodes and conserve energy. In [19], CVBF divides all nodes into the number of predefined clusters, and selects one node at the top of each cluster as a virtual sink. The rest of nodes transmit the data packets to their respective cluster virtual sink following the methodology of VBF routing protocol. The cluster virtual sink node forwards the aggregated data to the main sink node deployed on water surface through single-hop mechanism.

In Depth-based routing (DBR) protocol [4], the decision of forwarding the packet is based on the node depth and the depth of the previous sender. If the node can forward the packet, it will wait a holding time. During the holding time, the node discards the packet when it receives the same packet. For efficient energy consumption, EEDBR [6] computes holding time on the bases of residual energy of sensor nodes to enhance the network lifetime. And in [15], DSEEDBR provides enhanced network lifetime along with delay sensitivity to EEDBR by implementing Delay-Sensitive Holding time (DSH_t) and adaptive variations in d_{th} for sensor nodes. AMCTD [13] encourages the deployment of courier nodes and devises efficient weight functions to increase the stability period of the network.

Depth-Controlled Routing protocol (DCR) [5] adjusts the depths of some nodes in order to organize the network topology and forward data when the problem of communication void arises. DCR provides a centralized algorithm to identify the nodes which are disconnected and the nodes which are the void nodes, and then calculates the new depths of these nodes to improve both the network connectivity and rate data delivery.

Amara et al. [14] propose DEADS to improve reliability and efficiency. In DEADS, the authors propose DS (the Dominating Set) based cooperative routing algorithm with sink mobility. They discuss two mobility pattern of mobile sink: elliptical mobility pattern and linear mobility pattern. And DEADS works in three phases: neighbor selection, DS and CC set formation, and threshold based data sensing and routing.

In [16], EBECRP avoids depth base routing and uses mobile sinks to balance load on all nodes. It uses the concept of clustering to reduce multi hoping which results in more energy consumption. The selected Cluster Heads (CHs) collect data from one hope neighbor nodes to reduce global communication into locally compressed communication.

These routing protocols don't consider the energy balance between the nodes near the sink nodes and the nodes far from the sink nodes. Hence, the nodes near the sink nodes will die earlier because they have more data to forward. Inspired by the idea of adjusting the depth to improve the network connectivity and forwarding data, which is proposed in DCR [5], we propose Energy-Balanced and Depth-Controlled Routing Protocol (EBDCR) to improve the network lifetime by adjusting the depth of the sensor nodes.

3 Energy-Balanced and Depth-Controlled Routing Protocol

In this section, we present our Energy-Balanced and Depth-Controlled Routing Protocol in detail. Firstly we introduce the network architecture of EDBCR.

Secondly, we explain the phases of network initialization and data forwarding. And then we introduce our node replacement strategy to prolong the lifetime of the network and explain how our algorithm identifies which nodes should be replaced and which nodes can replace the low-energy nodes. At the end of the algorithm, we calculate their respective new depths and adjust them to new depths to continue forwarding data thereby prolonging the network lifetime.

3.1 Network Architecture

As shown in Fig. 1, UWSN consists of one or more sink nodes and lots of sensor nodes. The sink nodes are deployed on the surface of water with the help of the floating buoy or the anchor. The sink nodes are equipped with both acoustic and radio (e.g., Wi-Fi or Satellites) transceivers. These sink nodes use acoustic modem for communication with the sensor nodes to receive the data packets, while they can communicate with each other by radio links to forward the data packets collected from sensor nodes to the onshore data center or the research ship. We assume that all the sink nodes have enough energy because they can exchange the batteries expediently or they can utilize solar energy. So we don't consider the energy consumption of the sink nodes.

The sensor nodes are deployed underwater from the top to the bottom of the deployment region. They are equipped with a variety of sensors to sense the surrounding environment and they use acoustic modem to send the collected data towards the sink nodes. In this communication, the sensor node sends the data packet to its neighbor node which is selected as next-hop and then this neighbor node repeats this step. So the data packet is delivered to one of the sink nodes by multi-hops. Because the radio communication is much faster than

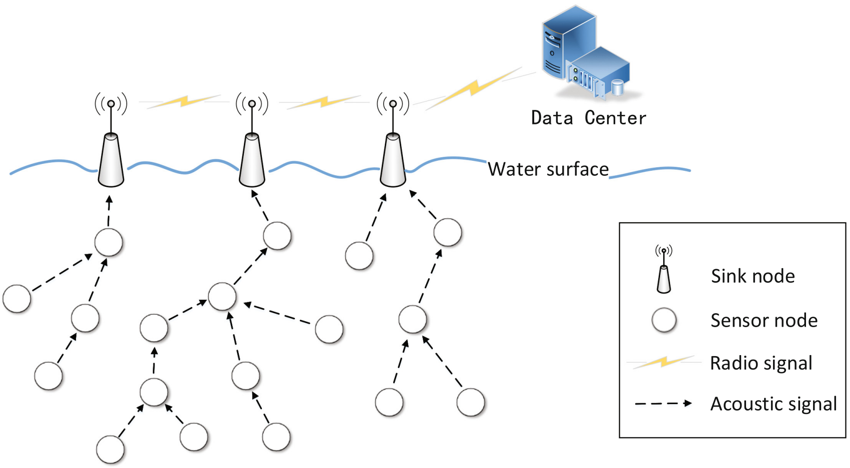


Fig. 1. Architecture of UWSN.

the acoustic communication, we assume that a packet is delivered successfully as long as it is received by one of the sink nodes.

In addition, all the sensor nodes can be fixed at a specific depth by the anchor, and we can adjust their depths by adjusting the length of the anchor chain (using winch-based module) [5]. We assume each sensor node has unique ID and has the same communication range which defined as R.

In this paper, we denote the sink node as S_i and the sensor nodes as N_i. The residual energy of the sensor node N_i is defined as E_i.

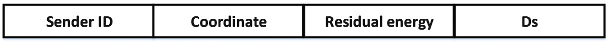
3.2 Network Initialization Phase

After the deployment of all the nodes, the network initialization phase begins. At first, all of the sink nodes obtain their locations by means of a positioning system like GPS. The coordinate of the sink node S_i is defined as $(X_{s_i}, Y_{s_i}, Z_{s_i})$. And then the sensor nodes use the AUV aided localization system [8] and the on-board pressure sensor to obtain their respective locations. During the AUV aided localization, the AUV broadcasts the locations of all the sink nodes which have been sent to the AUV before the localization. So each sensor node has its own coordinate denoted as (X_i, Y_i, Z_i) and all coordinates of the sink nodes. And then each sensor node uses the Algorithm 1 to calculate the Euclidean distance between itself and its nearest sink node. We define this distance as Ds. After that, the sensor node broadcasts its self-information packet to its neighbors periodically. The information packet contains the ID, coordinate, residual energy, and Ds. The format of the information packet is shown in Fig. 2a.

Algorithm 1. Ds Calculation

```
$(X_i, Y_i, Z_i)$ the coordinate of the sensor node $N_i$
$(X_{s_i}, Y_{s_i}, Z_{s_i})$ the coordinate of the sink node $S_i$
$Ds = infinite$
for all sink node $S_i$ do
    $D_{s_i} = \sqrt{(X_i - X_{s_i})^2 + (Y_i - Y_{s_i})^2 + (Z_i - Z_{s_i})^2}$
    if $D_{s_i} < Ds$ then
        $Ds = D_{s_i}$
    end if
end for
return $Ds$
```

When the sensor node receives the information packet from one of its neighbors, it compares its Ds with the neighbor's Ds. If the neighbor's Ds is smaller, it means that this neighbor is closer to the sink nodes than the sensor node, and the sensor node can forward the data packets to this neighbor to deliver the packets to the sink nodes. The sensor node will record this neighbor's information (such as ID, coordinate, residual energy) into the forwarding node candidate list. The format of the forwarding node candidate list is shown as Table 1. At the last of



(a) The information packet

(b) The data packet

Fig. 2. The format of packet

this phase, the sensor node sorts the forwarding node candidate list and selects the neighbor node which has smallest Ds from the forwarding node candidate list as the forwarding node(the next-hop node).

Table 1. The forwarding node candidate list.

Node ID	Coordinate	Ds	Residual energy (J)
1	$(10, 5, 7)$	7	90
18	$(10, 15, 27)$	30	80

3.3 Data Forwarding Phase

After network initialization, all the sensor nodes sense the surrounding environment and collect data of interest from the environment. Then it comes to data forwarding phase and sensor nodes start to send data packets. In network initialization phase, each sensor node uses greedy forwarding strategy to select the neighbor node which has smallest Ds from the forwarding node candidate list as the forwarding node (the next-hop node). After that, in order to balance the energy consumption of the neighbor nodes in the forwarding node candidate list, each sensor node periodically checks the residual energy of the neighbor nodes in the forwarding node candidate list and selects the node which has largest residual energy as its forwarding node. When the sensor node needs to send a data packet towards sink nodes, it just sends the packet to its forwarding node. The forwarding node (the next-hop node) receives the data packet and sends to its forwarding node. Finally the data packet will be delivered to one of the sink nodes by multi-hops.

As shown in Fig. 2b, the data packet includes five parts: sender ID, forwarder ID, sender's coordinate, sender's residual energy and the data. When a sensor node sends a data packet, all its neighbors can receive the packet in UWSN. In order to avoid redundant packets and save energy, the sensor node checks the forwarder ID when it receives a packet. If its ID is same with the forwarder ID, it will forward this packet. At the same time, the sensor node adds the sender's information into the child node list. The format of the child node candidate list is shown as Table 2. The sender ID, coordinate and the residual energy can obtain

Table 2. The child node candidate list.

Sender ID	Coordinate	P_s	Residual energy (J)
3	$(12, 20, 7)$	7	90
10	$(10, 15, 27)$	10	80

from the data packet directly. And the packet flow sent by the sender (defined as P_s) can be obtained by counting the number of packets received from the sender per unit time.

3.4 Node Replacement Strategy and Algorithm

Because during data forwarding phase each sensor node needs to send its collected data and the data that its child nodes send to it, the sensor nodes close to the sink nodes have more load than other sensor nodes far from the sink nodes. So the sensor nodes close to the sink nodes die earlier, which results that the sensor nodes far from the sink nodes still have energy to work when the network is dead. In order to overcome this problem and make full use of these residual energy to prolong the lifetime of the network, we propose our node replacement strategy.

The main idea of our proposed node replacement strategy is using the high-energy nodes which are far from the sink nodes to replace the low-energy nodes which are close to the sink nodes. Because each sensor node is equipped with the anchor and the length of the anchor chain can be adjusted, we can adjust the depths of the sensor nodes to achieve the node replacement. As shown in Fig. 3, the node replacement means that one sensor node (named B) replaces the function of another sensor node (named A) through depth adjustment. In other words, the child nodes of node A can send the data packets to node B, and then node B can forward these data packets to node A's forwarding node.

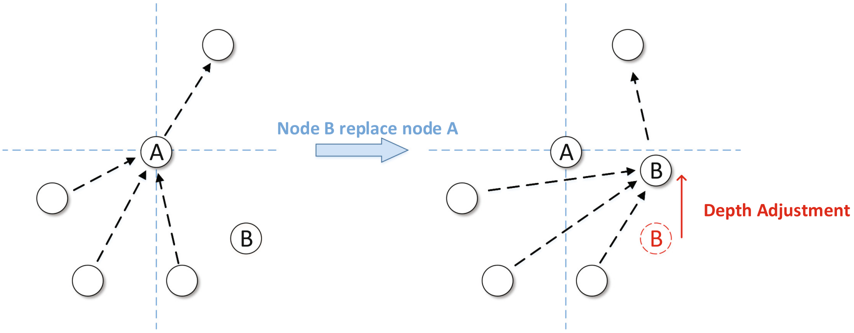


Fig. 3. The node replacement.

The node replacement strategy is divided into the following steps:

1. One of the sensor nodes meets node replacement condition which means its energy consumption is larger than a defined threshold.
2. The sensor node selects one of its child nodes that have more energy as the replacement node.
3. The replacement node adjusts its depth to replace the function of the sensor node. At the same time the sensor node adjusts its depth to replace the function of the replacement node.

Because the sink nodes are deployed on the water surface, the low-depth sensor nodes have more energy consumption than the high-depth sensor nodes. Through the node replacement, we decline the low-energy nodes and float up the high-energy nodes. So the high-energy nodes currently are closer to the sink nodes and forward more data packets than the low-energy nodes, this achieves the balance of the energy consumption between the nodes close to the sink nodes and the nodes far from the sink nodes. Hence, the lifetime of the network can be prolonged.

Node Replacement Condition. Since each sensor node is deployed at different locations and has different number of child nodes, each sensor node has different load. So each sensor node has different residual energy as time increases. In order to balance the energy of each sensor node quickly, the node replacement condition should meet the following requirements:

1. In the similar depth, the sensor node which has low energy should be replaced earlier than the node which has high energy.
2. In the similar depth, the sensor node which has large packet flow should be replaced earlier than the node which has small packet flow.
3. Since the sensor node which has high depth has fewer load, the energy consumption of high-depth sensor node is slow. If the node replacement condition is independent of the location of sensor node, the node with high depth will be replaced slower than the node with low depth. In order to balance the energy between the high-depth sensor nodes and the low-depth sensor nodes effectively and quickly, the sensor node with high depth should be replaced earlier than the node with low depth.

Therefore, the node replacement condition is that the energy consumption of the sensor node N_i is larger than threshold T_i. T_i is calculated as follows:

$$T_i = \frac{1}{x \cdot L_i} \cdot E_i \tag{1}$$

$$L_i = \lceil \frac{Ds}{R} \rceil \tag{2}$$

E_i is current residual energy of node N_i when T_i is calculated. Once the sensor node N_i adjusts its depth, T_i should be updated. x is a pre-defined positive constant and L_i is the ideal minimum number of hops for node N_i sending data to sink nodes. (L_i represents how far the node is from sink nodes.) It can be seen that the farther away from the sink nodes (the larger Ds), the larger L_i will be. So when two sensor nodes have same E_i and packet flow, the node with higher depth has smaller T_i, causing replacement earlier.

For example, we set $x = 2$, the sensor node A has 100 J initial energy, and its $Ds < R$. so $L_A = Ds/R = 1$ and its threshold $T_A = 1/(2 * 1) * 100\,\text{J} = 50\,\text{J}$. So when node A consumes 50 J of energy, it selects one of its child nodes as the replacement node and adjusts to new depth. After node replacement, we assume that the sensor node A consumes 10 J to adjust depth and its new $Ds \in (R, 2R)$. E_i is the current residual energy which is $50\,\text{J} - 10\,\text{J} = 40\,\text{J}$ and now its threshold $T'_A = 1/(2 * 2) * 40\,\text{J} = 10\,\text{J}$. Hence, the next replacement of node A occurs when it consumes 10 J.

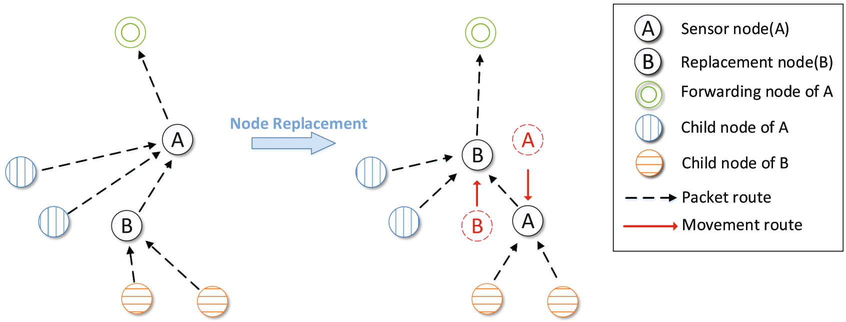


Fig. 4. The node replacement between node A and node B.

Replacement Node Selection. When the sensor node meets its node replacement condition, this means the energy consumption is large enough and the sensor node should select one of its child nodes as replacement node. Then the sensor node and the replacement node replace each other. As shown in Fig. 4, the node B is the replacement node of the node A. The sensor node A drops to an appropriate depth in order to receive the data packets from the child nodes of the replacement node B and forward them to the replacement node B. The replacement node B floats up to receive the data packets from the sensor node A's child nodes (except the replacement node B) and the data packets from the sensor node A. And then the replacement node B forwards them to the sensor node A's forwarding node.

Due to only the depth of the sensor node can be adjusted and the sensor nodes are different in horizontal position, it is difficult for the sensor node to find a child node which can reach the sensor node's position. When a child node is selected as the replacement node and adjusts its depth to replace the function of the sensor node, it is possible that some child nodes of the sensor node cannot communicate with the replacement node (the distance between the replacement node and the child node is larger R). As shown in Fig. 5, node F is the forwarding node of node A, node B,C,D are child nodes of A and node E,F are child nodes of B. The node communication radius is R. When node A selects node B as the replacement node, node B floats up to replace the function of node A. But node D is so far from node B that node B cannot communicate with node D. Similarly, the sensor node A needs drop to an appropriate depth to forward the data packets from the child nodes of node B to node B, but node A doesn't guarantee to communicate with all of node B's child nodes (node A cannot communicate with node E). After node replacement, node B only receives the packets from node A's child nodes which can communicate with node B and the packets that node A forwards. And the packets that node A forwards are from the child nodes of B that can communicate with A. Therefore, the number of packets forwarded by node B after the replacement is not more than the number of the packets forwarded by node A before the replacement. We define the number of packets forwarded by the replacement node (node B) per unit time after the replacement as reserved packet flow (P_{r}). In order to minimize the influence of node replacement on the other parts of the network, we should consider selecting the child node which can make the reserved packet flow (P_{r}) as large as possible as the replacement node.

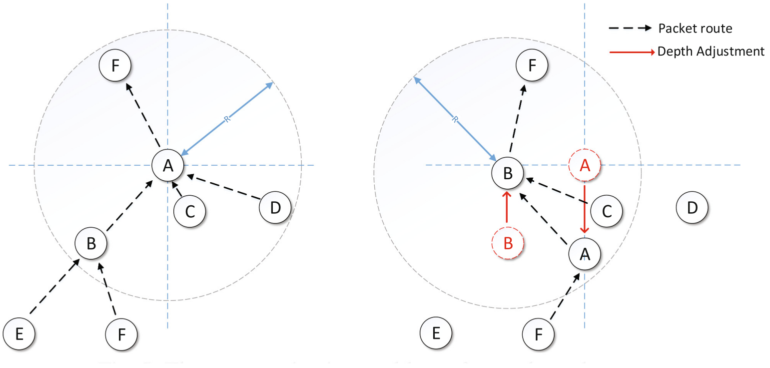


Fig. 5. The communication problem after node replacement.

In order to calculate the (P_{r}), the sensor node needs to know the information of its child nodes' child nodes, such as the coordinate and the packet flow (P_{s}). At the beginning of node selection, the sensor node sends request packet to all its child nodes and child nodes send the information recorded in their child node lists to the sensor node. After that the sensor node calculates the (P_{r}) of all child nodes which have more energy than it. Then it selects the replacement node according to the P_{r} and the residual energy. And during the calculation, the new depths of the sensor node and the replacement node are also calculated.

The Replacement Node Selection Algorithm: The notations used in our replacement node selection are shown in Table 3. As illustrated in Algorithm 2,

Algorithm 2. Replacement node selection.

```
1:  for all $c \in Child(n)$ do
2:    if $E(c) > E(n)$ then
3:      $P_{\mathrm{r}}(c) = P_{\mathrm{c}}(c)$
4:      $D(c) \leftarrow S(c,f) \leftarrow \left\{ z | \sqrt{(X_f - X_c)^2 + (Y_f - Y_c)^2 + (Z_f - z)^2} \le R \right\}$
5:      sort $Child(n)$ according the nodes' packet flow in descending order.
6:      for all $k \in Child(n) - \{c\}$ do
7:        $S(c,k) \leftarrow \left\{ z | \sqrt{(X_k - X_c)^2 + (Y_k - Y_c)^2 + (Z_k - z)^2} \le R \right\}$
8:        if $D(c) \cap S(c,k) \ne \emptyset$ then
9:          $D(c) \leftarrow D(c) \cap S(c,k)$
10:         $P_{\mathrm{r}}(c) = P_{\mathrm{r}}(c) + P_{\mathrm{s}}(k)$
11:       end if
12:     end for
13:     $Z'_c = z \quad (min|z - Z_c| \quad and \quad z \in D(c))$
14:     $P_{\mathrm{r}}(n) = P_{\mathrm{c}}(n)$
15:     $D(n) \leftarrow S(n,c') \leftarrow \left\{ z | \sqrt{(X_c - X_n)^2 + (Y_c - Y_n)^2 + (Z'_c - z)^2} \le R \right\}$
16:     sort $Child(c)$ according the nodes' packet flow in descending order.
17:     for all $j \in Child(c)$ do
18:       $S(n,j) \leftarrow \left\{ z | \sqrt{(X_j - X_n)^2 + (Y_j - Y_n)^2 + (Z_j - z)^2} \le R \right\}$
19:       if $D(n) \cap S(n,j) \ne \emptyset$ then
20:         $D(n) \leftarrow D(n) \cap S(n,j)$
21:         $P_{\mathrm{r}}(n) = P_{\mathrm{r}}(n) + P_{\mathrm{s}}(j)$
22:       end if
23:     end for
24:     $Z'_n = z \quad (min|z - Z_n| \quad and \quad z \in D(n))$
25:     $P_{\mathrm{r}}(c) = P_{\mathrm{r}}(c) + P_{\mathrm{r}}(n)$
26:     $E'(c) = E(c) - |Zc - Zc'| \cdot E_{\mathrm{a}}$
27:     $E'(n) = E(n) - |Zn - Zn'| \cdot E_{\mathrm{a}}$
28:     if $E'(c) > E(n)$ and $\frac{E'(n)}{P_{\mathrm{r}}(n) \cdot E_{\mathrm{f}}} \ge \frac{E(n)}{P_{\mathrm{s}}(n) \cdot E_{\mathrm{f}}}$ then
29:       add $c$ into $Candidates$
30:     end if
31:   end if
32: end for
33: select the child node whose $P_{\mathrm{r}}$ is largest as the replacement node from $Candidates$.
```

the sensor node n checks each its child node at first (line 1). If its child node c has smaller energy than node n, node n will check next child node because node c reduces the network lifetime if it replaces node n. Otherwise, node n initializes the P_{r} of node c as its created packet flow ($P_{\mathrm{c}}(c)$) and calculates the depth range (defined as $S(c,f)$ for convenience) where node c can communicate directly with the forwarding node f of node n. Simultaneously, the optimal depth range of node c (defined as $D(c)$) is initialized as $S(c,f)$ (lines 3–4). Then node n sorts its child nodes according the nodes' packet flow P_{s} (in descending order) (line 5), it checks each one (defined as k for convenience) in the set of node n's child nodes in descending order and calculates the depth range (defined as $S(c,k)$ for

Table 3. Notations in replacement node selection.

Notation	Definition
$Child(i)$	the set of child nodes of sensor node i
f	the forwarding node of the sensor node
$E(i)$	the energy of sensor node i
E_{a}	the energy consumption of adjusting depth per meter
E_{f}	the energy consumption of forwarding per packet
$D(i)$	the optimal depth range of node i
$S(i, j)$	the depth range of node i in which node i can communicate directly with j
$P_{\mathrm{r}}(i)$	the reserved packet flow of sensor node i
$P_{\mathrm{s}}(i)$	the packet flow sent by the sensor node i before the node replacement
$P_{\mathrm{c}}(i)$	the packet flow created by the sensor node i
$Z'(i)$	the new depth of node i
(X_i, Y_i, Z_i)	the coordinate of sensor node i
$Candidates$	The set of candidate replacement nodes

convenience) where node c can communicate with the node k (line 6). If the range $D(c)$ and $S(c, k)$ have an intersection, it means that node c can forward the packets from node k to the forwarding node of node n. Then the range $D(c)$ and the P_{r} of node c are updated (lines 8–11). After traversing all child nodes of node n except node c, $D(c)$ is the proper range where node c can forward the packets from node n's child nodes as more as possible. Then the new depth of node c is determined which minimizes the node c moving distance (line 13). According to the new depth of node c, node n calculates the optimal depth range for itself where node n forwards the packets from node c's child nodes to node c as more as possible (lines 14–23). The new depth of node n is determined (line 24) and the P_{r} of node c is updated (line 25). And then the residual energy that node n and node c have after replacement can be calculated (lines 26–27). If the residual energy of node c after replacement is larger than the energy of node n before the replacement and the lifetime of node n is prolonged through the replacement, node n adds node c into the set of $Candidates$ (lines 28–30). At last, node n selects the child node whose P_{r} is largest as the replacement node from $Candidates$.

Depth Adjustment. After the replacement node is selected, the sensor node (named A for easy explanation) broadcasts the new depth of the replacement node (named B for easy explanation) and the ID of the replacement node B. Each child node except the replacement node B checks whether the replacement node B can communicate with it at the new depth. If the replacement node B can forward the packets sent from the child node, the child node will use the replacement node B as its new forwarding node and delete the original forwarding node (node A) from the forwarding node candidate list. Otherwise, the child node will delete the original forwarding node (node A) from the forwarding node candidate list and then it will select the neighbor node which has largest residual energy from the forwarding node candidate list as its new forwarding node. Similarly, the replacement node B broadcasts the new depth of the sensor node A to its child nodes. The replacement node B's child nodes select their new forwarding nodes according whether they can communicate with the sensor node A.

After that, the sensor node A and the replacement node B adjust to their new depths. They update their forwarding node, coordinate, residual energy and Ds, and they broadcast their information (ID, coordinate, residual energy and Ds) to their new neighbors. Then they continue to work to forward packets and the network lifetime can be prolonged.

4 Experiments

4.1 Experimental Setup

In this section, we evaluate the performance of our proposed protocol EBDCR and compare it with DCR [5] and EEDBR [6]. We perform the simulations using Network Simulator (NS-2) [9]. In order to simulate the impairment of acoustic communication we use Aqua-Sim. Aqua-Sim is developed on NS-2 and can effectively simulate acoustic signal attenuation and packet collisions in underwater sensor networks [10]. In our simulations, sensor nodes are randomly deployed in a $1500\,\text{m} \times 1500\,\text{m} \times 1500\,\text{m}$ 3-D area. One or multiple sink nodes are deployed at the water surface, and we assume that all the sink nodes are stationary once deployed. All the sensor nodes have same communication range of 250 m, data rate of 50 Kbps, and CSMA MAC protocol, as in [11]. The packet generation rate for each sensor node is one packet per second. We consider that the data packet has size of 50 bytes, as [4]. Because the values of consumption in idling mode for all nodes are same, we don't consider the idling mode consumption for simplicity. The values of consumption in sending and receiving mode are 2 W and 0.1 W, respectively. And each node has initial energy of 60 WHr and consumes 15 J/m through vertical movement [12]. And the variable x in Formula 1 is set as 2. Simulation parameters are given in Table 4.

We used the following metrics for evaluating the performance of our proposed routing protocol:

- Network lifetime: The network lifetime is the time when the first node dies in the network because of the energy exhaustion.
- Average end-to-end delay: The average end-to-end delay is the average delay for the delivered packets.
- Average energy consumption: The average energy consumption is the energy consumption for every delivered packet.

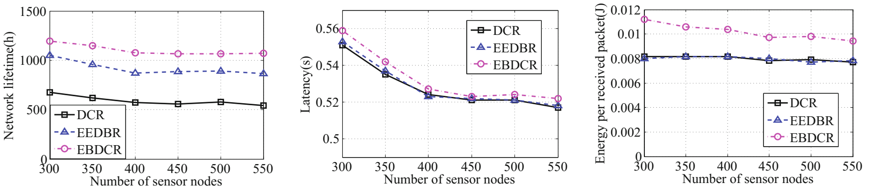
4.2 Experimental Results

In Fig. 6a, b and c, the number of sink nodes is set as 9. We first compare the network lifetime of three schemes with different number of sensor nodes.

Table 4. Simulation parameters.

Parameter	Value
Network size	1500 m × 1500 m × 1500 m
Communication range	250 m
Data rate	50 Kbps
Data packet size	50 bytes
Initial energy	60 WHr
Transmission power	2 W
Reception power	0.1 W
Idle power	10 mW
Vertical movement consumption	15 J/m

The result is shown as Fig. 6a. In DCR, each node forwards the packets to its neighbor which has smallest *Ds* all the time, it results that this neighbor will die earlier than the other neighbor nodes. So the lifetime of DCR is the smallest in three schemes. In EEDBR, each node selects the node with high residual energy in its neighbor as next forwarder and balances the energy consumption between its neighbors. Simulation shows that the lifetime of EEDBR is 50% higher than the lifetime of DCR. Our proposed EBDCR not only considers the balance of energy between each node's neighbors, but also balance the energy consumption between the nodes that have different depths. Hence, compared by the EEDBR, the lifetime of EBDCR is extended by 10% to 20%. And as the number of sensor nodes increases, the lifetime of three schemes decrease a little because the load of nodes near the sink nodes increases.



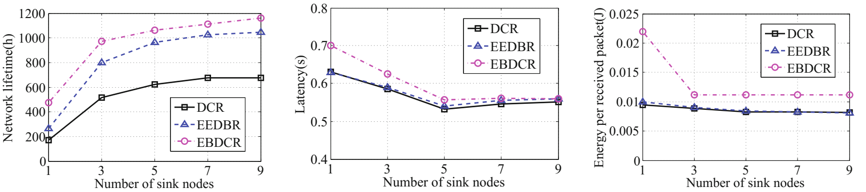
(a) Network lifetime (b) Average end-to-end delay (c) Average energy consumption

Fig. 6. Results with 9 sink nodes

Figure 6b shows the average end-to-end delay for all delivered packets with different number of sensor nodes. DCR and EEDBR have almost the same latency because the data packets are transmitted immediately when it arrives in a sensor node. The latency of our scheme increases by 1% to 2%. This is because the node replacement will affect the delivery of the data packets which are sent from the replaced node, the replacement node and their child nodes.

Furthermore, all the delays in three schemes decrease due to sensor nodes have greater selection to forward data as the number of nodes increases.

The energy consumption per delivered packet with different number of sensor nodes is shown in Fig. 6c. Because DCR and EEDBR don't adjust the depths of all sensor nodes, their energy consumption per delivered packet is almost same. In our scheme, sensor nodes need to adjust their depths to balance the energy. The energy consumption per delivered packet increases by 20% to 30%. Although our scheme needs a part of the energy to adjust, we take full advantage of the energy of the sensor nodes far from the sink nodes, thereby prolonging the network lifetime.



(a) Network lifetime (b) Average end-to-end delay (c) Average energy consumption

Fig. 7. Results with 300 sensor nodes

In Fig. 7a, b and c, the number of sensor nodes is set as 300. Figure 7a shows the lifetime with different number of sink nodes. As the number of sink nodes increases, the number of sensor nodes which can communicate with sink nodes directly increases. So the lifetime of three schemes also increases, and our scheme performs better than DCR and EEDBR. The average end-to-end delay with different number of sink nodes is shown in Fig. 7b. The fewer sink nodes we have, the more load the sensor nodes near the sink nodes have, the more data packets will be influenced when the nodes near the sink nodes are involved in node replacement. Hence, the latency of our scheme increases by 10% compared to DCR and EEDBR when the number of sink nodes is 1. As the sink nodes increase, the difference between our scheme and other two schemes decreases. Finally, we compare the energy consumption with different number of sink nodes in Fig. 7c. The sensor nodes near the sink nodes have more load when there is only one sink node. This caused that the sensor nodes around the sink node will be involved in node replacement more frequently. So the energy consumption per delivered packet in our scheme is higher. The difference between our scheme and other two schemes decreases as the sink nodes increase.

5 Conclusion

In this paper, we propose EBDCR, an energy-balanced and depth-controlled routing protocol for UWSN. In order to balance the energy consumption between the nodes near the sink nodes and the nodes far from the sink nodes, we decline the low-energy nodes and float up the high-energy nodes through the node replacement strategy we proposed. Moreover, we provide an algorithm to select a appropriate node to replace the node which has low energy. After that, we adjust them to their new depths to continue working. Finally, the experimental results demonstrate that our proposed routing protocol effectively prolongs the network lifetime.

Acknowledgement. This research is sponsored by National Key R&D Program of China No. 2017YFB0902602, the State Key Program of National Natural Science Foundation of China No. 61533011, Shandong Provincial Natural Science Foundation under Grant No. ZR2015FM001 and the Fundamental Research Funds of Shandong University No. 2015JC030.

References

1. Akyildiz, I.F., Pompili, D., Melodia, T.: Underwater acoustic sensor networks: research challenges. Ad Hoc Netw. **3**(3), 257–279 (2005)
2. Partan, J., Kurose, J., Levine, B.N.: A survey of practical issues in underwater networks. In: The Workshop on Underwater Networks, WUWNET 2006, Los Angeles, CA, USA, September, pp. 17–24. DBLP (2006)
3. Davis, A., Chang, H.: Underwater wireless sensor networks. In: Oceans, pp. 1–5. IEEE (2012)
4. Yan, H., Shi, Z.J., Cui, J.-H.: DBR: depth-based routing for underwater sensor networks. In: Das, A., Pung, H.K., Lee, F.B.S., Wong, L.W.C. (eds.) NETWORKING 2008. LNCS, vol. 4982, pp. 72–86. Springer, Heidelberg (2008). doi:10.1007/978-3-540-79549-0_7
5. Coutinho, R.W.L., Vieira, L.F.M., Loureiro, A.A.F.: DCR: depth-controlled routing protocol for underwater sensor networks. In: Proceedings - International Symposium on Computers and Communications, pp. 000453–000458 (2013)
6. Wahid, A., Lee, S., Jeong, H.-J., Kim, D.: EEDBR: energy-efficient depth-based routing protocol for underwater wireless sensor networks. In: Kim, T., Adeli, H., Robles, R.J., Balitanas, M. (eds.) AST 2011. CCIS, vol. 195, pp. 223–234. Springer, Heidelberg (2011). doi:10.1007/978-3-642-24267-0_27
7. Xie, P., Cui, J.-H., Lao, L.: VBF: vector-based forwarding protocol for underwater sensor networks. In: Boavida, F., Plagemann, T., Stiller, B., Westphal, C., Monteiro, E. (eds.) NETWORKING 2006. LNCS, vol. 3976, pp. 1216–1221. Springer, Heidelberg (2006). doi:10.1007/11753810_111
8. Erol, M., Vieira, L.F.M., Gerla, M.: AUV-aided localization for underwater sensor networks. In: International Conference on Wireless Algorithms, Systems and Applications, pp. 44–54. IEEE Xplore (2007)
9. The ns mannual. http://www.isi.edu/nsnam/ns/doc/index.html
10. Xie, P., Zhou, Z., Peng, Z., Yan, H.: Aqua-Sim: an NS-2 based simulator for underwater sensor networks, pp. 1–7. IEEE (2009)
11. Lee, U., Wang, P., Noh, Y., Vieira, L.F.M., Gerla, M., Cui, J.H.: Pressure routing for underwater sensor networks. In: Conference on Information Communications, pp. 1676–1684. IEEE Press (2016)

12. O'Rourke, M., Basha, E., Detweiler, C.: Multi-modal communications in underwater sensor networks using depth adjustment. In: ACM International Conference on Underwater Networks and Systems, pp. 1–5. ACM (2012)
13. Jafri, M.R., Ahmed, S., Javaid, N., Ahmad, Z., Qureshi, R.J.: AMCTD: adaptive mobility of courier nodes in threshold-optimized DBR protocol for underwater wireless sensor networks. Int. J. Distrib. Sens. Netw. **2**, 218–222 (2014)
14. Umar, A., Javaid, N., Ahmad, A., et al.: DEADS: depth and energy aware dominating set based algorithm for cooperative routing along with sink mobility in underwater WSNs. Sensors **15**(6), 14458–14486 (2015)
15. Javaid, N., Jafri, M.R., Ahmed, S., et al.: Delay-sensitive routing schemes for underwater acoustic sensor networks. Int. J. Distr. Sensor Netw. **11**, 532676 (2015)
16. Majid, A., Azam, I., Waheed, A., et al.: An energy efficient and balanced energy consumption cluster based routing protocol for underwater wireless sensor networks. J. Intell. Rob. Syst. **27**(4), 324–333 (2016)
17. Ahmed, M., Salleh, M., Channa, M.I.: Routing protocols based on node mobility for Underwater Wireless Sensor Network (UWSN): a survey. J. Netw. Comput. Appl. **78**, 242–252 (2016)
18. Li, N., Martínez, J.F., Chaus, J.M.M., Eckert, M.: A survey on underwater acoustic sensor network routing protocols. Sensors **16**(3), 414 (2016)
19. Ibrahim, D.M., Eltobely, T.E., Fahmy, M.M., et al.: Enhancing the vector-based forwarding routing protocol for underwater wireless sensor networks: a clustering approach. In: The Tenth International Conference on Wireless and Mobile Communications, ICWMC 2014, pp. 98–104 (2014)