Research Article

An Energy-Efficient Routing Algorithm for Underwater Wireless Sensor Networks Inspired by Ultrasonic Frogs

Ming Xu,1,2 Guangzhong Liu,1 and Huafeng Wu3

1 College of Information Engineering, Shanghai Maritime University, 1550 Haigang Avenue, Shanghai 201306, China
2 Shanghai Key Lab of Intelligent Information Processing, Fudan University, 220 Handan Road, Shanghai 200433, China
3 Merchant Marine College, Shanghai Maritime University, 1550 Haigang Avenue, Shanghai 201306, China

Correspondence should be addressed to Ming Xu; mingxu@shmtu.edu.cn

Received 15 August 2013; Accepted 24 December 2013; Published 13 February 2014

Academic Editor: Kun Hua

Copyright © 2014 Ming Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The area of three-dimensional (3D) underwater wireless sensor networks (UWSNs) has attracted significant attention recently due to its applications in detecting and observing phenomena that cannot be adequately observed by means of two-dimensional UWSNs. However, designing routing protocols for 3D UWSNs is a challenging task due to stringent constraints imposed by acoustic communications and high energy consumption in acoustic modems. In this paper, we present an ultrasonic frog calling algorithm (UFCA) that aims to achieve energy-efficient routing under harsh underwater conditions of UWSNs. In UFCA, the process of selecting relay nodes to forward the data packet is similar to that of calling behavior of ultrasonic frog for mating. We define the gravity function to represent the attractiveness from one sensor node to another. In order to save energy, different sensor nodes adopt different transmission radius and the values can be tuned dynamically according to their residual energy. Moreover, the sensor nodes that own less energy or locate in worse places choose to enter sleep mode for the purpose of saving energy. Simulation results show the performance improvement in metrics of packet delivery ratio, energy consumption, throughput, and end-to-end delay as compared to existing state-of-the-art routing protocols.

1. Introduction

Underwater wireless sensor networks (UWSNs) have a lot of potential application areas such as oceanographic data collection, disaster prevention, pollution monitoring, offshore exploration, and military surveillance [1–3]. Radio frequency (RF) signals suffer from severe attenuation in water and have been successfully deployed only at very low frequencies, involving large antenna and high transmission power. Hence, acoustic signals have been used for wireless communication in current underwater physical layer, which has challenges to be overcome such as long propagation delay resulting from low speed of sound propagation, severely limited bandwidth, and time-varying multipath propagation [4]. All the above distinct features of UWSNs give birth to new challenge areas for every level of the network protocol suite. UWSNs mainly consist of two communication architectures: two-dimensional and three-dimensional (3D) underwater networks [1]. 3D UWSNs are used to detect and observe phenomena that cannot be adequately observed by means of ocean bottom underwater sensor nodes, that is, to perform cooperative sampling of the 3D ocean environment [5, 6]. In 3D UWSNs, sensors float at different depths to observe a given phenomenon. Many problems arise with UWSNs that need to be solved in order to enable underwater monitoring in the new environment. Among them, providing efficient routing is a very challenging task due to the unique characteristics of UWSNs. In UWSNs, battery power is a vital resource for each wireless sensor because of the difficulty and cost in recharging sensor batteries once the network is deployed. Many methods for energy conservation at different layers have been investigated in terrestrial wireless sensor networks. However, these methods are not applicable to UWSNs. According to their architectures, the routing protocols of UWSNs can be divided into three categories: location-based routing, flat routing, and hierarchical routing [7]. Location-based routing has good scalability, but it requires
a positioning system or positioning algorithm to help the nodes to calculate the location information. Flat routing protocols have better robustness, but the excessive overhead for maintaining routing information restricts their application to small-scale underwater acoustic circumstances. Hierarchical routing also has good scalability, but the cluster maintenance overhead and the failure of key nodes will affect the routing efficiency.

Frog calling and hearing have been shown to be important for species recognition, mate assessment, and localization. Most interestingly, an unusual species called concave-eared torrent frog (Amolops tormotus) lives near the noisy Yellow Mountain in eastern China and produces diverse bird-like melodic calls that often contain spectral energy in the ultrasonic range (frequencies greater than 20 KHz) [8]. It is demonstrated that the male frogs emit advertisement calls using ultrasound to avoid masking by the wideband background noise of local fast-flowing streams. Although the female frogs exhibit no ultrasonic sensitivity, they emit courtship calls that evoke extraordinarily precise phonotaxis of the male frogs, rivalling that of vertebrates with the highest localization acuity (barn owls, dolphins, elephants, and humans) [9, 10]. Calling is linked to physical size and females may be attracted to more vigorous calls. The smallest frogs must consume lots of energy to produce calls. In male-male competition, some male frogs may stop calling or remain in chorus (each frog calls in turn) for longer periods of time based on a comparison between the benefit of obtaining a higher mating probability and the cost of losing more energy. In this paper, we present an ultrasonic frog calling algorithm (UFCA) for routing in UWSNs, which has been ethologically inspired by the calling behavior of concave-eared torrent frog. UFCA does not require fixed routing tables or periodic flooding messages for the routing path discovery. Therefore, it is resistant to node mobility and temporary loss of connectivity which are prevalent in UWSNs. In UFCA, different sensor nodes adopt different transmission radius, which can be tuned dynamically according to their residual energy. Moreover, the sensor nodes that own less energy or locate in worse places choose to enter sleep mode for the purpose of saving energy. As a distributed routing algorithm, no topology information needs to be exchanged among neighboring nodes and only a small fraction of the sensor nodes are involved in routing to ensure energy-efficient operations for surveillance and monitoring applications.

The remainder of the paper is organized as follows: Section 2 presents a brief overview of related work, while Section 3 introduces the proposed scheme in detail. Performance evaluation is described in Section 4. Finally, we conclude the paper in Section 5.

2. Related Work

The underwater environment introduces difficulties in designing efficient routing protocols not experienced terrestrially, such as transmission loss due to geometric spreading and absorption by the ocean [11, 12]. Tan et al. [13] proposed a new protocol based on hop-by-hop hybrid implicit/explicit acknowledgment scheme which is proposed for a multihop UWSN. In the protocol, data packets forwarded by downstream nodes can work as implicit ACKs for previous transmitted data packets. Vahdat and Becker [14] proposed epidemic routing (ER) protocol where each node replicates a packet to every encountered node. ER can utilize every opportunity to deliver a packet to the destination and maximize successful delivery ratio and minimize average end-to-end delay in unconstrained networks. However, this routing protocol consumes too many resources that make it not desirable in resource constrained networks such as UWSNs. Pompili et al. [15] introduced two distributed routing algorithms for delay-insensitive and delay-sensitive applications, respectively, with the objective of minimizing the energy consumption taking the varying condition of the underwater channel and the different application requirements into account.

Vector-based forwarding (VBF) [16] is a geographic approach, which allows the nodes to weigh the benefit to forward packets and reduce energy consumption by discarding low benefit packets. Therefore, over a multipath, only the nodes that are located within a pipe of given width between the source and the destination are considered for relaying. However, in the areas of low density of nodes VBF may not find the path close to the routing vector. Similarly, Jornet et al. proposed focused-beam routing (FBR) [17] protocol that is suitable for networks containing both static and mobile nodes. The objective of FBR is to determine which nodes are candidates for relaying. Candidate nodes are those that lie within a cone of angle $\pm \theta/2$ emanating from the transmitter towards the final destination. An RTS/CTS handshake is set up to isolate closer nodes within this cone. If a node determines that it is within the transmitter's cone, it will respond to the RTS. Those nodes that are outside the cone will not respond. A theoretical argument supporting geographic routing has been discussed in [18] based on simple propagation and energy consumption models for underwater networks. The study shows that an optimal number of hops along a path exist and that increasing the number of hops by choosing closer relays is preferable with respect to keeping the route shorter. In view of this, several position-based routing algorithms are proposed and compared; results show that selecting relays closer than a given maximum distance before seeking farther ones achieves in fact optimal energy consumption.

Depth-based routing (DBR) [19] can handle network dynamics efficiently without the assistance of a localization service. DBR forwards data packets greedily towards the water surface (i.e., the plane of data sinks). In DBR, a data packet has a field that records the depth information of its recent forwarder and is updated at every hop. But DBR has only greedy forwarding mode, which alone is not able to achieve high delivery ratios in sparse areas. Wahid et al. [20] proposed an energy-efficient routing protocol, called ERP2R (energy-efficient routing protocol based on physical distance and residual energy) based on the idea of utilizing the physical distances of the sensor nodes towards the sink node. ERP2R also takes into account the residual energy of the sensor nodes in order to extend the network lifetime. However, ERP2R may lead to longer routing path with the growth
of network density depending on the physical distances
towards the sink node, which in turn consumes additional energy. Moreover, the characteristics of node mobility in
UWSNs often make the problem worse. Ayaz and Abdullah
[21] proposed a hop-by-hop dynamic addressing based (H2-
DAB) routing protocol to provide scalable and time-efficient
routing for UWSN. The H2-DAB routing protocol does not
require any dimensional location information or any extra
specialized hardware compared with many other routing
protocols in the same area. However, the problem of multihop
routing still exists as it is based on multihop architecture,
where nodes near the sinks drain more energy because they
are used more frequently.

Packet redundancy and multiple paths can be exploited
in order to increase the reliability of UWSNs. Ayaz et al.
[22] provided a two-hop acknowledgment reliability model in
order to insure the reliable data deliveries to the surface sinks,
where two copies of the same data packet are maintained in
the network without extra burden on the available resources.
A relay node that has data packets to forward will not reply
with the acknowledgment until it cannot find the next hop
towards the destination. But if a node is unable to find the
next hop due to any failure, or even if it is lost, then packets
in the buffer are not considered lost. All the nodes that send
the data packets towards this node will wait for a certain
amount of time before trying again for the next hop. Xu et al.
[23] proposed a multiple-path forward error correction (M-
FEC) approach that integrated multiple-path communica-
tions and Hamming coding to eliminate retransmission and
enhance reliability in underwater sensor networks. Moreover,
a Markov model and a dynamical decision and feedback
scheme were developed to decrease the number of the paths
in order to save energy and ensure the desirable packet error
rate. However, M-FEC may cause much long delay because
of additional process of encoding and decoding the data
packets.

3. Proposed Scheme

3.1. Network Model and Energy Consumption. We consider
that 3D UWSNs are composed of a certain number of
sensor nodes uniformly scattered in monitoring fields. We
present a generic model for a 3D UWSN that is represented
by \( G = (V, E) \) with \( n \) sensor nodes. Each sensor node
is assigned with a triplet of coordinates \((x, y, z)\). We also
assume that all sensor nodes know their own locations
through a certain localization service [24]. Such assumption
is justified in underwater systems where fixed bottom-
mounted nodes have location information upon deployment.
In fact, the underwater localization is a nontrivial task
for which relatively very few options are available. Many
researchers have proposed a variety of localization schemes
and techniques to address this issue specially [25, 26]. It is
not always feasible to deploy anchor nodes at the sea floor for
deep water environment. In this case, mobile beacon nodes
such as autonomous underwater vehicles (AUVs), which
are equipped with internal navigation systems, are exploited
as reference nodes to assist in corresponding distributed
localization algorithms. This paper takes advantage of these
research results as existing preconditions.

**Definition 1.** The function \( \delta(u, v) \) defines the distance
between two nodes \( s_u \) and \( s_v \) in a 3D Euclidean space as
\[
\delta : N \times N \rightarrow \Gamma : \delta (u, v),
\]
\[
\delta (u, v) = \sqrt{(u_x - v_x)^2 + (u_y - v_y)^2 + (u_z - v_z)^2}.
\]  \tag{1}

Underwater wireless sensor nodes are equipped with
sensing devices. They collect data from the external environ-
ment and transmit these data by one or multihop to the sink
node. Sink node is the node that generates data aggregation
results and also the target location of the data transmission.
Each sensor node can either transmit or receive data packets.
All sensor nodes can tune their transmission radius ranged
from \( r_{\text{min}} \) (minimum transmission radius) to \( r_{\text{max}} \) (maximal
transmission radius).

Consider two sensor nodes at minimum hop distance \( h \),
there exist two values \( u(h) \) and \( v(h) \) such that the Euclidean
distance \( \delta(u, v) \) between the two nodes is bounded; that is,
\( u(h) \leq \delta(u, v) \leq v(h) \). The quality of the bounds depends on
the network density \( \rho \). In particular for each \( h > 0 \) holds
\[
\lim_{p \to \infty} v(h) - u(h) = r_{\text{min}},
\]  \tag{2}
where \( r_{\text{min}} \) is the minimum transmission range of the sensor
nodes.

Sensing devices generally have widely different theoret-
ical and physical characteristics. Thus, numerous models of
varying complexity can be constructed based on application
needs and device features. However, for most kinds of
sensors, the sensing ability diminishes as distance increases.

**Definition 2.** For a sensor \( s \), the general sensing model \( S \) at an
arbitrary point \( p \) is expressed as
\[
S(s, p) = \frac{\lambda}{[d(s, p)]^k},
\]  \tag{3}
where \( d(s, p) \) is the Euclidean distance between the sensor \( s \)
and the point \( p \), and positive constants \( \lambda \) and \( k \) are sensor
technology-dependent parameters [27].

We assume that all sensor nodes are equipped with
limited battery resources without recharging or replacing
node batteries after deployment. The network lifetime is
defined as the time until the first sensor node in the network
depletes its energy. The energy consumption model is the
same as that in [28] where the attenuation and the energy
spreading factor (1 is for cylindrical, 1.5 is for practical, and
2 is for spherical spreading) are taken into consideration.

Acoustic signal has different transmission modes in
shallow water (where the depth of the water is lower than
100 meters) and deep water (where the depth of the water
is above 100 meters). In shallow water, the transmission
of the acoustic signal is limited to a cylindrical area from
bottom to the surface, while in deep water, the transmission of

International Journal of Distributed Sensor Networks
the acoustic signal is mainly with spherical diffusion and the energy consumption is caused by spherical diffusion and water absorption. This paper concentrates on the shallow water scenario.

The passive sonar equation [29] characterizes the signal-to-noise ratio (SNR) of an emitted underwater signal at the receiver, which is presented by

\[
\text{SNR} = \text{SL} - \text{TL} - \text{NL} + \text{DI},
\]

where \( \text{SL} \) is the target source level or noise generated by the target, \( \text{TL} \) is the transmission loss, \( \text{NL} \) is the noise level, and \( \text{DI} \) is the directivity index (a function of the receiver’s directional sensitivity).

The transmission loss \( \text{TL} \) can be defined as the accumulated decrease in acoustic intensity as an acoustic pressure wave propagates outwards from a source. The transmission loss for cylindrically spread signals is calculated as

\[
\text{TL} = 10 \log_{10} \left( \frac{I_n}{I_t} \right) \text{in dB/Km},
\]

where \( I_n \) is the intensity of the noise level and \( I_t \) is the intensity of the transmitted signal.

\[
\alpha = 0.11 f^2 + \frac{44 f^2}{4100 + f^2} + 2.75 \times 10^{-4} f^2 + 0.003,
\]

where \( f \) is in KHz and \( \alpha \) is in dB/Km.

The noise level \( \text{NL} \) in shallow water is mainly affected by waves, shipping traffic, wind level, and the activities of large mammals. For simplicity, we consider an average value for the noise level \( \text{NL} \) to be 70 dB as a representative shallow water case [30].

SL can be defined as the intensity of the radiated sound in decibels related to the transmitted signal intensity at 1 meter from the source according to the following expression:

\[
\text{SL} = 10 \log_{10} \left( \frac{I_t}{1 \text{µPa}} \right),
\]

where \( I_t \) is in µPa. Solving for \( I_t \) yields

\[
I_t = 10^{\text{SL}/10} \times 0.67 \times 10^{-18}.
\]

As a result, the transmitter power \( P_t \) that achieves intensity \( I_t \) at a distance of 1 meter from the transmitter in the direction to the receiver is calculated as

\[
P_t = 2 \pi \times H \times I_t,
\]

where \( P_t \) is in watts and \( H \) is the water depth in meters.

### 3.2. Ultrasonic Frog Calling Strategy

UFCA is inspired from the calling behavior of concave-eared torrent frog. Male concave-eared torrent frogs can produce diverse bird-like melodic advertisement calls with pronounced frequency modulations that often contain spectral energy in the ultrasonic range. Although female concave-eared torrent frogs exhibit no ultrasonic sensitivity, their courtship calls can evoke extraordinarily precise phonotaxis of the male frogs with high localization acuity.

Suppose there are six concave-eared torrent frogs randomly distributed in a space as shown in Figure 1. Frog \( f_i \) is a gravid female frog (with tone bursts frequency range 1–14 KHz [10]). Others are male frogs that can emit ultrasonic sound and have ultrasonic hearing capacity in response to tone bursts at frequency ranged from 1 KHz to 35 KHz [10]. At first, \( f_i \) emits a courtship call in order to attract some nearby male frogs. The solid circle with radius \( r_i \) represents the covering space of \( f_i \)’s courtship call. The number in each frog denotes its body size. As \( f_i \) is the nearest male frog to \( f_i \), it will emit an advertisement call at frequencies ranged from normal sound to ultrasonic sound immediately after the reception of \( f_i \)’s courtship call. The dashed circle with radius \( r_j \) represents the covering space of \( f_j \)’s advertisement call, which is bigger than the covering space of \( f_i \)’s courtship call. After the male frog \( f_k \) receives \( f_i \)’s courtship call and \( f_j \)’s advertisement call, it extracts the body size information from these calls. As \( f_k \)’s body size is smaller than that of \( f_i \), it will not broadcast any advertisement call in order to save energy. The male frog \( f_k \) can also hear \( f_j \)’s advertisement call, but it still keeps silent since \( f_k \)’s is located outside of the covering space of \( f_j \)’s courtship call. Both the male frogs \( f_p \) and \( f_q \) locate within the covering space of \( f_j \)’s courtship call. Suppose \( f_p \) and \( f_q \) receive \( f_j \)’s advertisement call simultaneously, they compare their body sizes and conclude that the probability of winning the competition is high. Therefore, both \( f_p \) and \( f_q \) directly reply with advertisement calls to \( f_j \), which include the information of their body sizes and locations. Judging from advertisement calls of different male frogs, \( f_j \) selects \( f_q \) as its mate because \( f_q \) owns the biggest body size among the three mating candidates \( f_j \), \( f_p \), and \( f_q \). At last, \( f_i \) calculates \( f_q \)’s position and leaps to \( f_q \).
3.3. Routing Algorithm. UFCA consists of two phases: candidate discovery phase and relay node selection phase. Figure 2 illustrates the candidate discovery phase. Each frog denotes a sensor node and each number in the frog denotes the residual energy of local sensor node. Sink nodes do not have any energy constraints because they are equipped with both radio-frequency (RF) and acoustic modems and are deployed at the water surface. As for static sink nodes, they only need to broadcast their positions to the whole network one time at the initial stage of the network operation, which would not produce significant energy dissipation [31]. The sensor node that holds the data packet is the transmitter, which is similar to the gravid female frog in Figure 1. Each data packet carries the positions of the source node, the sink node, and the relay node (i.e., the node that transmits this packet). Suppose $s_i$ is a transmitter as shown in Figure 2 then other sensor nodes are receivers before the data packet is forwarded. At first, $s_i$ transmits a courtship packet with radius $r_{\min}$, which includes the positions of $s_i$ and the sink node $s_j$. As $s_j$ is the nearest receiver to $s_i$, it will calculate the cosine of the angle between the direction from $s_i$ to $s_j$ and the direction from $s_i$ to $s_k$ (denoted by $A$ in Figure 2) upon receipt of $s_i$’s courtship packet. If the cosine value is not below zero, $s_j$ will transmit an advertisement packet with radius $r_j$, which is calculated as

$$r_j = \text{MIN} \left\{ \left( 1 + \frac{e_{\text{res}}}{e_{\text{max}}} \right) \cdot r_{\min}, r_{\max} \right\},$$

where $e_{\text{res}}$ denotes the residual energy of sensor node $s_j$ and $e_{\text{max}}$ denotes the maximum energy of sensor node $s_j$. Thus, $r_j$ ranges from $r_{\min}$ to $2r_{\min}$. In the best case, the residual energy of $s_j$ is full and $r_{\max} > 2r_{\min}$ and $r_j$ equals $2r_{\min}$ according to formula (10), which is enough to cover $s_j$’s transmission circle. In the worst case, the residual energy of $s_j$ is almost exhausted; it will only transmit an advertisement packet with radius $r_{\min}$ in order to reach the position of $s_j$. Moreover, $s_k$’s position and residual energy information is included in its advertisement packet. After $s_k$ receives $s_j$’s courtship packet and $s_j$’s advertisement packet, it extracts the position and the residual energy information from these packets. As $s_k$’s residual energy is less than that of $s_j$, it chooses to enter sleep mode in order to save energy without transmitting any advertisement packet. Another receiver $s_q$ can also receive $s_j$’s courtship packet and $s_j$’s advertisement packet. But $s_q$ will choose to enter sleep mode because the cosine of the angle between the direction from $s_i$ to $s_q$ and the direction from $s_j$ to $s_q$ is below zero. In other words, $s_q$ locates in a worse place compared with other receivers. Although the receiver $s_h$ locates within the transmission radius of $s_j$’s advertisement packet, it still keeps sleep mode since $s_h$ cannot receive $s_j$’s courtship packet. After $s_q$ receives $s_j$’s courtship packet and $s_j$’s advertisement packet, it extracts the position and the residual energy information from these packets. As $s_q$’s residual energy is more than that of $s_j$’s and the cosine of the angle between the direction from $s_i$ to $s_q$ and the direction from $s_j$ to $s_q$ is not below zero, it concludes that the probability of winning the competition is high. Therefore, $s_q$ will transmit an advertisement packet with radius $r_{p}$, which includes the information of its location and residual energy. At last, $s_i$ will add $s_j$ and $s_q$ to its candidate set after the receipt of their advertisement packets. The sensor node that goes to sleep mode will wake up immediately after another sensor node broadcasts a courtship packet and the sleep sensor node locates exactly within its transmission range.

The process of selecting a candidate as the relay node to forward the data packet is illustrated in Figure 3. After the transmitter $s_j$’s candidate set is constructed, it will select the most attractive candidate as the relay node according to a certain standard, which is described as the gravity function in this paper.

**Definition 3.** Given a sensor node $s_i$ and its neighbor node $s_j$, the gravity function from $s_i$ to $s_j$ is defined as $G_{ij}$ and its value is calculated as

$$G_{ij} = \frac{e_{\text{res}} \cdot e_{\text{res}} \cdot \cos A}{\delta(i, j)^3},$$

where $e_{\text{res}}$ and $e_{\text{res}}$ denote the residual energy of sensor nodes $s_i$ and $s_j$, $A$ is the intersection angle between the direction from $s_i$ to $s_j$ and the direction from $s_i$ to $s_k$, and $\delta(i, j)$ is the Euclidean distance from $s_i$ to $s_j$.

At last, the transmitter $s_i$ computes the gravity values with every sensor node in its candidate set and chooses the candidate with maximal gravity value to be the relay node that is in charge of forwarding the data packet.

Algorithm 1 describes the process of building the routing path with ultrasonic frog calling algorithm in detail.
All data packets at relay nodes should have limited lifetime, which are controlled by TTL (time-to-live) information carried in the packet header. At first, the routing path \( p \) is created as an empty queue structure after initialization as described in line 1. While TTL value is bigger than zero and the sink node is not reached, the process of building the routing path is repeatedly executed. Then, the source node \( s_i \) resets its candidate set and transmits a courtship packet with the minimum transmission radius \( r_{\text{min}} \) in order to find some candidates as described from line 3 to line 4. After that, all sensor nodes that locate within the covering space of \( s_i \)'s transmission radius will check their positions. Suppose \( s_j \) is the first receiver with \( \delta(i, j) < r_{\text{min}} \). If the cosine of the angle between the direction from \( s_i \) to \( s_j \) and the direction from \( s_i \) to \( s_j \) is below zero, then \( s_j \) chooses to enter sleep mode for saving energy. Otherwise, \( s_j \) adds \( s_j \) to its candidate set and \( s_j \) transmits an advertisement packet with radius \( r_j \) according to formula (10). And then, all sensor nodes that locate within the covering space of \( s_j \)'s transmission radius will compare their residual energy with that of \( s_j \). Suppose \( s_k \) is a sensor node that receives \( s_j \)'s courtship packet and \( s_k \)'s advertisement packet. If \( s_k \)'s residual energy is less than that of \( s_j \), it will choose to enter sleep mode without competing with \( s_j \). Otherwise, \( s_k \) will add \( s_j \) to its candidate set. The operation is iterated until all candidates are discovered as described from line 5 to line 17. During the phase of relay node selection, \( s_j \) calculates the gravity values with every sensor node in its candidate set. Suppose \( s_j \) is the candidate with the maximal gravity value among all candidates. As a result, \( s_j \) is selected as the relay node and is added to the routing path \( p \) as described from line 18 to line 19. Hereafter, \( s_j \) becomes the transmitter and continues to find the relay node of next hop. Meanwhile, the TTL value is decreased by 1 so as to control the lifetime of the data packet as described from line 20 to line 23. At last, if the sink node \( s_j \) is found within the given TTL value, an optimized routing path \( p \) is returned. Otherwise, all elements will be removed from \( p \), which means no sink node is found as described from line 24 to line 28.

In many proactive routing protocols, the active sensor nodes must send periodic update packets to other nodes even when the routing information is similar to the previous one. Moreover, the storage overhead for routing table maintenance also grows quickly as the size of the network increases. Although some reactive routing protocols can avoid the overhead incurred by routing table maintenance, the periodic flooding messages for the routing path discovery is another deadly cost in resource-constraint underwater wireless sensor networks. In UFCA, the update of candidate set is evoked only when this sensor node is selected as a transmitter. After that, it can determine where to forward a data packet without the need of routing table maintenance or any flooding mechanism.

4. Performance Evaluation

4.1. Simulation Settings. We use Aqua-Sim [32] as simulation framework to evaluate our approach. Aqua-Sim is an ns-2 based underwater sensor network simulator developed by underwater sensor network lab at University of Connecticut. To simulate acoustic channels, we extend Aqua-Sim with spherical path loss and Thorp attenuation. We use a 3D region with size 1000 m × 1000 m × 1000 m and different number of sensor nodes varied from 100 to 600. Six sink nodes are randomly deployed at the water surface, which are assumed stationary in all simulations. The sensor nodes follow the random-walk mobility pattern. Each sensor node randomly selects a direction and moves to the new position with a random speed between the minimal speed and maximal speed, which are 0 m/s and 4 m/s, respectively. The data generating rate varies from one packet per second to 6 packets per second with a packet size of 50 bytes (i.e., from 400 bps to 2.4 kbps). The communication parameters are similar to those on a commercial acoustic modem and the bit rate is 10 kbps. TTL (time-to-live) value is set to 30 hops for each data packet. Each result is obtained from the average run of 40 times.

As the long propagation delay and limited bandwidth of acoustic channels make the existing MAC protocols widely used in radio networks unpractical for UWSNs, this paper adopts R-MAC [32] protocol as the underlying MAC protocol in order to avoid data packet collision. R-MAC schedules the transmission of control packets and data packets at both the sender and the receiver to avoid data packet collisions. Therefore, we do not distinguish courtship packet and advertisement packet from each other in MAC layer. In fact, we only need to make certain that which node is the sender and which node is the receiver in this session.
Input: source node $s_i$, sink node $s_t$, TTL;
Output: routing path $p$;
(1) Queue $p \leftarrow \Phi$; // routing path initialization
(2) while ($TTL > 0$) and ($s_i \neq s_t$) do
(3) $s_i.CandiSet \leftarrow \Phi$;
(4) $s_i$ transmits a courtship packet with radius $r_{min}$;
(5) for all $s_j$ with $\delta(i, j) < r_{min}$ do
(6) if $\cos(\angle TJI) < 0$ then
(7) $s_j.sleep();$
(8) else $s_i.CandiSet.add(s_j);$;
(9) $s_j$ transmits an advertisement packet with radius $r_j$
according to formula (10);
(10) for all $s_k$ with $\delta(j, k) < r_j$ do
(11) if $e_{j}^{m} < e_{k}^{m}$ then
(12) $s_j.sleep();$
(13) else $s_i.CandiSet.add(s_k);$;
(14) endif
(15) endfor
(16) endif
(17) endfor
(18) if $[G_{ij} = \max(G_{ik}) \forall s_k \in s_i.CandiSet]$ then
(19) $p.enqueue(s_j);$;
(20) $s_i \leftarrow s_j;$;
(21) TTL $-$; 
(22) endif
(23) endwhile
(24) if $s_i \neq s_t$ then
(25) $p.clear();$
(26) return $\Phi$;
(27) else return $p$;
(28) endif

Algorithm 1: Building the routing path with UFCA.

We use the following metrics to evaluate the performance of different routing protocols.

(1) **Packet delivery ratio** is defined as the ratio of the number of distinct data packets received successfully at the sinks to the total number of data packets generated at the source node.

(2) **Energy consumption** takes into account the total energy consumed in packet delivery, including transmitting, receiving, and idling energy consumption of all nodes in the network.

(3) **Throughput** equals the total data bits received at the sink nodes divided by the simulation time.

(4) **Average end-to-end delay** represents the average time taken by a data packet that travels from a source node to any sink node.

We compared the performance of ultrasonic frog calling algorithm (UFCA) with that of vector-based forwarding (VBF) and ERP$^2$R (energy-efficient routing protocol based on physical distance and residual energy). In the simulations of UFCA, the minimal and maximal transmission range is set to 50 meters and 100 meters, respectively, in all directions, while the transmission range in VBF and ERP$^2$R is fixed at 100 meters. Moreover, the routing pipe radius in VBF is set to 20 meters, which is a default value in [16].

4.2. Simulation Results. In the first set of simulations, we compared the packet delivery ratio with the number of nodes in different routing protocols. The average speed of nodes is set to 2 m/s. As shown in Figure 4, the packet delivery ratio of three routing protocols is proportional to the number of nodes. UFCA performs best among the three routing protocols in the same circumstances and VBF achieves higher packet delivery ratio than that of ERP$^2$R. Moreover, the curve of VBF rises faster than other protocols. This is because, with the growth of network density, more sensor nodes will fall in the routing pipe of VBF with fixed radius as the transmission range. The packet delivery ration of UFCA is significantly improved over other protocols especially when the network is sparse as UFCA can find more routing paths for data delivery in sparse networks. Specifically, UFCA improves 34.3% of the packet delivery ratio than that of ERP$^2$R and 11.9% of the packet delivery ratio than that of VBF on average.

Figure 5 illustrates the comparison of the packet delivery ratio with average speed of nodes in different routing protocols. The number of sensor nodes is set to 400 for each protocol. Overall, the packet delivery ratio of three
routing protocols is inversely proportional to average speed of nodes. UFCA achieves higher packet delivery ratio than that of ERP²R and VBF when their speeds of nodes are the same. The packet delivery ratio of ERP²R decreases rapidly with the growth of node mobility. This is because the rate of updating routing information in ERP²R cannot catch up with the increase of node mobility. Specifically, UFCA improves 32.5% of the packet delivery ratio than that of ERP²R and 6.4% of the packet delivery ratio than that of VBF on average.

In the second set of simulations, we compared the energy consumption with the number of nodes in different routing protocols. The average speed of nodes is set to 2 m/s. As shown in Figure 6, the energy consumption of three routing protocols is proportional to the number of nodes. UFCA performs better than other routing protocols in the same circumstances. Moreover, the curve of UFCA has a gentler slope compared with that of ERP²R and VBF. This is mainly due to more sensor nodes entering the sleep mode with the increase in sensor nodes in UFCA. ERP²R consumes less energy than VBF because energy factor is not given in the routing determination of VBF. As a result, UFCA decreases 26.1% of the energy consumption than ERP²R and 41.5% of the energy consumption than VBF on average.

Figure 7 illustrates the comparison of energy consumption with average speed of nodes in different routing protocols. The number of nodes is set to 400 for each protocol. The energy consumption of three routing protocols is proportional to the TTL value. UFCA consumes less energy than ERP²R and VBF when their speeds of nodes are the same. Moreover, the curve slopes of UFCA and VBF are rather gentle compared with that of ERP²R, which means that the factor of node mobility has slight influence on energy consumption of UFCA and VBF. ERP²R consumes less energy than VBF except when average speed of nodes reaches 4 m/s. On average, UFCA decreases 25.7% of the energy consumption than ERP²R and 36.2% of the energy consumption than VBF.
In the third set of simulations, we compared the throughput with the number of nodes in different routing protocols. The average speed of nodes is set to 2 m/s for each protocol. As shown in Figure 8, the throughput of three routing protocols is proportional to the number of nodes. The front parts of curves indicate rapid increases in throughput while the rear parts of curves show slow growth rates after the number of nodes has reached high value. The reason is that with the growth of network density, the routing paths become more crowded and downstream nodes cannot receive data packets from several of its upstream nodes simultaneously. Overall, UFCA performs better than other routing protocols in the same circumstances. VBF achieves higher throughput than ERP²R. On average, UFCA improves 21.5% of the throughput than ERP²R and 9.3% of the throughput than VBF.

Figure 9 depicts the comparison of the throughput with average speed of nodes in different routing protocols. The number of nodes is set to 400 for each protocol. The throughput of three routing protocols is inversely proportional to average speed of nodes. UFCA achieves higher throughput than that of ERP²R and VBF when their average speeds of nodes are the same. Noticeably, the throughput of ERP²R decreases sharply when average speed of nodes is more than 2 m/s. This is because more routing cost and residual energy of the nodes as well as their neighbors along routing paths have to be recalculated with the increase in average speed of nodes in ERP²R. On average, UFCA improves 15.4% of the throughput than ERP²R and 6.5% of the throughput than VBF.

In the last set of simulations, we compared the average end-to-end delay with the number of nodes in different routing protocols. The average speed of nodes is set to 2 m/s for each protocol. As shown in Figure 10, the average end-to-end delay of three routing protocols is inversely proportional to the number of nodes. UFCA achieves less end-to-end delay than ERP²R and VBF when the number of nodes is the same. The reason is that UFCA introduces less control...
packets than other protocols for communicating with the related sensor nodes during the process of routing. The cost for the computation of residual energy and gravity values in UFCA is far less than that in network communication. ERP$^2$R performs better than VBF because the highest priority node in ERP$^2$R has a holding time of zero, which can reduce the end-to-end delay to a certain degree. On average, UFCA decreases 11.2% of the average end-to-end delay than ERP$^2$R and 31.2% of the average end-to-end delay than VBF.

Figure II shows the comparison of the average end-to-end delay with the average speed of nodes in different routing protocols. The number of nodes is set to 400 for each protocol. Overall, the average end-to-end delay of three routing protocols is inversely proportional to the average speed of nodes. UFCA achieves less end-to-end delay than ERP$^2$R and VBF when their average speeds of nodes are the same. It is worth noting that ERP$^2$R owns a curve with rapid increasing trend. This is because more sensor nodes in ERP$^2$R need to reevaluate their distances to the sink node with the growth of node mobility. Specifically, UFCA decreases 8.1% of the average end-to-end delay than ERP$^2$R and 26.3% of the average end-to-end delay than VBF on average.

4.3. Discussion. Compared to algorithms such as VBF and ERP$^2$R, UFCA is totally a different approach. In VBF, only the sensor nodes located in a predefined routing pipe are eligible for packet forwarding, and those which are not close to the routing pipe do not forward the packets no matter whether they are suitable for building a shorter routing path. Therefore, the routing performance in VBF mainly depends on the node density and it cannot benefit from the deployment of multiple sink nodes if they are not close to each other. In ERP$^2$R, forwarding nodes are selected based on the physical distance of the sensor nodes. Each sender selects the nodes nearer to the sink node for routing decision, which is not always helpful when the node density is sparse. Although ERP$^2$R can balance the energy consumption using a residual energy-based timer, its performance decreases dramatically with the growth of node mobility. UFCA is inspired by the calling behavior of concave-eared torrent frog. In UFCA, the process of finding an optimal routing path is similar to the process of mating with an appropriate frog with characteristics of accurate and energy-efficient. Consequently, UFCA achieves better routing performance than VBF and ERP$^2$R regardless of node density and mobility. Moreover, different sensor nodes adopt different transmission radius according to their residual energy in UFCA and the sensor nodes that own less energy or locate in worse places choose to enter sleep mode for the purpose of saving energy. Through these means, the energy consumption is somehow equalized on the whole and the network lifetime is prolonged. Thus, the inherent adaptive nature of such algorithm is one of the main attractions in biologically inspired approaches.

5. Conclusion

Finding an optimal routing path in adverse underwater environment in 3D UWSNs has always been a challenging task, especially when the factor of energy consumption is taken into consideration. Inspired by the calling behavior of ultrasonic frog in mating, this paper proposed an ultrasonic frog calling algorithm (UFCA) that aims to achieve energy-efficient routing under harsh underwater conditions.
of UWSNs. UFCA does not require fixed routing tables or periodic flooding messages for the discovery of routing path. In UFCA, different sensor nodes adopt different transmission radius, which can be tuned dynamically according to their residual energy. Moreover, the sensor nodes that own less energy or locate in worse places choose to enter sleep mode for the purpose of saving energy. Simulation results show the performance improvement in metrics of packet delivery ratio, energy consumption, throughput, and end-to-end delay as compared to existing state-of-the-art routing protocols.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was sponsored by the National Nature Science Foundation of China (61202370, 51279099), the Innovation Program of Shanghai Municipal Education Commission (14YZ110), the Shanghai Pujiang Program from Science and Technology Commission of Shanghai Municipality (11PJ1404300), and the Open Program of Shanghai Key Laboratory of Intelligent Information Processing (IIPL-2011-008).

References

[1] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," Ad Hoc Networks, vol. 3, no. 3, pp. 257–279, 2005.

[2] C. Detweiler, M. Doniec, I. Vasilescu, and D. Rus, "Autonomous depth adjustment for underwater sensor networks: design and applications," IEEE/ASME Transactions on Mechatronics, vol. 17, no. 1, pp. 16–24, 2012.

[3] S. Basagni, C. Petrioli, R. Petriccoa, and M. Stojanovic, "Optimized packet size selection in underwater wireless sensor network communications," IEEE Journal of Oceanic Engineering, vol. 37, no. 3, pp. 321–337, 2012.

[4] J. M. Jornet, M. Stojanovic, and M. Zorzi, "On joint frequency and power allocation in a cross-layer protocol for underwater acoustic networks," IEEE Journal of Oceanic Engineering, vol. 35, no. 4, pp. 936–947, 2010.

[5] G. Isbitiren and O. B. Akan, "Three-dimensional underwater target tracking with acoustic sensor networks," IEEE Transactions on Vehicular Technology, vol. 60, no. 8, pp. 3897–3906, 2011.

[6] D. Pompili, T. Melodia, and I. F. Akyildiz, "Three-dimensional and two-dimensional deployment analysis for underwater acoustic sensor networks," Ad Hoc Networks, vol. 7, no. 4, pp. 778–790, 2009.

[7] M. Ayaz, I. Baig, A. Abdullah, and I. Faye, "A survey on routing techniques in underwater wireless sensor networks," Journal of Network and Computer Applications, vol. 34, no. 6, pp. 1908–1927, 2011.

[8] A. S. Feng, P. M. Narins, C.-H. Xu et al., "Ultrasonic communication in frogs," Nature, vol. 440, no. 7082, pp. 333–336, 2006.

[9] J.-X. Shen, A. S. Feng, Z.-M. Xu et al., "Ultrasonic frogs show hyperacute phonotaxis to female courtship calls," Nature, vol. 453, no. 7197, pp. 914–916, 2008.

[10] J.-X. Shen, Z.-M. Xu, Z.-L. Yu, S. Wang, D.-Z. Zheng, and S.-C. Fan, "Ultrasonic frogs show extraordinary sex differences in auditory frequency sensitivity," Nature Communications, vol. 2, no. 1, article 342, 2011.

[11] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "The challenges of building scalable mobile underwater wireless sensor networks for aquatic applications," IEEE Network, vol. 20, no. 3, pp. 12–18, 2006.

[12] I. F. Akyildiz, D. Pompili, and T. Melodia, "State-of-the-art in protocol research for underwater acoustic sensor networks," in Proceedings of the 1st ACM International Workshop on Underwater Networks, pp. 7–16, Los Angeles, Calif, USA, September 2006.

[13] H.-P. Tan, W. K. G. Seah, and L. Doyle, "A multi-hop ARQ protocol for underwater acoustic networks," in Proceeding of the OCEANS ’07, p. 1–6, Aberdeen, Scotland, June 2007.

[14] A. Vahdat and D. Becker, "Epidemic routing for partially connected Ad Hoc networks," Tech. Rep. TR CS-200006, 2000.

[15] D. Pompili, T. Melodia, and I. F. Akyildiz, "Routing algorithms for delay-insensitive and delay-sensitive applications in underwater sensor networks," in Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MOBICOM ’06), pp. 298–309, Los Angeles, Calif, USA, September 2006.

[16] P. Xie, J.-H. Cui, and L. Lao, "VBF: vector-based forwarding protocol for underwater sensor networks," in NEnERGYWORKing 2006. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems, vol. 3976 of Lecture Notes in Computer Science, pp. 1216–1221, 2006.

[17] J. M. Jornet, M. Stojanovic, and M. Zorzi, "Focused beam routing protocol for underwater acoustic networks," in Proceedings of the 3rd International Workshop on Underwater Networks (WUWNet ’08), pp. 73–82, San Francisco, Calif, USA, September 2008.

[18] M. Zorzi, P. Casari, N. Baldo, and A. F. Harris III, "Energy-efficient routing schemes for underwater acoustic networks," IEEE Journal on Selected Areas in Communications, vol. 26, no. 9, pp. 1754–1766, 2008.

[19] H. Yan, Z. J. Shi, and J.-H. Cui, "DBR: depth-based routing for underwater sensor networks," in NETWORKING 2008 Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet, vol. 4982 of Lecture Notes in Computer Science, pp. 72–86, Springer, 2008.

[20] A. Wahid, S. Lee, and D. Kim, "An energy-efficient routing protocol for UWSNs using physical distance and residual energy," in Proceedings of the OCEANS ’11, pp. 1–6, Santander, Spain, June 2011.

[21] M. Ayaz and A. Abdullah, "Hop-by-hop dynamic addressing based (H2-DAB) routing protocol for underwater wireless sensor networks," in Proceedings of the International Conference on Information and Multimedia Technology (ICIMT ’09), pp. 436–441, Jeju Island, South Korea, December 2009.

[22] M. Ayaz, A. Abdullah, and I. Faye, "Hop-by-hop reliable data deliveries for underwater wireless sensor networks," in Proceedings of the 5th International Conference on Broadband Wireless Computing, Communication and Applications (BWCCA ’10), pp. 363–368, November 2010.

[23] J. Xu, K. Li, and G. Min, "Reliable and energy-efficient multipath communications in underwater sensor networks," IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 7, pp. 1326–1335, 2012.
[24] Z. Zhou, Z. Peng, J.-H. Cui, Z. Shi, and A. Bagtzoglou, “Scalable localization with mobility prediction for underwater sensor networks,” IEEE Transactions on Mobile Computing, vol. 10, no. 3, pp. 335–348, 2011.

[25] W. Cheng, A. Y. Teymorian, L. Ma, X. Cheng, X. Lu, and Z. Lu, “Underwater localization in sparse 3D acoustic sensor networks,” in Proceedings of the 27th IEEE Communications Society Conference on Computer Communications (INFOCOM ’08), pp. 798–806, Phoenix, Ariz, USA, April 2008.

[26] H.-P. Tan, Z. A. Eu, and W. K. G. Seah, “An enhanced underwater positioning system to support deepwater installations,” in Proceedings of the MTS/IEEE OCEANS 2009, Marine Technology for Our Future: Global and Local Challenges, pp. 1–8, Biloxi, Miss, USA, October 2009.

[27] S. Meguerdichian, F. Koushanfar, G. Qu, and M. Potkonjak, “Exposure in wireless ad-hoc sensor networks,” in Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, pp. 139–150, Rome, Italy, July 2001.

[28] E. M. Sozer, M. Stojanovic, and J. G. Proakis, “Underwater acoustic networks,” IEEE Journal of Oceanic Engineering, vol. 25, no. 1, pp. 72–83, 2000.

[29] R. J. Urick, Principles of Underwater Sound, McGraw-Hill, 1983.

[30] R. J. Urick, Principles of Underwater Sound, Peninsula Publishing, 3rd edition, 1996.

[31] S. Shen, A. Zhan, P. Yang, and G. Chen, “Exploiting sink mobility to maximize lifetime in 3D underwater sensor networks,” in Proceedings of the IEEE International Conference on Communications (ICC ’10), pp. 1–5, Cape Town, South Africa, May 2010.

[32] P. Xie, Z. Zhou, Z. Peng et al., “Aqua-sim: an NS-2 based simulator for underwater sensor networks,” in Proceedings of the MTS/IEEE OCEANS 2009, Marine Technology for Our Future: Global and Local Challenges, pp. 1–7, Biloxi, Miss, USA, October 2009.
