

Article

A Mobile Anchor Node Assisted RSSI Localization Scheme in Underwater Wireless Sensor Networks

Yanlong Sun ¹, Yazhou Yuan ^{1,*}, Qimin Xu ², Changchun Hua ¹ and Xiping Guan ²

¹ Institute of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China; yanlongsun@stumail.ysu.edu.cn (Y.S.); cch@ysu.edu.cn (C.H.)

² Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China; qiminxu@sjtu.edu.cn (Q.X.); xpguan@sjtu.edu.cn (X.G.)

* Correspondence: zzyuan@ysu.edu.cn; Tel.: +86-335-8387-556

Received: 31 August 2019; Accepted: 8 October 2019; Published: 10 October 2019



Abstract: In this paper, a mobile anchor node assisted RSSI localization scheme in underwater wireless sensor networks (UWSNs) is proposed, which aims to improve location accuracy and shorten location time. First, to improve location accuracy, we design a support vector regression (SVR) based interpolation method to estimate the projection of sensor nodes on the linear trajectory of the mobile anchor node. The proposed method increases the accuracy of the nonlinear regression model of noisy measured data and synchronously decreases the estimation error caused by the discreteness of measured data. Second, to shorten location time, we develop a curve matching method to obtain the perpendicular distance from sensor nodes to the linear trajectory of the mobile anchor node. The location of the sensor node can be calculated based on the projection and the perpendicular distance. Compared with existing schemes that require the anchor node to travel at least two trajectories, the proposed scheme only needs one-time trajectory to locate sensor nodes, and the location time is shortened with the reduction in the number of trajectories. Finally, simulation results prove that the proposed scheme can obtain more accurate sensor node location in less time compared with the existing schemes.

Keywords: underwater wireless sensor networks (UWSNs); localization; mobile anchor node; RSSI

1. Introduction

With the rapid development of communication and sensor technology, underwater wireless sensor networks (UWSNs) have become a fast-growing field [1]. UWSNs have a wide range of applications, such as pollution monitoring, off-shore oil exploration, and oceanography data collection [2–4]. In UWSNs, localization is a fundamental issue, because only the data associated with the location of the sensor node can be meaningfully interpreted by data users [5]. Compared with the terrestrial localization issue, underwater localization has some fundamental differences. First, the Global Positioning System (GPS) technology is unavailable in the underwater environment, because of the strong attenuation of radio waves in water [6]. Thus, the underwater node cannot directly locate itself through the GPS. Second, due to the large propagation delay in underwater acoustic channels, the cost of maintaining clock synchronization is high and underwater nodes are usually clock asynchronous [7]. Third, the transmitting energy consumption of the underwater sensor node is several times or even hundreds of times the receiving energy consumption. To save energy, the underwater sensor node should reduce message transmission during the localization process [8].

The localization method based on Received Signal Strength Indicator (RSSI) does not require clock synchronization, which avoids the high cost of maintaining clock synchronization [9]. Localization methods require anchor nodes to provide reference locations [10]; since the cost of the anchor node

is much higher than that of the ordinary sensor node, it is expensive to deploy a large number of static anchor nodes in UWSNs [11]. Hence, using a small number of mobile anchor nodes to assist localization is an economical method [12]. Localization schemes [13–15] jointly use the RSSI and one mobile anchor node to reduce location cost. Guo [13] proposed a mobile anchor node-assisted localization scheme called Perpendicular Intersection (PI). By comparing the received RSSI values on a node, this scheme achieved the locating of nodes using the geometric relationship between the node and the trajectory of the mobile beacon. In this scheme, the error of determined projection in the localization process led to the error of the node location. To improve location accuracy, Pratap et al. [14] adopted a piecewise polynomial curve fitting method to reduce the error of determined projection. Zhao et al. [15] applied the polynomial curve fitting method and the derivative method to reduce the error of determined projection. Compared with the scheme in [13], the schemes in [14,15] improve the location accuracy. However, there are two points worthy of further study in [13–15]: (1) To improve location accuracy, is there any other better way to reduce the error of determined projection than existing schemes? (2) In existing schemes, the anchor node needs to travel at least two trajectories to locate the sensor node. Can the anchor node only travel one trajectory to shorten the location time?

Based on the above two considerations, in this paper, we propose a mobile anchor node assisted RSSI localization scheme in UWSNs. First, to improve location accuracy, we design a support vector regression (SVR) based interpolation method to estimate the projection of sensor nodes on the linear trajectory of the mobile anchor node. The proposed method increases the accuracy of the nonlinear regression model of noisy measured data and synchronously decreases the estimation error caused by the discreteness of measured data. The SVR based interpolation method achieves an improvement in location accuracy by reducing the error between the determined projection and the actual projection. Second, we develop a curve matching method to obtain the perpendicular distance from sensor nodes to the linear trajectory of the mobile anchor node. The location of the sensor node can be calculated based on the projection and the perpendicular distance. Compared with existing schemes that require the anchor node to travel at least two trajectories, the proposed scheme only needs one trajectory to locate sensor nodes, and the location time is shortened with the reduction in the number of trajectories. Finally, simulations prove that, compared with existing schemes, the proposed scheme can obtain more accurate sensor node location in less time. The main contributions of this paper are twofold:

- To improve location accuracy, we design an SVR based interpolation method to estimate the projection of sensor nodes on the linear trajectory of the mobile anchor node. This method increases the accuracy of the nonlinear regression model of noisy measured data and synchronously decreases the estimation error caused by the discreteness of measured data.
- To shorten location time, we develop a curve matching method to obtain the perpendicular distance from sensor nodes to the linear trajectory of the mobile anchor node. Compared with existing schemes, the proposed scheme only needs one trajectory of the mobile anchor node to locate sensor nodes, which significantly shortens the location time.

The remainder of this paper is organized as follows. Section 2 discusses the related work. In Section 3, we describe the details of the proposed localization scheme. Performance evaluations by simulation are presented in Section 4, followed by the conclusion and future work in Section 5.

2. Related Works

2.1. Overview of Localization Approaches

Generally speaking, localization approaches can be classified into range-free and range-based [16]. Range-free approaches usually employ the local topology connectivity to estimate the location of the sensor node and obtain a coarser location [17]. The authors of [18] proposed a range-free method, named Area Localization Scheme (ALS). In ALS, anchor nodes broadcast acoustic signals that can be transmitted at varying power levels, underwater sensor nodes passively listen to the signals and record the information, and the sink node can use the information to localize the sensor within

a certain area. In contrast, range-based approaches determine the location of sensor nodes using various ranging methods and provide an accurate location [19]. Ranging methods can be classified into four categories [20]: Time of Arrival (TOA) [21–24], Time Difference of Arrival (TDOA) [25–27], Angle of Arrival (AOA) [28–30] and Received Signal Strength Indicator (RSSI) [31–33]. In underwater environments, distance information obtained by the TOA method is usually used, such as wide coverage positioning system (WPS) [34], GPS-less localization protocol (GPS-less) [35], motion-aware sensor localization (MASL) [36], underwater sparse positioning (USP) [37] and 3D underwater localization (3DUL) [38]. TOA and TDOA methods require highly precise time synchronization of sensor nodes, which is hard to achieve due to the characteristics of acoustic signal when traveling in water [7]. The AOA method requires special devices for directional transmission and reception, which increase the cost of localization [19]. The RSSI method is cost-efficient and convenient because it does not require time synchronization and special devices. Since the RSSI value is vulnerable to environment disturbances, it is difficult to achieve accurate ranging by directly mapping the absolute RSSI value to the physical distance [9].

2.2. Localization Approaches Related to the Mobile Anchor Node

Both range-free and range-based localization approaches require anchor nodes, which are equipped with GPS units to provide reference locations [10]. Since the cost of the anchor node is much higher than that of the sensor node, it is expensive to deploy a large number of anchor nodes in UWSNs [11]. Hence, using mobile anchor nodes to assist nearby sensor nodes in localization is an economical and flexible approach [12]. In [39], a Dive and Rise (DNR) positioning method is presented, in which mobile anchor nodes are used to replace static ones. While sinking and rising, DNR anchor nodes broadcast their positions. Unknown nodes are localized by passively listening to DNR anchor nodes' messages. DNR method reduces communication overhead and energy consumption by passively listening, but localization performance heavily depend on frequency of location updates and the number of anchor nodes.

2.3. Localization Approaches Jointly Using the RSSI and the Mobile Anchor Node

Localization schemes [13–15] not only jointly use the RSSI and the mobile anchor node to reduce location cost, but also use a method of contrasting RSSI values to avoid the inaccurate ranging caused by directly mapping the RSSI value to the physical distance. In [13], a mobile anchor node moves along a linear trajectory and sends a message at each broadcast point on the trajectory. The sensor node measures the RSSI value of each received message. The broadcast point with the maximum measured RSSI value is approximately determined as the projection of the sensor node on the trajectory. Using two non-parallel trajectories, the location of the sensor node can be determined by calculating the intersection of perpendiculars passing through the determined projection on each trajectory. The location accuracy depends on the error between the determined projection and the actual projection. To improve location accuracy, different schemes are proposed to make the determined projection closer to the actual projection. Pratap et al. [14] tried to find an increasing as well as decreasing trends of measured RSSI values for each trajectory, and then used polynomial curve fitting for each trend, and determined the intersection of the increasing trend polynomial and the decreasing trend polynomial as the projection of the node on the trajectory. Zhao et al. [15] applied the polynomial curve fitting method to measured RSSI values, used the derivative method to calculate the point with the maximum value of the polynomial, and determined the point as the projection of the nodes on the trajectory. Compared with the scheme in [13], the schemes in [14,15] reduce the error between the determined projection and the actual projection, and improve the location accuracy. However, the error of the determined projection of existing schemes can be further reduced, and the location accuracy of the sensor node can be further improved. In addition, existing schemes require at least two trajectories; the number of trajectories can be reduced to one and the location time can be further shortened. In this

paper, we mainly research these two points, and the details of our proposed localization scheme are given in Section 3.

3. Detailed Description of the Proposed Localization Scheme

The proposed localization scheme consists of four steps: Firstly, the network system architecture and trajectory of the mobile anchor node are designed to obtain a set of measured RSSI values. Secondly, an SVR-based interpolation method is designed to process measured RSSI values to estimate the projection of a sensor node on the trajectory. Thirdly, a curve matching method is developed to process measured RSSI values to obtain the perpendicular distance from a sensor node to the trajectory. Finally, the location of the sensor node is calculated by a geometric method. Table 1 shows the main notations used in this paper.

Table 1. Main notations used in this paper.

Notation	Definition
L	The length of the trajectory
ΔL	The interval between adjacent broadcast points
m	The number of broadcast points on the trajectory
x_i	The x-coordinate of the i th broadcast point
\mathbf{X}	The location vector consisting of x-coordinates of all broadcast points
r_i	The noiseless RSSI value of the message from the i th broadcast point
\mathbf{R}	The noiseless RSSI value vector
\tilde{r}_i	The measured RSSI value of the message from the i th broadcast point
$\tilde{\mathbf{R}}$	The measured RSSI value vector
\hat{r}_i	The predicted RSSI value of the message from the i th broadcast point
$\hat{\mathbf{R}}$	The predicted RSSI value vector
\hat{x}_{ik}	The x-coordinate of the k th point inserted after the i th broadcast point
$\hat{\mathbf{X}}$	The extended location vector
\hat{r}_{ik}	The RSSI value of the k th point inserted after the i th broadcast point
$\hat{\mathbf{R}}$	The extended RSSI value vector
\mathbf{R}_i	The reference RSSI value vector of the i th reference trajectory
d_i	The perpendicular distance corresponding to the i th reference trajectory
$\lfloor \cdot \rfloor$	Floor function
$\tilde{\mathbf{X}}$	The vector formed by the location of all feature points
$\tilde{\mathbf{R}}$	The vector formed by the RSSI value of all feature points
$\bar{\mathbf{X}}$	The vector formed by the location of selected feature points
$\bar{\mathbf{R}}_i$	The vector formed by the RSSI value of selected feature points on the i th reference trajectory

3.1. Network System Architecture, Mobile Anchor Node Trajectory, and Obtaining a Set of Measured RSSI Values

3.1.1. Network System Architecture

Figure 1 illustrates the network system architecture of the proposed localization scheme, which consists of one mobile anchor node and a number of underwater sensor nodes. The mobile anchor node is a ship which patrols on the sea surface of the sensing area based on a planned trajectory. The mobile anchor node is fitted with a Global Positioning System (GPS) receiver, which can obtain its real-time geographic location. The mobile anchor node can broadcast messages through a spherical omnidirectional acoustic transceiver. Sensor nodes are deployed in the sensing area and stay at their locations by anchor chains and sub-surface buoys. Each sensor node is equipped with an acoustic transceiver, which can receive messages and measure the RSSI value of each message, and has a pressure sensor installed, which can obtain its depth information.

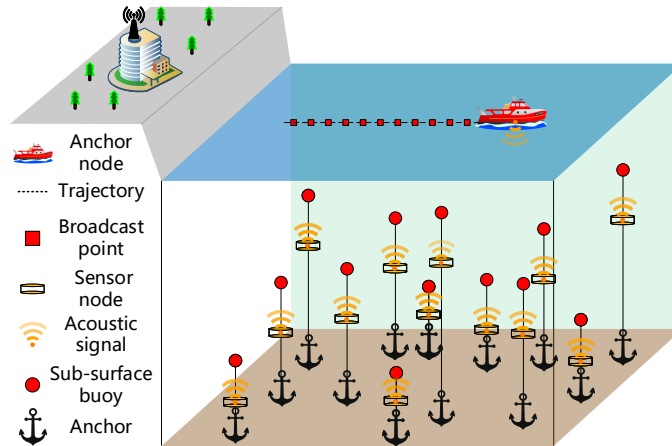


Figure 1. The network system architecture of the proposed localization scheme.

3.1.2. The Trajectory of the Mobile Anchor Node

As shown in Figure 2, the mobile anchor node starts at point x_1 , moves along a linear trajectory, and ends at point x_m . In the whole process, the mobile anchor node moves at a uniform speed and periodically broadcasts its real-time location message at a fixed transmission power. The length of the trajectory is L , the interval between adjacent broadcast points is ΔL , and the number of broadcast points on the trajectory is

$$m = L/\Delta L + 1 \tag{1}$$

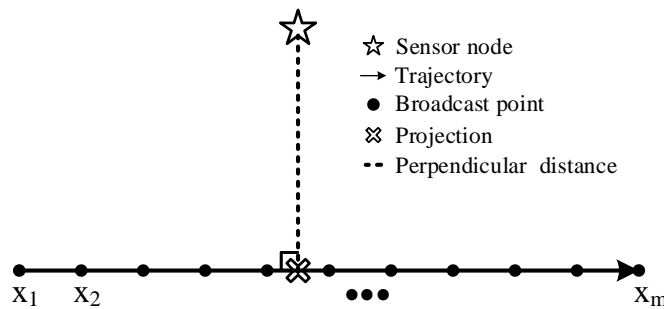


Figure 2. The trajectory of the mobile anchor node.

3.1.3. Obtaining a Set of Measured RSSI Values

When the mobile anchor node moves along the trajectory, the underwater sensor node receives messages and measures the RSSI value of each messages. The measured RSSI value in the underwater environment can be expressed as [40]

$$\tilde{r}(l) = SL - TL(l) + \zeta \tag{2}$$

where SL is source level, ζ is a random noise which is assumed to be a zero-mean gaussian random variable with variance σ^2 , and $TL(l)$ is the transmission loss for an underwater acoustic signal, which can be expressed as

$$TL(l) = 10 \cdot k \cdot \log(l) + \alpha \cdot l \tag{3}$$

where k is the spreading coefficient, α is the absorption coefficient, and l is the transmission distance, which is expressed as

$$l = \sqrt{(x - x_N)^2 + (y - y_N)^2 + (z - z_N)^2} \tag{4}$$

where (x, y, z) is the location of any point on the trajectory and (x_N, y_N, z_N) is the location of a sensor node. Since the trajectory is on the surface of the sea and is a straight line, z is 0, and y can be linearly

represented by x . We use the x -coordinate to indicate the location of broadcast points. The x -coordinate of all broadcast points of the trajectory can form an original location vector

$$\mathbf{X} = [x_1, x_2, \dots, x_m] \quad (5)$$

All measured RSSI values can also form a vector

$$\tilde{\mathbf{R}} = [\tilde{r}_1, \tilde{r}_2, \dots, \tilde{r}_m] \quad (6)$$

When ζ in Equation (2) is 0, a noiseless RSSI value vector is

$$\mathbf{R} = [r_1, r_2, \dots, r_m] \quad (7)$$

The sensor node uses \mathbf{X} and $\tilde{\mathbf{R}}$ to determine the projection of a sensor node on the trajectory and the perpendicular distance from a node to the trajectory, and then locates itself using the projection and the perpendicular distance.

3.2. Determining the Projection of a Node on the Trajectory

In this section, to determine the projection of a sensor node on the trajectory, a three-step method is designed, as shown in Figure 3. First, to model the nonlinear relationship between the arbitrary point location and the RSSI value on the trajectory, we perform SVR training on the discrete original location and the measured RSSI value to obtain a regression function. Then, to get more points and RSSI values, we extend the original location vector and the measured RSSI vector. An extended location vector is obtained by uniformly inserting new points in the original location vector, and the extended location vector is input to the regression function to predict the extended RSSI value vector. Finally, to select the point with the largest RSSI value, we compare the elements in the extended RSSI value vector. The point corresponding to the largest RSSI value is determined as the projection of the node on the trajectory.

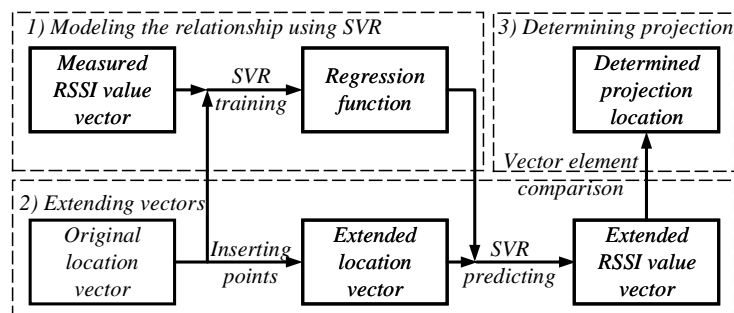


Figure 3. The process of determining the projection of a sensor node on the trajectory.

3.2.1. Modeling the Nonlinear Relationship Using SVR

According to Equations (2)–(4), there is a nonlinear relationship between \mathbf{X} and $\tilde{\mathbf{R}}$. Since SVR has a rather strong nonlinear approximation ability and antinoise ability [41], we use SVR to learn the regression function by identifying structures in the mapping of input \mathbf{X} to output $\tilde{\mathbf{R}}$.

The mobile anchor node starts at point x_1 , moves along a linear trajectory, and ends at point x_m . Then, the node to be located obtains data: the location of m broadcast points x_1, x_2, \dots, x_m , and the measured RSSI value $\tilde{r}_1, \tilde{r}_2, \dots, \tilde{r}_m$. The location of i th broadcast points and the i th measured RSSI value together form a data sample $\{(x_i, \tilde{r}_i) | x_i \in \mathbf{X}, \tilde{r}_i \in \tilde{\mathbf{R}}, i = 1, 2, \dots, m\}$. These m data samples are used as a dataset for SVR training. We need find an approximation function with the form

$$f(x) = w \cdot \varphi(x) + b \quad (8)$$

where $\varphi(x)$ is a mapping function, w is the weight, and b is the threshold. Define the ε -insensitive loss function as

$$|\tilde{r}_i - f(x_i)|_\varepsilon = \begin{cases} 0, & |\tilde{r}_i - f(x_i)| \leq \varepsilon \\ |\tilde{r}_i - f(x_i)| - \varepsilon, & |\tilde{r}_i - f(x_i)| > \varepsilon \end{cases} \tag{9}$$

By introducing slack variables ξ_i and ξ_i^* , the SVR problem is transformed into

$$\min_{w, b, \xi_i, \xi_i^*} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m (\xi_i + \xi_i^*) \tag{10}$$

subject to

$$\begin{cases} f(x_i) - \tilde{r}_i \leq \varepsilon + \xi_i \\ \tilde{r}_i - f(x_i) \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \\ i = 1, 2, \dots, m \end{cases} \tag{11}$$

where C is the penalty factor. By introducing Lagrange multipliers $\beta_i, \beta_i^*, \alpha_i, \alpha_i^* \geq 0$, the Lagrangian function of Equation (10) can be expressed as

$$\begin{aligned} \mathcal{L} = & \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m (\xi_i + \xi_i^*) - \sum_{i=1}^m \beta_i \xi_i - \sum_{i=1}^m \beta_i^* \xi_i^* \\ & + \sum_{i=1}^m \alpha_i (f(x_i) - \tilde{r}_i - \varepsilon - \xi_i) \\ & + \sum_{i=1}^m \alpha_i^* (\tilde{r}_i - f(x_i) - \varepsilon - \xi_i^*) \end{aligned} \tag{12}$$

Letting the partial derivative of a function \mathcal{L} with respect to the variable w, b, ξ_i, ξ_i^* be equal to 0, we can obtain

$$w = \sum_{i=1}^m \alpha_i^* x_i - \sum_{i=1}^m \alpha_i x_i \tag{13}$$

$$0 = \sum_{i=1}^m (\alpha_i^* - \alpha_i) \tag{14}$$

$$C = \alpha_i + \beta_i \tag{15}$$

$$C = \alpha_i^* + \beta_i^* \tag{16}$$

Substituting Equations (13)–(16) into Equation (12), we can obtain

$$\begin{aligned} \max_{\alpha, \alpha^*} & \left(-\frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m (\alpha_i^* - \alpha_i) (\alpha_j^* - \alpha_j) K(x_i, x_j) \right. \\ & \left. - \varepsilon \sum_{i=1}^m (\alpha_i^* + \alpha_i) + \sum_{i=1}^m \tilde{r}_i (\alpha_i^* - \alpha_i) \right) \end{aligned} \tag{17}$$

subject to

$$\begin{cases} \sum_{i=1}^m (\alpha_i^* - \alpha_i) = 0 \\ 0 \leq \alpha_i, \alpha_i^* \leq C \end{cases} \tag{18}$$

where $K(x_i, x_j)$ is a RBF kernel function

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{\sigma^2}\right) \tag{19}$$

KKT conditions for Equation (18) are

$$\begin{cases} \alpha_i (f(x_i) - \tilde{r}_i - \varepsilon - \zeta_i) = 0 \\ \alpha_i^* (\tilde{r}_i - f(x_i) - \varepsilon - \zeta_i^*) = 0 \\ \beta_i \zeta_i = 0, \beta_i^* \zeta_i^* = 0 \\ (C - \alpha_i) \zeta_i = 0, (C - \alpha_i^*) \zeta_i^* = 0 \end{cases} \tag{20}$$

The final regression function is obtained:

$$f(x) = \sum_{i=1}^m (\alpha_i^* - \alpha_i) K(x, x_i) + b \tag{21}$$

The regression function is the nonlinear relationship between \mathbf{X} and $\tilde{\mathbf{R}}$. We use the regression function to predict the location vector \mathbf{X} , and obtain the predicted RSSI vector

$$\hat{\mathbf{R}} = [\hat{r}_1, \hat{r}_2, \dots, \hat{r}_m] \tag{22}$$

We embed \mathbf{X} of Equation (5), $\tilde{\mathbf{R}}$ of Equation (6), \mathbf{R} of Equation (7), and $\hat{\mathbf{R}}$ of Equation (22) into Figure 4. The measured RSSI curve (black dashed line) is drawn using all the elements in $\tilde{\mathbf{R}}$. The predicted RSSI curve (red dash-dot line) is drawn using all the elements in $\hat{\mathbf{R}}$. The noiseless RSSI curve (purple solid line) is drawn using all the elements in \mathbf{R} . Since the SVR has a good noise reduction effect, the predicted RSSI curve should be closer to the noiseless RSSI curve than the measured RSSI curve.

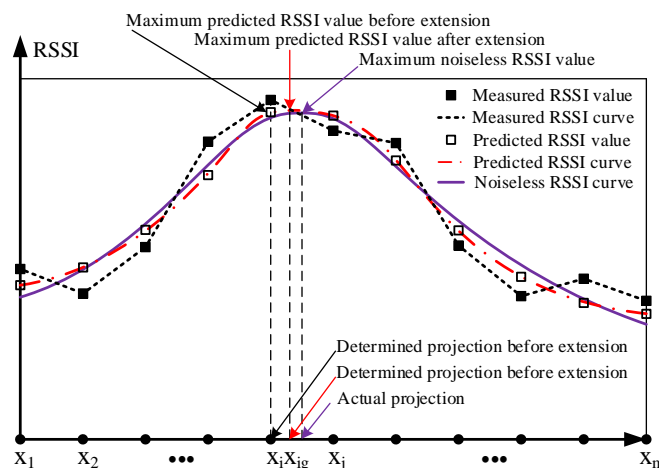


Figure 4. Determining the projection of a node on the trajectory by the SVR-based interpolation method.

3.2.2. Extending Vectors

Since the trained regression function by SVR is a black box, it can transform the input into the output, but the inner working is difficult to interpret because the underlying models are based on complex mathematical systems [42]. The derivative method cannot be used to find the point with the maximum value on the predicted RSSI curve. We extend the location vector \mathbf{X} and predicted RSSI vector $\hat{\mathbf{R}}$ to determine a point closer to the actual projection.

As shown in Figure 4, the predicted RSSI curve has first and second maximum RSSI values at points x_i and x_j , respectively. Before extending vectors, x_i is determined as the projection, but has a large error with the actual projection. Although shortening the distance between broadcast points can reduce the error, it is difficult to reduce this distance to a small value in an actual sea environment. To determine a point closer to the actual projection position, we evenly insert k points between x_i and x_j , as shown in Figure 5. The extended location vector is

$$\hat{\mathbf{X}} = [\hat{x}_1, \dots, \hat{x}_i, \hat{x}_{i1}, \hat{x}_{i2}, \dots, \hat{x}_{ik}, \hat{x}_j, \dots, \hat{x}_m] \tag{23}$$

After using the regression function to predict all the elements in $\hat{\mathbf{X}}$, the extended RSSI vector can be obtained

$$\hat{\mathbf{R}} = [\hat{r}_1, \dots, \hat{r}_i, \hat{r}_{i1}, \hat{r}_{i2}, \dots, \hat{r}_{ik}, \hat{r}_j, \dots, \hat{r}_m] \tag{24}$$

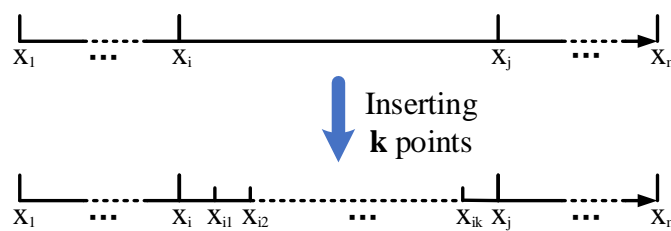


Figure 5. The extension of the location vector.

3.2.3. Determining the Projection

We find the maximum element in $\hat{\mathbf{R}}$ by the element comparison, and determine the corresponding element in $\hat{\mathbf{X}}$ as the projection of a node on the trajectory. As shown in Figure 4, the predicted RSSI curve has the maximum predicted RSSI value at x_{ig} after vectors are extended, and x_{ig} is determined as the projection of the node. x_{ig} is closer to the actual projection than x_i , and the error between the determined projection and the actual projection is reduced.

3.3. Determining the Perpendicular Distance from a Node to the Trajectory

In this section, to determine the perpendicular distance from a node to the trajectory, we design a three-step method, as shown in Figure 6. First, to establish the reference RSSI curve library, we calculate the RSSI value of the broadcast point on n reference trajectories. Then, to accurately compare the similarities between curves, we extract the parts to be compared from the predicted curve and reference curves, and select the feature points on the extracted parts, so as to get the feature point vectors for comparison. Finally, to select the feature point vector with the highest similarity, we compare the similarity between the feature point vectors. The distance corresponding to the selected feature point vector is determined as the perpendicular distance from the node to the trajectory.

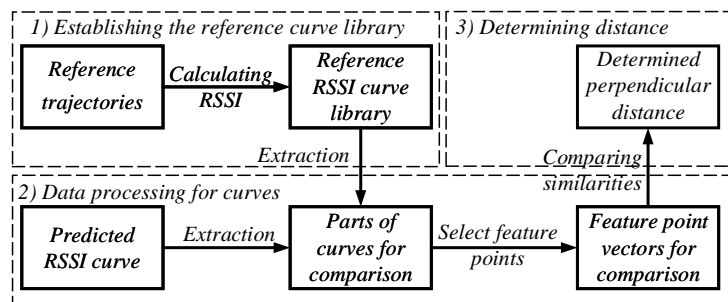


Figure 6. The process of determining the perpendicular distance from a node to the trajectory.

3.3.1. Establishing the Reference RSSI Curve Library

As shown in Figure 7a, there are n parallel and identical trajectories with m broadcast points on each trajectory. The spacing between adjacent trajectories is Δd , and the interval between adjacent broadcast points is ΔL . The projection of the reference node on all the trajectories are their respective midpoints. The perpendicular distance from the reference node to trajectory 1, trajectory 2, ..., trajectory n are d_1, d_2, \dots, d_n , respectively. When the random noise ζ in Equation (2) is 0, we name the RSSI value received by the reference node as the reference RSSI value. Each trajectory corresponds to a reference RSSI curve, as shown in Figure 7b. For the i th trajectory, a reference RSSI vector \mathbf{R}_i can be constructed using the reference RSSI value of the broadcast point at x_1, x_2, \dots, x_m ,

$$\mathbf{R}_i = [r_{i1}, r_{i2}, \dots, r_{im}] \tag{25}$$

We use \mathbf{R}_i to represent the i th reference RSSI curve. All reference RSSI curves can establishing a reference RSSI curve library, which can be expressed as

$$\mathbf{L}_c = \begin{cases} \mathbf{R}_1 = [r_{11}, r_{12}, \dots, r_{1m}] \\ \mathbf{R}_2 = [r_{21}, r_{22}, \dots, r_{2m}] \\ \vdots \\ \mathbf{R}_n = [r_{n1}, r_{n2}, \dots, r_{nm}] \end{cases} \tag{26}$$

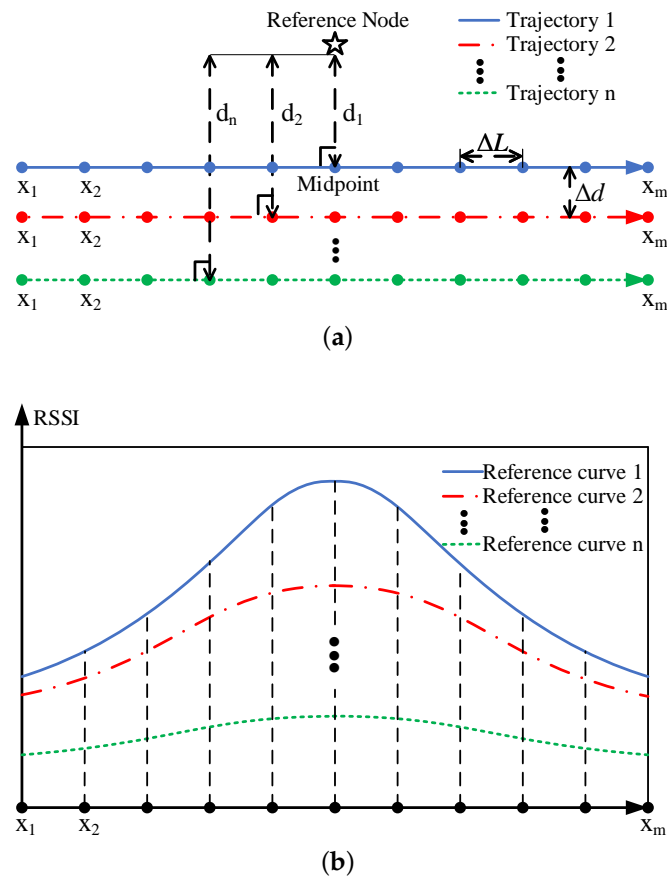


Figure 7. The establishment of the reference RSSI curve library: (a) the positional relationship between the reference node and n trajectories; and (b) the reference RSSI curve library.

Since the perpendicular distance from the node to each trajectory is different, the transmission distance from the same broadcast point of each trajectory to the reference node are different, the reference RSSI values of the respective trajectories at the same broadcast point are different, and the reference RSSI vector of each trajectory are different. Therefore, there is a mapping relationship between the perpendicular distance from the reference node to the trajectory and the reference RSSI curve, which can be expressed as

$$d_i = F(\mathbf{R}_i) \quad (27)$$

where $i = 1, 2, \dots, n$, F represents the mapping relationship. All mapping relationships between the reference RSSI vector and its perpendicular distance can establish a mapping relationship library, which can be expressed as

$$\mathbf{L}_r = \begin{cases} d_1 = F(\mathbf{R}_1) \\ d_2 = F(\mathbf{R}_2) \\ \vdots \\ d_n = F(\mathbf{R}_n) \end{cases} \quad (28)$$

3.3.2. Data Processing for Curves Used for Comparison

Three steps of data processing are required before the measured RSSI curve is compared with each curve in the reference RSSI curve library. Firstly, the measured RSSI value contains noise which would cause a large error in the result of the curve matching. Since SVR has a strong ability [41], we use the predicted RSSI curve instead of the measured RSSI curve to perform curve matching.

Secondly, the projection of a node on the trajectory is difficult to coincide with the midpoint of the trajectory, which causes the predicted RSSI curve to be offset from the reference RSSI curve. This offset causes us to extract only a part of the predicted RSSI curve to compare with the corresponding part of the reference RSSI curve.

As shown in Figure 8, the projection of the node has three positional relationships with the midpoint of the trajectory: on the left (Figure 8a), coincident (Figure 8b), and on the right (Figure 8c). When the projection of the node is on the left of the midpoint, the predicted RSSI curve has a left offset relative to the reference RSSI curve. The predicted RSSI curve can be divided into four parts: S1, S2, S3, and S4. The reference RSSI curve can be divided into three parts: R1, R2, and R3. The S1 part of the predicted RSSI curve corresponds to the R1 part of the reference RSSI curve. However, the left offset of the curve causes the trajectory of the S1 part to exceed the actual travel range of the mobile anchor node, that is, the S1 part does not exist, and does not need to be compared with R1. The trajectory of the S4 part is within the range of actual travel of the mobile anchor node. However, the distance from the node to the point on the S4 part of the trajectory is too large, so that there is no part on the reference RSSI curve that can be compared with S4. Therefore, when the projection of the node is on the left of the midpoint, S2 + S3 of the predicted RSSI curve are compared with R2 + R3 of the reference RSSI curve. Similarly, when the node projection coincides with the midpoint, S1 + S2 of the predicted RSSI curve are compared with R1 + R2 of the reference RSSI curve. When the projection of the node is on the right of the midpoint, S2 + S3 of the predicted RSSI curve are compared with R1 + R2 of the reference RSSI curve.

Thirdly, we need select feature points on the predicted RSSI curve to compare with the corresponding feature points on the reference RSSI curve. As shown in Figure 8a, when the projection of the node is on the left of the midpoint, assuming that the projection of the node on the trajectory is x_{pro} , the midpoint of the trajectory is x_{mid} . The offset of the predicted RSSI curve from the reference RSSI curve is

$$L_{off} = |x_{mid} - x_{pro}| \quad (29)$$

Starting from the projection, we select a feature point every ΔL in the S2 part and the S3 part, respectively. The number of feature points in the S2 part, the S3 part, the S2 + S3 part are NL , NR , and NT , respectively. They can be expressed as

$$\begin{cases} NL &= \lfloor (L/2 - L_{off})/\Delta L \rfloor \\ NR &= \lfloor (L/2)/\Delta L \rfloor \\ NT &= NL + 1 + NR \end{cases} \quad (30)$$

The location of the leftmost feature point is

$$\check{x}_1 = x_{pro} - NL * \Delta L \quad (31)$$

and the location of any feature point in S2 + S3 is

$$\check{x}_i = \check{x}_1 + (i - 1) * NT * \Delta L \quad (32)$$

where $i = 1, 2, \dots, NT$.

The vector formed by the location of each feature point is

$$\check{\mathbf{X}} = [\check{x}_1, \check{x}_2, \dots, \check{x}_{NT}] \quad (33)$$

We use the obtained regression function in Section 3.2 to predict each element in $\check{\mathbf{X}}$, and obtain the RSSI value vector of the feature point, which can be expressed as

$$\check{\mathbf{R}} = [\check{r}_1, \check{r}_2, \dots, \check{r}_{NT}] \quad (34)$$

We select the broadcast point of the R2 + R3 of the reference RSSI curve as the location of the feature point, and the RSSI value corresponding to the broadcast point as the value of the feature point. The location vector $\bar{\mathbf{X}}$ and RSSI value vector $\bar{\mathbf{R}}$ formed by the feature points on the R2 + R3 of the i th reference RSSI curve are

$$\begin{cases} \bar{\mathbf{X}} &= [x_{m-(N_i-1)}, x_{m-(N_i-2)}, \dots, x_m] \\ \bar{\mathbf{R}}_i &= [r_{i(m-(N_i-1))}, r_{i(m-(N_i-2))}, \dots, r_{im}] \end{cases} \quad (35)$$

$\check{\mathbf{R}}$ and $\bar{\mathbf{R}}_i$ are used for similarity comparison. When the node projection coincides with the midpoint, $\check{\mathbf{R}}$ and $\bar{\mathbf{R}}_i$ are

$$\begin{cases} \check{\mathbf{R}} &= [\hat{r}_1, \hat{r}_2, \dots, \hat{r}_m] \\ \bar{\mathbf{R}}_i &= [r_{i1}, r_{i2}, \dots, r_{im}] \end{cases} \quad (36)$$

When the projection of the node is to the right of the midpoint, some equations that differ from the projection on the left of the midpoint are

$$NL = \lfloor (L/2)/\Delta L \rfloor \quad (37)$$

$$NR = \lfloor (L/2 - L_{off})/\Delta L \rfloor \quad (38)$$

$$\bar{\mathbf{R}}_i = [r_{i1}, r_{i2}, \dots, r_{i(m-(NT-1))}] \quad (39)$$

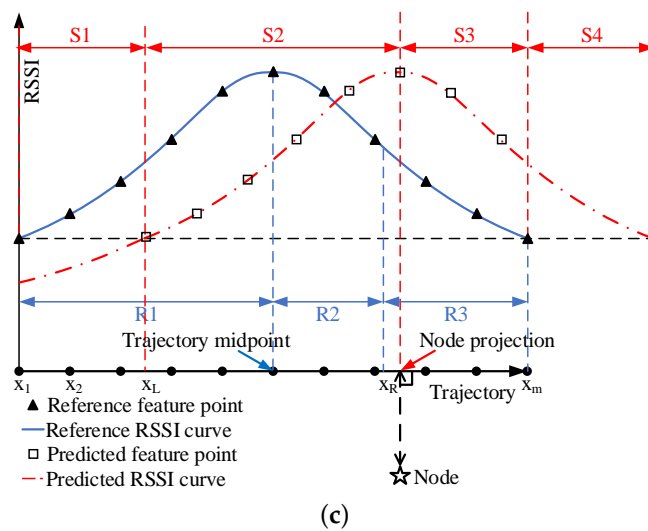
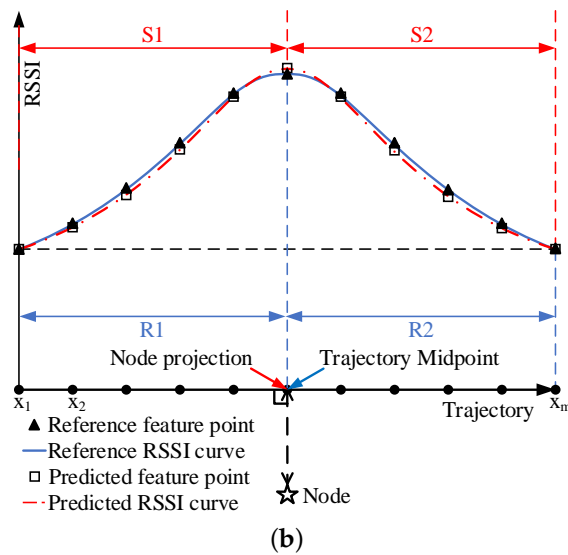
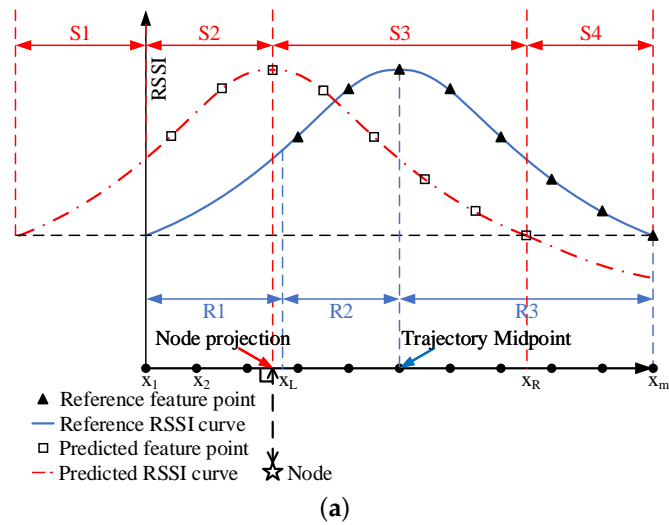


Figure 8. Three positional relationships between the projection of the node and the midpoint of the trajectory: (a) the projection of the node is on the left of the midpoint; (b) the node projection coincides with the midpoint; and (c) the projection of the node is on the right of the midpoint.

3.3.3. Determining the Perpendicular Distance by Comparing Curve Similarities

We need compare the similarity between $\check{\mathbf{R}}$ and each $\bar{\mathbf{R}}_i$, find the $\bar{\mathbf{R}}_i$ that is most similar to $\check{\mathbf{R}}$, and determine the d_i corresponding to the $\bar{\mathbf{R}}_i$ as the perpendicular distance from the node to the trajectory. Before calculating the similarity, it is necessary to normalize the elements of the vector to avoid the weight of the elements with larger absolute values in the similar calculation results being too large. We combine the $\check{\mathbf{R}}$ with all $\bar{\mathbf{R}}_i$ to construct a new matrix

$$\mathbf{A} = [\check{\mathbf{R}}, \bar{\mathbf{R}}_1, \bar{\mathbf{R}}_2, \dots, \bar{\mathbf{R}}_n]^T \quad (40)$$

x is the element at the i th row and j th column of the matrix \mathbf{A} , and the normalized value of x is

$$y = (x - x_{mean}) \times (y_{std}/x_{std}) + y_{mean} \quad (41)$$

where x_{mean} is the mean value for the j th column of the matrix \mathbf{A} , x_{std} is the standard deviation for the j th column of the matrix \mathbf{A} , y_{mean} is 0, and y_{std} is 1. The normalized vector $\check{\mathbf{R}}$ is represented as $\check{\mathbf{R}}'$, the normalized vector $\bar{\mathbf{R}}_i$ is represented as $\bar{\mathbf{R}}'_i$. We compare the similarity of $\check{\mathbf{R}}'$ and $\bar{\mathbf{R}}'_i$ by calculating the Euclidean distance:

$$E_i = \|\check{\mathbf{R}}' - \bar{\mathbf{R}}'_i\| \quad (42)$$

The smaller is the value of E_i , the higher is the similarity between vectors. The reference RSSI curve with the smallest euclidean distance is most similar to the predicted RSSI curve. According to Equation (27), the perpendicular distance corresponding to the reference curve is determined as the perpendicular distance from the node to the trajectory.

3.4. Calculating the Node Location

In this section, the location of the sensor node is determined using the obtained projection in Section 3.2 and perpendicular distance in Section 3.3. We first transform the three-dimensional (3D) underwater sensor node localization problem into a two-dimensional (2D) localization problem. Then, we describe in detail the calculation process of localization.

3.4.1. Transforming 3D Localization into 2D Localization

As shown in Figure 9, the to-be-localized node A is at the lower right of the trajectory, B is the projection of node A on the trajectory, and C is the projection of node A on the trajectory plane. The coordinates of point A, B, and C are (x_A, y_A, z_A) , $(x_B, y_B, 0)$, and $(x_C, y_C, 0)$, respectively. $(x_B, y_B, 0)$ can be obtained by the method in Section 3.2. d_{AB} is the perpendicular distance from node A to the trajectory, which can be obtained by the method in Section 3.3. d_{AC} or z_A is the depth of node A, which can be measured by a pressure sensor. d_{BC} is the projection of d_{AB} on the plane, as well as the perpendicular distance from C to the trajectory. $\triangle ABC$ is a right-angle triangle, and d_{BC} can be calculated as

$$d_{BC} = \sqrt{d_{AB}^2 - d_{AC}^2} \quad (43)$$

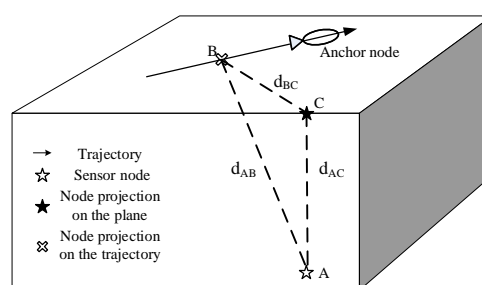


Figure 9. Transforming 3D localization into 2D localization.

After the above operations, the task of localizing node A in a 3D space has been transformed into localizing C in a 2D plane.

3.4.2. The Calculation Process of Localization

The equation of the trajectory can be expressed as

$$y = k \times x + b \quad (44)$$

k and b can be obtained by bringing the coordinates of two broadcast points into Equation (44). The coordinates of point C can be computed by solving

$$\begin{cases} d_{BC} = \frac{|k \times x_C - y_C + b|}{\sqrt{k^2 + 1}} \\ \frac{y_C - y_B}{x_C - x_B} = -\frac{1}{k} \end{cases} \quad (45)$$

However, solving Equation (45) will get two candidate positions, as shown in Figure 10. To determine the true position of point C, when the anchor node reaches the end of the trajectory, it continues to travel a distance along the right perpendicular line of the trajectory. If RSSI values received by the node become larger relative to the RSSI value of the end point, point C in Figure 10 is the true position. If RSSI values received by the node become smaller relative to the RSSI value of the end point, then point C' in Figure 10 is the true position. After obtaining the true coordinates of point C, the coordinates of point A can be determined as (x_C, y_C, z_A) .

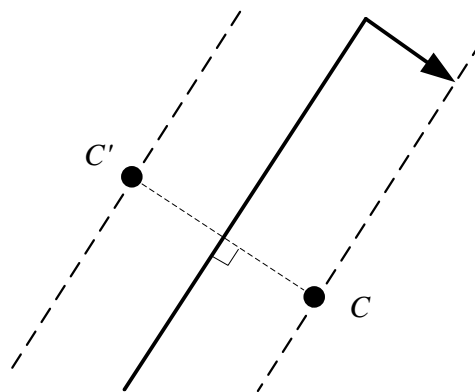


Figure 10. Judging the location of the point C.

4. Simulation

The performance of the proposed schemes was evaluated through MATLAB simulations. Since the proposed scheme addresses the shortcomings of the PI scheme and the DuRT scheme, we compared our scheme with the PI scheme [13] and the DuRT scheme [14] in the simulation. In [14], the DuRT scheme actually includes three localization methods, DuRT-M1, DuRT-M2, and DuRT-ALL, and the working conditions of the three methods are different. In our simulation, the conditions for using DuRT-M1 were satisfied, but the conditions for using DuRT-M2 and DuRT-ALL were not satisfied. Thus, in the simulation, we used DuRT-M1. For convenience, DuRT-M1 is denoted as DuRT.

4.1. Simulation Parameter Setting

Since the PI scheme and DuRT scheme require two trajectories, we designed a triangular prism sensing area. The top of the triangular prism is an equilateral triangle, the left side of the top triangle is the trajectory 1 of the mobile anchor node, and the right side of the top triangle is the trajectory 2. The length of the trajectory is $L = 600$ m, and the depth of the triangular prism is $H = 100$ m.

$N = 100$ nodes are randomly deployed in a triangular prism. The interval between adjacent broadcast points ΔL was set to 1 m, 5 m, 10 m, 15 m, and 20 m. The interval between adjacent extended broadcast points is $\Delta L_{ext} = 0.5$ m. The spacing between adjacent reference trajectories Δd is 1 m. The acoustic parameters are: $SL = 100$ db, $f = 24$ KHz, and $k = 2$. The speed of the ship is $v = 5$ m/s. In the localization process of this paper, the underwater sensor node does not need to transmit messages, thus the coverage distance of the underwater sensor node is short and is set to $cd_s = 100$ m. The mobile anchor node needs a large enough coverage distance so that the transmitted message can be received by all sensor nodes in the area, and is set to $cd_a = 600$ m. For the convenience of viewing, the parameter settings of the simulation are as shown in Table 2.

Table 2. Parameters used in the simulation.

Parameter	Value	Parameter	Value
L	600 m	H	100 m
N	100	ΔL	1, 5, 10, 15, 20 m
ΔL_{ext}	0.5 m	Δd	1 m
SL	100 db	f	24 KHz
k	2	v	5 m/s
cd_s	100 m	cd_a	600 m

4.2. Simulation Results and Discussion

In this section, we provide the system architecture used in the simulation, and analyze simulation results from three aspects. First, we use three schemes to determine the projection of the node, and compare the mean error of node projections. Then, we compare the measured RSSI curve with the selected RSSI curve, and analyze the error of the perpendicular distance. Finally, we compare the mean location error of the three schemes.

4.2.1. The Network System Architecture in the Simulation

As shown in Figure 11, 100 nodes (circles of various colors) are randomly deployed in the triangular prism area. The mobile anchor node starts at $(0, 0, 100)$, moves along trajectory 1 (blue solid line) and trajectory 2 (blue dotted line), and ends at $(600, 0, 100)$. After the mobile anchor node finishes moving, underwater sensors use receiving messages to locate themselves. The localization process needs to run five times. In the five localization processes, the interval between adjacent broadcast points ΔL is set to 1 m, 5 m, 10 m, 15 m, and 20 m, respectively.

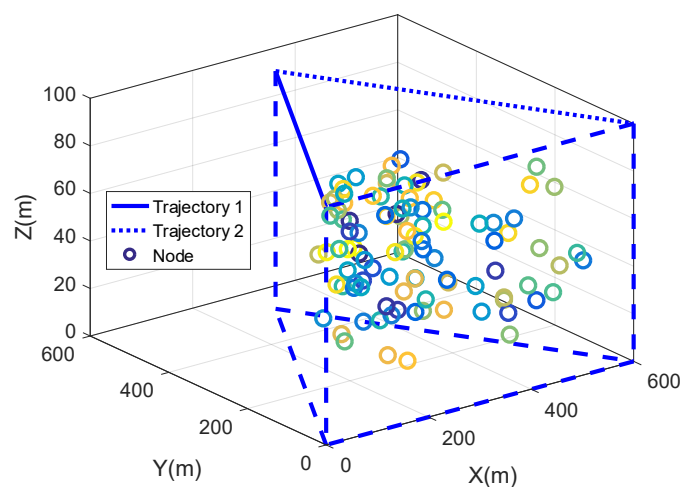


Figure 11. The network system architecture in the simulation.

4.2.2. The Error of a Node Projection on the Trajectory

The PI scheme, the RuDT scheme and the scheme proposed in this paper must first determine the projection position of the node on the trajectory. Therefore, we first compare the error of one node projection of different schemes. As shown in Figure 12, the actual projected position of the node on the trajectory is the position corresponding to the highest point of the curve without noise (Blue curve). The PI scheme selects the position with the largest value on the RSSI curve with noise (Magenta curve with diamond) as the node projection. The DuRT scheme uses a polynomial fitting method to fit an increased curve (Green dotted line) to the left part and a decreased curve (Light blue dotted line with circles) to the right part. The position corresponding to the intersection of the increased curve and the decreased curve is the node projection determined by the DuRT scheme. The scheme in this paper uses SVR to fit a curve and select the position corresponding to the maximum value of this curve as the node projection. The actual projection position is 141. The projection positions determined by the PI scheme, the DURT scheme, and the scheme in this paper are 130, 155, and 140, respectively. The scheme in this paper has the least error of the three schemes.

As shown in Figure 13, we compare the mean error of node projections for 100 nodes at different broadcast point intervals in three schemes. The mean error of the three schemes increases as the broadcast point interval increases. In the case of the same broadcast point interval, the mean error of the three schemes is sorted from biggest to smallest: the PI scheme, the RuDT scheme, and the scheme proposed in this paper. Due to the influence of noise, the broadcast position with the largest RSSI value is often not the closest broadcast position to the actual projection, as shown in Figure 13. Since the PI scheme only selects the broadcast position with the largest RSSI value, the PI scheme has a large error in determining the node projection. Although the polynomial curve fitting method can fit all RSSI values, the fitting effect is poor. Thus, the DuRT scheme fits the RSSI values on the left and right sides separately. When the left polynomial and the right polynomial have an intersection between the broadcast position with the largest RSSI value and the broadcast position with the second largest RSSI value, the DuRT scheme selects this intersection as the node projection. However, this intersection can only be controlled between two broadcast locations, and it is not possible to control the intersection to be close to the actual projection. Thus, the error of the DuRT scheme is smaller than the error of the PI method, but it is still very large. The scheme in this paper solves the noise problem and fits a good curve to all RSSI values. Therefore, our scheme is an effective scheme for reducing the error of the determined node projection.

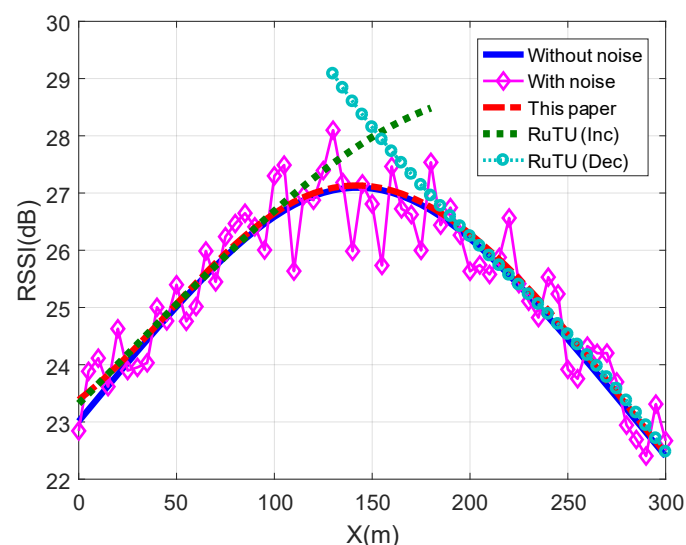


Figure 12. Determining the projection of one node on the trajectory by three schemes ($\Delta L = 10$ m).

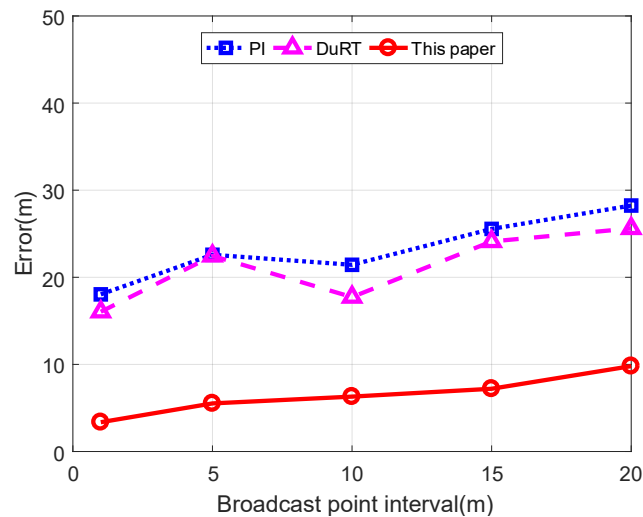


Figure 13. Mean error of the projection of the node on the trajectory determined by different schemes.

4.2.3. The Error of the Perpendicular Distance from the Node to the Trajectory

In Figure 14, the blue curve with circles is the measured curve after data processing, and the red curve with triangles is the selected reference curve after data processing. After the normalization operation, the range of the RSSI curve is reduced, which narrows the weight difference between the comparison results of each feature point in all comparison results. At the same time, the shape of the two curves has been greatly changed from that of Figure 12 after normalization. The total length of the X-axis of the two curves is shortened from 300 to 60, indicating that only a part of the curve is compared. The marks on the two curves are the selected feature points. As shown in Figure 14, the measured curve is very similar to the selected reference curve, so the perpendicular distance corresponding to the reference curve can be determined as the perpendicular distance from the node to the trajectory.

As shown in Figure 15, the mean error of the perpendicular distance does not become larger as the broadcast point interval becomes larger, and it is greater than the mean error of the projection.

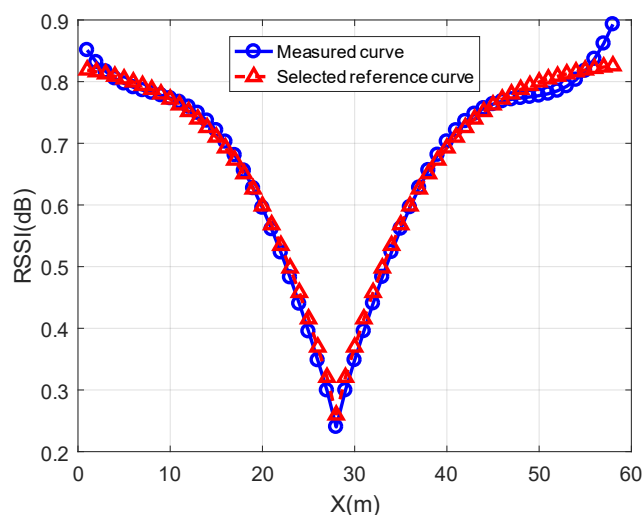


Figure 14. Comparison of curves after data processing.

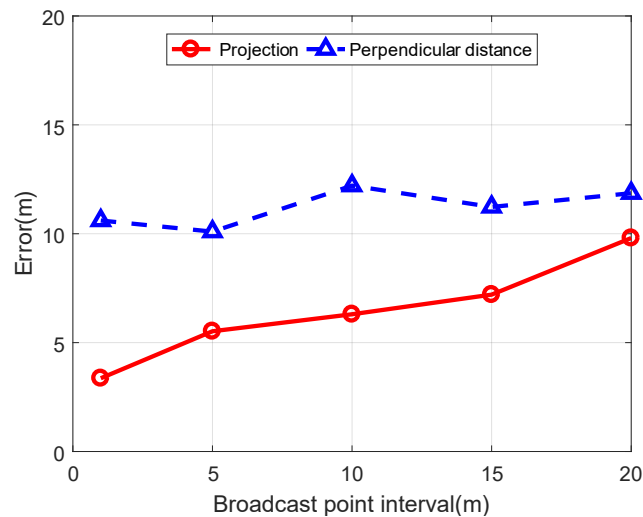


Figure 15. Comparison of mean error between node projection and Perpendicular distance.

4.2.4. The Error of the Node Location

Figure 16 shows the comparison of the mean location error of the three schemes. In the case of different broadcast point intervals, the location error of the three schemes becomes larger as the broadcast point interval becomes larger. In the case of the same broadcast position interval, the location errors of the three schemes are in descending order: PI scheme, DuRT scheme, and the scheme in this paper. This sorting indicates that the localization scheme proposed in this paper is far superior to the PI scheme and the DuRT scheme.

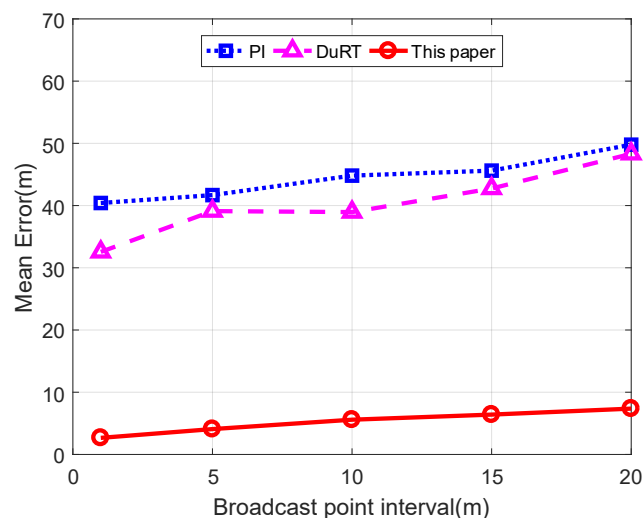


Figure 16. Comparison of the mean location error of the three schemes.

5. Conclusions and Future Work

In this paper, a novel mobile anchor node assisted RSSI localization scheme for UWSNs is proposed. After one mobile anchor travels along the planned linear trajectory on the water surface, all underwater sensor nodes in the area near the trajectory can locate their positions. We design an SVR-based interpolation method to improve the technique of determining node projection in existing methods. This method increases the accuracy of the nonlinear regression model of noisy measured data and decreases the estimation error caused by the discreteness of measured data, which greatly reduces the error of the determined projection and then improves the location accuracy of the underwater sensor node. By developing a curve matching method to obtain the perpendicular distance from the sensor nodes to the linear trajectory, the location of the sensor node can be calculated based on the

determined projection and the obtained perpendicular distance. Compared with existing schemes that require the anchor node to travel at least two trajectories, the proposed scheme only needs one trajectory to locate sensor nodes, and the location time is shortened with the reduction in the number of trajectories. Simulation results show the validity of our proposed localization scheme, the SVR-based interpolation method, and the curve matching method. In the future, we aim to enhance the curve matching method to reduce the error between the obtained perpendicular distance and the actual perpendicular distance, so as to further improve the location accuracy.

Author Contributions: Conceptualization, Y.S. and Y.Y.; Data curation, Y.S.; Funding acquisition, Y.Y.; Investigation, Y.S. and Y.Y.; Methodology, Y.S. and Y.Y.; Software, Y.S.; Supervision, Y.Y.; Validation, Y.S.; Visualization, Y.S.; Writing—original draft, Y.S.; and Writing—review and editing, Q.X., C.H. and X.G.

Funding: This work was supported in part by National Natural Science Foundation of China (61803328) and in part by the Natural Science Foundation of Hebei Province (F201703084).

Conflicts of Interest: The authors declare no conflict of interest.

References

- Ghoreyshi, S.M.; Shahrabi, A.; Boutaleb, T. Void-handling techniques for routing protocols in underwater sensor networks: Survey and challenges. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 800–827. [[CrossRef](#)]
- Sun, Y.; Yuan, Y.; Xu, Q.; Guan, X. An adaptive sampling algorithm for target tracking in underwater wireless sensor networks. *IEEE Access* **2018**, *6*, 68324–68336. [[CrossRef](#)]
- Felemban, E.; Shaikh, F.K.; Qureshi, U.M.; Sheikh, A.A.; Qaisar, S.B. Underwater Sensor Network Applications: A Comprehensive Survey. *Int. J. Distrib. Sens. Netw.* **2015**, *11*, 896832. [[CrossRef](#)]
- Kim, Y.; Noh, Y.; Kim, K. RAR: Real-time Acoustic Ranging in Underwater Sensor Networks. *IEEE Commun. Lett.* **2017**, *21*, 2328–2331. [[CrossRef](#)]
- Erol-Kantarci, M.; Mouftah, H.; Oktug, S. A survey of architectures and localization techniques for underwater acoustic sensor networks. *IEEE Commun. Surv. Tutor.* **2015**, *13*, 487–502. [[CrossRef](#)]
- Yan, J.; Xu, Z.; Wan, Y. Consensus estimation-based target localization in underwater acoustic sensor networks. *Int. J. Robust Nonlinear Control* **2017**, *27*, 1607–1627. [[CrossRef](#)]
- Yan, J.; Zhang, X.; Luo, X. Asynchronous Localization with Mobility Prediction for Underwater Acoustic Sensor Networks. *IEEE Trans. Veh. Technol.* **2017**, *67*, 2543–2556. [[CrossRef](#)]
- Coutinho, R.; Boukerche, A.; Vieira, L.; Loureiro, A.A.F. Geographic and opportunistic routing for underwater sensor networks. *IEEE Trans. Comput.* **2016**, *65*, 548–561. [[CrossRef](#)]
- Xu, T.; Hu, Y.; Zhang, B.; Leus, G. RSS-based sensor localization in underwater acoustic sensor networks. In Proceedings of the ICASSP, Shanghai, China, 20–25 March 2016; pp. 3906–3910.
- Han, G.; Yang, X.; Liu, L.; Guizani, M.; Zhang, W. A Disaster Management-Oriented Path Planning for Mobile Anchor Node-Based Localization in Wireless Sensor Networks. *IEEE Trans. Emerg. Top. Comput.* **2019**. [[CrossRef](#)]
- Halder, S.; Ghosal, A. A survey on mobility-assisted localization techniques in wireless sensor networks. *J. Netw. Comput. Appl.* **2016**, *60*, 82–94. [[CrossRef](#)]
- Han, G.; Zhang, C.; Jiang, J.; Yang, X.; Guizani, M. Mobile anchor nodes path planning algorithms using network-density-based clustering in wireless sensor networks. *J. Netw. Comput. Appl.* **2017**, *85*, 64–75. [[CrossRef](#)]
- Guo, Z.; Guo, Y.; Hong, F.; Yang, X.; He, F.; Liu, Y. Perpendicular Intersection: Locating Wireless Sensors With Mobile Beacon. *IEEE Trans. Veh. Technol.* **2010**, *59*, 3501–3509. [[CrossRef](#)]
- Pratap, K.S.; Wu, E.H.K.; Jagruti, S. DuRT: Dual RSSI Trend Based Localization for Wireless Sensor Networks. *IEEE Sens. J.* **2013**, *13*, 3115–3123.
- Zhao, Y.; Xu, J.; Jiang, J.; Leus, G. RSSI Based Localization with Mobile Anchor for Wireless Sensor Networks. In Proceedings of the GSKI 2018, Wuhan, China, 23–29 September 2018; pp. 176–187.
- Luo, J.; Fan, L.; Wu, S.; Yan, L. Research on Localization Algorithms Based on Acoustic Communication for Underwater Sensor Networks. *Sensors* **2018**, *18*, 67. [[CrossRef](#)] [[PubMed](#)]
- Han, G.; Jiang, J.; Zhang, C.; Duong, T.Q. A survey on mobile anchor node assisted localization in wireless sensor networks. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 2220–2243. [[CrossRef](#)]

18. Chandrasekhar, V.; Seah, W. An area localization scheme for underwater sensor networks. In Proceedings of the OCEANS 2006, Singapore, 16–19 May 2006; pp. 1–8.
19. Zhou, Z.; Cui, J.-H.; Zhou, S. Efficient Localization for Large-Scale Underwater Sensor Networks. *Ad Hoc Netw.* **2010**, *3*, 267–279. [[CrossRef](#)]
20. Han, G.; Jiang, J.; Shu, L.; Xu, Y.; Wang, F. Localization algorithms of Underwater Wireless Sensor Networks: A survey. *Ad Hoc Netw.* **2012**, *12*, 2026–2061. [[CrossRef](#)]
21. Guvenc, I.; Chong, C.-C. A survey on TOA based wireless localization and NLOS mitigation techniques. *IEEE Commun. Surv. Tutor.* **2009**, *11*, 107–124. [[CrossRef](#)]
22. Shen, J.; Molisch, A.F.; Salmi, J. Accurate passive location estimation using TOA measurements. *IEEE Trans. Wirel. Commun.* **2011**, *11*, 2182–2192. [[CrossRef](#)]
23. Zhang, B.; Zheng, L.; Wu, J.; Zhuang, Z. Joint Synchronization and Localization for Underwater Sensor Networks Considering Stratification Effect. *IEEE Access* **2017**, *5*, 26932–26943. [[CrossRef](#)]
24. Wang, G.; Chen, H.; Li, Y.; Ansari, N. NLOS error mitigation for TOA-based localization via convex relaxation. *IEEE Trans. Wirel. Commun.* **2014**, *13*, 4119–4131. [[CrossRef](#)]
25. Yang, M.; Jackson, D.R.; Chen, J.; Xiong, Z.; Williams, J.T. A TDOA Localization Method for Non-Line-of-Sight Scenarios. *IEEE Trans. Antenn. Propag.* **2019**. [[CrossRef](#)]
26. Le, T.-K.; Ono, N. Closed-form and near closed-form solutions for TDOA-based joint source and sensor localization. *IEEE Trans. Signal Process.* **2017**, *65*, 1207–1221. [[CrossRef](#)]
27. Wang, G.; So, A.M.-C.; Li, Y. Robust convex approximation methods for TDOA-based localization under NLOS conditions. *IEEE Trans. Signal Process.* **2016**, *64*, 3281–3296. [[CrossRef](#)]
28. Lu, L.; Wu, H.-C. Novel robust Direction-of-Arrival-Based source localization algorithm for wideband signals. *IEEE Trans. Wirel. Commun.* **2012**, *11*, 3850–3859. [[CrossRef](#)]
29. Zhang, W.; Yin, Q.; Chen, H.; Gao, F.; Ansari, N. Distributed angle estimation for localization in wireless sensor networks. *IEEE Trans. Wirel. Commun.* **2013**, *12*, 527–537. [[CrossRef](#)]
30. Wang, Y.; Ho, D. Unified Near-Field and Far-Field Localization for AOA and Hybrid AOA-TDOA Positionings. *IEEE Trans. Wireless Commun.* **2017**, *17*, 1242–1254. [[CrossRef](#)]
31. Nguyen, T.L.N.; Shin, Y. An Efficient RSS Localization for Underwater Wireless Sensor Networks. *Sensors* **2019**. [[CrossRef](#)]
32. Hu, Y.; Leus, G. Robust differential received signal strength-based localization. *IEEE Trans. Signal Process.* **2017**, *65*, 3261–3276. [[CrossRef](#)]
33. Xu, Z.; Wang, R.; Yue, X.; Liu, T.; Chen, C.; Fang, S.H. FaceME: Face-to-Machine Proximity Estimation Based on RSSI Difference for Mobile Industrial Human–Machine Interaction. *IEEE Trans. Ind. Informat.* **2018**, *14*, 3547–3558. [[CrossRef](#)]
34. Tan, H.P.; Gabor, A.F.; Eu, Z.A.; Seah, W.K. A wide coverage positioning system (WPS) for underwater localization. In Proceedings of the IEEE International Conference on Communications, Cape Town, South Africa, 23–27 May 2010; pp. 1–5.
35. Othman, A. GPS-less localization protocol for underwater acoustic networks. In Proceedings of the 15th IFIP International Conference on Wireless and Optical Communications Networks, Surabaya, Indonesia, 5–7 May 2008; pp. 1–6.
36. Mirza, D.; Schurgers, C. Motion-aware self-localization for underwater networks. In Proceedings of the WuWNeT '08, San Francisco, CA, USA, 15 September 2008; pp. 51–58.
37. Teymorian, A.Y.; Cheng, W.; Ma, L.; Cheng, X.; Lu, X.; Lu, Z. 3D underwater sensor network localization. *IEEE Trans. Mobile Comput.* **2009**, *8*, 1610–1621. [[CrossRef](#)]
38. Isik, M.T.; Akan, O.B. A three dimensional localization algorithm for underwater acoustic sensor networks. *IEEE Trans. Wirel. Commun.* **2009**, *8*, 4457–4463. [[CrossRef](#)]
39. Erol, M.; Vieira, L.F.M.; Gerla, M. Localization with Dive'N'Rise (DNR) Beacons for Underwater Acoustic Sensor Networks. In Proceedings of the Workshop on Underwater Networks, Montreal, QC, Canada, 14 September 2007; pp. 97–100.
40. Zhang, B.; Wang, H.; Xu, T.; Zheng, L.; Yang, Q. Received signal strength-based underwater acoustic localization considering stratification effect. In Proceedings of the OCEANS 2016, Shanghai, China, 10–13 April 2016.

41. Ban, B.; Yang, J.; Chen, P.; Xiong, J.; Wang, Q. Ship Track Regression Based on Support Vector Machine. *IEEE Access* **2017**, *5*, 18836–18846. [[CrossRef](#)]
42. Lantz, B. Black Box Methods—Neural Networks and Support Vector Machines. In *Machine Learning with R*; Jones, J., Sheikh, A., Eds.; Packt Publishing: Birmingham, UK, 2013; pp. 205–242.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).