Bidirectional Prediction-Based Underwater Data Collection Protocol for End-Edge-Cloud Orchestrated System

Tian Wang, Dan Zhao, Shaobin Cai, Weijia Jia, and Anfeng Liu

***Abstract*— The proliferation of advanced underwater technology and the emergence of various cloud services promote the horizon of cloud-based underwater acoustic sensor network (UASN). Sending end data to cloud for analysis is becoming a prominent trend, driving cloud computing as an indispensable computing paradigm. However, UASN bears tremendous burdens with respect to data collection from end to cloud, such as large transmission power consumption and high delay, which makes it difficult to meet the delay-sensitive and context-aware service requirements by using cloud computing alone. To this end, a two-level bidirectional data prediction model for end-edge-cloud orchestration is proposed in this article. The mobility and computing ability of edge elements are exploited to analyze and collect data. Edge elements predict the future data based on historical information and trend to decrease acoustic communication. Moreover, a data collection protocol with mobile edge elements is designed. With this protocol, computing paradigms are shifted from centralized cloud to distributed edge, and the differentiated capability of heterogeneous devices is exploited. After extensive experiments, the results show that the data collection cost is dramatically decreased while the bandwidth utilization is increased, which is critical for underwater acoustic communication. The proposed method and protocol strike a good balance between data accuracy and energy consumption for the new end-edge-cloud orchestrated system.**

***Index Terms*—AUV (autonomous underwater vehicle) node, data collection, edge computing, prediction, underwater acoustic sensor network (UASN).**

I. Introduction

UNDERWATER acoustic sensor networks (UASNs) ushered in an era where collected data and sensor devices are increasing dramatically following the applications of environmental detection, tactical surveillance, marine energy exploration, and disaster monitoring [1]. It is widely believed that the innovative research of UASN boosts the sustainable development of ocean economy [2]. As a result, cloud computing is introduced as a dominant paradigm to analyze huge underwater data. However, UASN has significant burdens on the increasing energy-limited elements and data traffic, the high cost of battery replacement or charge in terms of end sensor nodes, and the inefficient underwater acoustic communication mode with low transmission rate (generally $\leq$ 100 kbps) which may cause multipath effect, frequency selective fading, and data delay, thus leading to the failure of timely response when dangerous situations occur. Therefore, existing solutions to terrestrial wireless networks are unfeasible for UASN, thus a timely and reliable solution to underwater data collection is of particular importance to address substantial challenges ahead [3], [4].

Moreover, recent efforts to overcome limitations of the underwater environment have focused on endowing them with orchestrated systems for computing ambient [5]. To some extent, forwarding strategy of these methods can be utilized to reduce bandwidth pressure and device latency, while the core of the problem remains unresolved [6]. Existing data collection methods such as multihop, possess mature technical support. However, they failed to address the cost issue of energy consumption. Tilak *et al.* [7] shows that the energy consumed by the sensor network in executing 3000 instructions is equivalent to 100 meters of 1-bit data. Facts indicate that main energy consumption originates in mass data and the long transmission distance especially underwater environment. On this account, prediction-based data collection is gradually coming into sight which can reduce energy consumption for acoustic communication and relieve bandwidth pressure.

Manuscript received May 12, 2019; revised June 24, 2019 and August 27, 2019; accepted September 4, 2019. Date of publication September 11, 2019; date of current version March 17, 2020. This work was supported in part by the Open Fund of Key Laboratory of Data mining and Intelligent Recommendation, Fujian Province University under Grant DM201902, in part by the General Projects of Social Sciences in Fujian Province under Grant FJ2018B038, in part by the National Natural Science Foundation of China (NSFC) under Grant 61872154, Grant 61772148, and Grant 61672441, in part by the Natural Science Foundation of Fujian Province of China under Grant 2018J01092, in part by the Chinese National Research Fund (NSFC) under Project 61532013, Project 61872239, Project 0007/2018/A1, and Project 0060/2019/A1, in part by Science and Technology Development Fund under DCT-MoST Joint-Project 025/2015/AMJ, in part by Macao S.A.R (FDCT), China, in part by the University of Macau under Grant MYRG2018-00237-RTO, Grant CPG2019-00004-FST, and Grant SRG2018-00111-FST, and in part by the Postgraduates' Innovative Fund in Scientific Research of Huaqiao University under Grant 17014083028. Paper no. TII-19-1839. *(Corresponding author: Weijia Jia.)*

T. Wang, D. Zhao, and S. Cai are with the College of Computer Science and Technology, Huaqiao University, Xiamen 361021, China (e-mail: cs_tianwang@163.com; zhaodan_hqu@163.com; caishaobin@hqu.edu.cn).

W. Jia is with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Macau 519000, China (e-mail: weijiaj@gmail.com).

A. Liu is with the School of Computer Science and Engineering, Central South University, Changsha 410006, China (e-mail: afengliu@mail.csu.edu.cn).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2019.2940745

TABLE I
COMPARISON OF DIFFERENT METHODS

Method	Algorithm	Energy Consumption	Latency	Bandwidth	Available Technology	Historical Data
Multi-hop data collection	VBF [15]DFR [17]	High	High	High	Old	×
AUV-aided data collection	AEDG [16]SEDG [18]	High	Middle	Middle	Widespread	×
Detection-based data collection	SDA [19]	Middle	High	Middle	Developing	×
Prediction-based data collection	DBP [20]	Middle	Middle	Low	Advanced	√
Proposed data collection	TBDP	Low	Middle	Low	Advanced	√

In addition, the centralized cloud computing structure is becoming unbearable to meet delay-sensitive and context-aware requirements of applications [8]. As edge computing shifts computing tasks from the centralized cloud to the edge [9], transferred data are enormously reduced by the preprocessing procedures and edge computing is closer to the seabed which can respond to demand more quickly [10]. Therefore, edge computing model is quite suitable to be introduced for the underwater environment since it is likely to afford certain subtasks to lessen data traffic and delay, where AUV (autonomous underwater vehicle) equipped with computing chips serves as edge elements to provide computing service [11], [12]. One problem is how to reliably combine edge computing with complex underwater environment that confuses conventional methods [13].

In this article, a two-level bidirectional data prediction model (TBDP) for end-edge-cloud orchestration system is proposed. The bidirectional prediction method ensures that one party who possesses sensor readings verifies the accuracy of the predicted value so as to determine whether the collected data need to be sent or not. Once error occurs, the predicted value can be corrected timely by taking advantage of the mobility of AUV [14]. TBDP method can eliminate the traditional multihop transmission procedure between nodes, therefore, the overhead decreases with the end-edge-cloud orchestration system [15]. The main contributions are listed as follows.

1) Edge computing and prediction scheme are innovatively combined with underwater data collection to meet the delay-sensitive and context-aware underwater requirements, and a novel underwater structure for end-edge-cloud orchestration system is built.
2) A bidirectional prediction method is designed to save energy, decrease latency, and improve data accuracy. Acoustic communication can be immensely lessened by edge elements predicting the imminent data based on historical information and trend.
3) Considering the computing ability and differentiated capability of diverse heterogeneous nodes, a data collection protocol with different prediction models is creatively proposed in order to face divisional layers.
4) A large number of simulation experiments show that data collection cost is dramatically decreased and a good balance between data accuracy and energy consumption for the new end-edge-cloud orchestrated system is struck.

The rest of this article is organized as follows. Section II reviews related work. Section III introduces end-edge-cloud data collection orchestrated model design. The simulation results are revealed in Section IV. Section V concludes this article.

II. RELATED WORK

Traditional data collection methods in UASN are multihop based [16] or AUV based [17]. However, many different data collection methods are proposed, such as monitoring data collection and predictive data collection. Table I shows characteristic comparisons of several existing methods and newly proposed method. Next, prior work will be reviewed.

The AUV collaborative data collection method has been adopted by many researchers. Ilyas *et al.* [17] designed an AUV-assisted efficient data collection routing protocol (AEDG). The member nodes associate the gateway node with the shortest path and send the data packets to AUV through the gateway node. Although energy consumption of nodes is effectively reduced in this method, it is easy to cause the hot area problem. Naveed *et al.* [19] improved AEDG and proposed a scalable and efficient data acquisition routing protocol (SEDG). In SEDG, AUV dynamically allocates the stay time of gateway nodes (GNs) according to the number of packets received and associated member nodes. Compared with AEDG, SEDG has higher energy efficiency and scalability. Cheng *et al.* [22] proposed a hybrid data collection method that emphasizes the importance of data. The frequently-used data is collected by underwater vehicle AUV from end to top in a spiral trajectory. This scheme can balance the energy consumption of the whole network node, extend the network lifetime, and ultimately reduce the delay of important data [23]. However, the small proportion of important data has little effect on reducing the energy consumption and delay of the whole network.

Zhou *et al.* [20] applied a sensory data collection technology (SDA) for event coverage detection. It is observed that an event may correspond to the deviation between the perceptual data of multiple adjacent sensor nodes and the normal perceptual range. A single sensor node is likely to reflect errors or anomalies on the device or system. Considering this point, the sensor nodes communicate with their neighbors about their perceived data values to see whether the prespecified threshold is exceeded or not. Only when adjacent sensor nodes collectively detect the occurrence of possible events, will these sensor nodes' sensing data be collected and routed to the sink node. This strategy avoids unnecessary routing of perceived data which may be unfavorable to event detection, reduces energy consumption, and increases network capacity. However, this prediction can save energy for the network only when it is used to judge emergencies, and cannot be used for data collection at each time point.

Fang *et al.* [21] proposed a derivative-based prediction (DBP) technology, whose distinguishing feature is mainly reflected

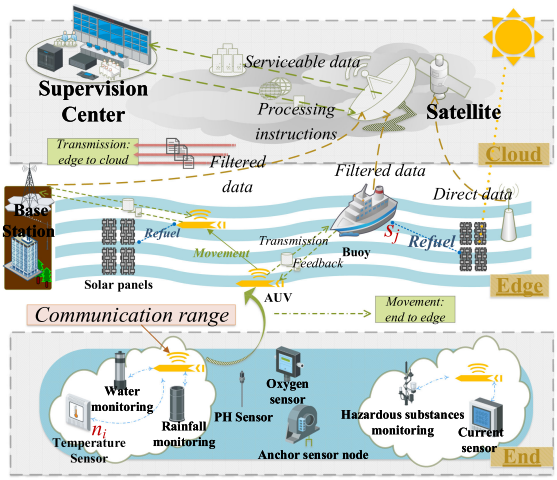


Fig. 1. End-edge-cloud system structure.

in the way of getting new data. The core idea of DBP is to capture the trend of data change with a simple model, and to calculate and deal with the existing interference with flexible rules simultaneously. Users can change the parameter control model if they take samples from historical data collected by sensor nodes, such as using short-term or long-term historical data for training. Although a similar linear model has already existed, the difference of DBP prediction model is that the previous model focuses more on reducing the current error or variance, but DBP on capturing the data change trend under the premise of allowing certain errors. However, one of the limitations of DBP is that the prediction accuracy is low. This algorithm is applicable to data that shows a linear trend in the short term.

Focusing on the underwater application scenarios and the characteristics of AUVs, we analyze the appropriate underwater data collection protocol in order to ensure the accuracy of collected values. A novel and effective model is proposed on the basis of the existing work.

III. End-Edge-Cloud Data Collection Orchestrated Model Design

Underwater data collection is divided into three layers according to the attributes and positions of nodes, namely end layer, edge layer, and cloud layer which form the structure of end-edge-cloud, as shown in Fig. 1. The edge layer is composed of sink nodes, buoys, and mobile nodes. AUVs are equipped with radio and acoustic transceiver, while the end layer contains sensor nodes in the deep ocean [24]. All kinds of nodes placed in different underwater depths involved in this article coordinate to detect ocean conditions. Nodes near the surface can collect energy through solar panels mounted on floating devices, such as mooring buoys [25]. In this section, the orchestrated model proposed will be introduced in detail.

A. Problem Description

In a sea area, n fixed sensor nodes are deployed in three dimensions of underwater space [18]. Let's assume that the data generated by a single node in a unit time is d bit/s; AUVs and surface buoys are deployed as the sink node S. The speed of edge device AUV is v m/s, which has access to common sensor nodes to collect data and then move to surface sink node to deliver data [26]. The energy consumption of calculation operation is low-energy, far less than that of underwater acoustic transmission, therefore, the calculation energy of AUV occupies a small percentage. S needs to transmit data of all underwater sensor nodes to the cloud every t time. C_ψ represents the energy consumption of receiving data, while C_ξ the energy consumption of sending data [27]. Therefore, the energy consumption model needs to meet the following requirement:

$$E_{n_i} = R - C_\xi(\text{data}, L_{n_i \to S_j}) - \sum_{x \epsilon S,n} C_\psi(\text{data}) \quad (1)$$

where R is whole energy of node n_i, data_{n_i} represents the volume that n_i needs to forward, and $L_{n_i \to S}$ is transmission distance. To maximize the network residual energy E_{total}, the energy consumption model needs to meet

$$E_{\text{total}} = \arg\max \sum_{n_i \epsilon n} (E_{n_i}). \quad (2)$$

The underwater data delay includes three parts: $T_{n_i-\text{send}}$, the end node transmits to the mobile node; T_{AUV}, the mobile node transmits to the edge node; and $T_{\text{edge-cloud}}$, the edge node transmits to the cloud. The delay generated from AUV includes mobility delay $T_{\text{AUV}-m}$ and transmission delay $T_{\text{AUV}-t}$. The total delay is shown as follows:

$$T_{\text{total}} = T_{n_i-\text{send}} + T_{\text{AUV}-m} + T_{\text{AUV}-t} + T_{\text{edge-cloud}}. \quad (3)$$

B. Data Collection Protocol for Orchestrated System

Considering the high power consumption and low data rate of acoustic communication, a two-level bidirectional data prediction (TBDP)-based collection protocol is presented, which is mainly based on prediction and supplemented by acoustic communication. The data collection process in this article is shown in Algorithm 1 [28]. Specific process is shown in Fig. 2. Algorithm 1 (line:1–11) indicating the first-level prediction process mainly considers the characteristics of the end nodes (weak computing capacity, difficulty in power storage or node replacement etc.). When the AUV/n_i receives/sends a transmission request, it will conduct an assessment of its own remaining energy, judging whether residual energy is enough or not. When the energy value of node (n_i) falls below the preset energy threshold (σ), it will close itself. The node closes all the circuits for lacking enough energy to support its operation (such as sensing, computing, communication, etc.). When the energy value of AUV (E_{AUV}) falls below the preset energy threshold (σ), it will return to the water surface for charging. When the end node n_i transmits a packet P to the sink node S_j in unit time T, it will broadcast REQUEST control packets to the AUV [29]. Afterwards, AUV and n_i use $AlgorithmFirst$ to collect data predicted value P'. In addition, $compare$ is leveraged to check whether the predited value is credible or not.

Algorithm 1 (line:12–22) indicating the second-level prediction process eliminates the error caused by the first-level

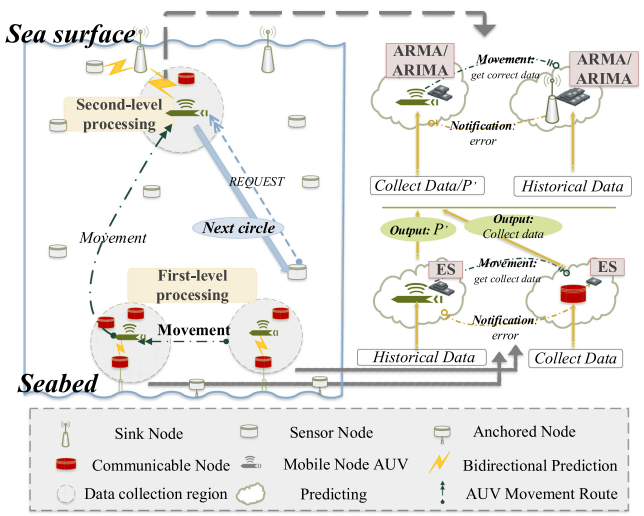


Fig. 2. Two-level bidirectional data prediction structure.

Algorithm 1: Underwater Data Collection Protocol.

Input: A set of charged AUVs; The position of sensor nodes; Error threshold, thd; Prescribed period, T;
Output: Generated data per unit time, $Collect/P'$;

```
if E_{n_i} > σ then
    //generate the first-level prediction;
    P' = AlgorithmFirst
    β = compare(P,' thd)//β = {0, 1};
    if β = 1 then
        send sure to AUV within T time units;
    else
        AUV receives data(d bits) from n_i;
close the node n_i
if t > T then //t is the time since n_i was last collected;
    AUV moves to n_i to get the update data Collect;
if E_AUV > σ then
    //generate the second-level prediction;
    P'' = AlgorithmSecond
    λ = compare(P,'' thd)//λ = {0, 1};
    if λ = 1 then
        discard p;
    else
        if λ = 0 then
            AUV sends data to S_j within Tth time uint;
else
    AUV moves to solar panels to charge;
```

prediction. If the error exceeds the threshold, then AUV quickly receives the correct data at a speed of v bit/s to reduce time delay. After AUV obtains the end node data ($Collect/P'$), $AlgorithmSecond$ is used to obtain the predicted value P'' [30]. Because AUV and sink nodes can be charged or replaced more easily than the end nodes, it is possible for an algorithm model with higher accuracy to achieve the zero error. If the final value P'' is within the error range, there is no need to transmit data.

C. First-Level Bidirectional Prediction Model

The first-level bidirectional data prediction structure is mainly shown in the lower half of Fig. 2. The first-level prediction focuses on the interaction of AUV and end nodes. In this level, the exponential smoothing method is adopted as $AlgorithmFirst$ aiming at the data collection between the end node and AUV. The prediction based on the exponential smoothing method is a weighted average of past observations, and the weights drop exponentially over time. Through the historical statistical sequence of the predicted target, the layer-by-layer smoothing calculation is performed to eliminate the influence caused by random factors [31], [32].

Let the time series observation values be $v_1, v_2, \ldots, v_t$.

1) When there is no obvious trend change in the data of node n_i, the observation of the $t+1$th period can be predicted by using the basic exponential smoothing method in the tth cycle, then the first exponential smoothing formula and the model can be established.
2) When the change of time series demonstrates a linear trend, the first choice uses the law of hysteresis deviation to find out the development direction and trend of the curve, and then a linear trend prediction model is established.
3) If the change of time series shows a curve trend or seasonal trends, cubic exponential smoothing method can be really useful. Considering the specific underwater situation, we adopt single factor value notation-three exponential smoothing [33]. The calculation formula and model are as follows:

$$\begin{cases} V_t^{(1)} = \alpha v_t + (1-\alpha)V_{t-1}^{(1)} \\ V_t^{(2)} = \alpha V_t^{(1)} + (1-\alpha)V_{t-1}^{(2)} \\ V_t^{(3)} = \alpha V_t^{(2)} + (1-\alpha)V_{t-1}^{(3)} \end{cases}. \tag{4}$$

$V_t^{(1)}, V_t^{(2)}$, and $V_t^{(3)}$ represent the exponential smoothing values of the tth period, T is the last predicted cycle, and α is called smoothing coefficient.

$$\begin{cases} \hat{v}_{t+1} = \alpha v_t + (1-\alpha)\hat{v}_t \\ \hat{v}_{t+T} = \varpi(\alpha) + \chi(\alpha)T + \zeta(\alpha)T^2 \end{cases} \tag{5}$$

where, $\hat{v}_{t+T}$ is the predicted value of v_{t+T}th period; the parameters $\varpi(\alpha), \chi(\alpha), \zeta(\alpha)$ are coefficients with α being the only variable.

To obtain the predicted value, α is vital to the model, reflecting the changing trend of data and determining the accuracy of prediction.

Since the algorithm is implemented underwater, the process should be as simple and easy as possible. Thus, the trial-and-error method is adopted for α selection. The larger the coefficient is, the greater the influence of the actual collection value on the predicted value of the cycle is. When the time series shows a stationary trend, the value α is small; on the contrary, when the time series has a large fluctuation, the value α is large. When α changes, root mean squared error (RMSE) and R^2 are used to

judge the accuracy of prediction. Ideally, RMSE is minimized and R^2 is infinitely close to 1. The calculation formulas are as follows.

Goodness of Fit-R-squared

$$R^2 = \max \sum_{k}^{N} (\hat{v}_i - \bar{v})^2 / \sum_{k}^{N} (v_i - \bar{v})^2. \tag{6}$$

Root mean squared error

$$\text{RMSE} = \min \sqrt{\sum\nolimits_{k}^{N} (v_i - \bar{v})^2 / n}. \tag{7}$$

As this level ends, the data has been collected by the edge device AUV and transmitted from the end layer. Next, the data is gathered to the sink node of edge layer, and then uniformly transmitted to the cloud.

D. Second-Level Bidirectional Prediction Model

The second-level bidirectional data prediction structure is mainly shown in the upper half of Fig. 2. Its prediction focuses on the communication of AUV and sink nodes. In order to improve the accuracy and minimize the first-level prediction bias, second-level prediction of edge layer is introduced, and the most commonly-used ARMA model is used for fitting sequences, which we have chosen as $AlgorithmSecond$. It can be subdivided into three categories: AutoRegressive model (AR), Moving-average model (MA), and AutoRegressive-Moving-Average mode (ARMA) model [34]. If the sequence is non-stationary, the differential model, namely, the AutoRegressive Integrated Moving Average (ARIMA) model, can be used.

The ARMA(p, q) model is expressed by mathematical formula as follows:

$$\hat{X}_t = c + \varepsilon_t + \sum_{i=1}^{p} \varphi_i X_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j}. \tag{8}$$

When $q = 0$, the model degenerates into AR(P) model; when $p = 0$, the model degenerates into MA(q) model. The predicted value $\hat{X}_t$ in the above formula is expressed by a linear combination of p previous observations $\{X_{t-1}, X_{t-2}, \ldots, X_{t-p}\}$ and q error values $\{\varepsilon_{t-1}, \varepsilon_{t-2}, \ldots, \varepsilon_{t-p}\}$. Therefore, this problem requires that the optimal set of parameters $(\{\varphi_1, \varphi_2, \ldots, \varphi_p\}, \{\theta_1, \theta_2, \ldots, \theta_q\})$ be obtained.

The selection of the parameter set is very important for the prediction result. To this end, an appropriate method called *extended Kalman filter* (*EKF*) is used for parameter estimation which can estimate the parameters of permission prediction model more effectively. The advantages are:

1) improving the accuracy of the ARMA prediction model, reducing the impact of measurement errors;
2) performing better when there is a nonlinear time series.

More details can be found in reference [35]. The basic *EKF* state space model is as follows:

$$\begin{cases} H_i = A_i H_{i-1} + V_i \\ Y_i = C_i H_i + V_i \end{cases}. \tag{9}$$

In the above formula, H is the n-dimensional system state matrix at time i, A is a state transition matrix that changes with time; V is the system error matrix at time i, W is the measurement error matrix at time i; state noise V_i and observation noise C_i are both white noises that are irrelevant and independent of each other. The combined covariance matrix (noise level) of the two is defined as follows:

$$\begin{cases} E[V_i V_i^T] = Q_i \\ E[W_i W_i^T] = R_i \end{cases} \tag{10}$$

where Q and R are the covariances of the state noise and the observed noise. For the above system, the *EKF* algorithm is displayed as follows. K represents the value obtained by Kalman and P represents an update value of the error covariance

$$\begin{cases} K_i = [A_i P_i {C_i}^T][C_i P_i {C_i}^T + R_i]^{-1} \\ H_{i+1} = A_i H_i + K_i[Y_i - C_i H_i] \\ P_{i+1} = [A_i - K_i C_i]^T P_i [A_i - K_i C_i] + K_i R_i K_i^T \end{cases}. \tag{11}$$

The coefficients in ARMA model are obtained from the three stages. After parameter estimation, the predicted values are solved. Once underwater collection is completed, the coefficients are expressed as C matrix

$$C = [\varphi_1 \varphi_2 \ldots \varphi_p \theta_1 \theta_2 \ldots \theta_q]^T. \tag{12}$$

If there is no random input and noise when measuring the value, H matrix can be constructed

$$H = [y_i \ldots y_{i-p} \varepsilon_i \ldots \varepsilon_{i-q}]. \tag{13}$$

When matrix C and H are determined, the predicted value can be expressed as $Y_i = C_i H_i$.

If the prediction result is within the error range, the data collection (AUV→sink node) will be successful in the $i + 1$th cycle and a complete underwater data collection will be completed (end node→AUV→sink node). When the prediction error occurs, acoustic communication is used for data collection [36].

IV. Performance and Evaluation

In this part, we simulate the data prediction process, and analyze the experimental results.

A. Performance of TBDP Models

Underwater sensors can collect a wide variety of data, such as temperature, salinity, pressue, dissolved oxygen, NH4-N, and other information. According to National Oceanic and Atmospheric Administration (NOAA) database and its KEO site data in recent years, temperature and salinity data are observed as examples. The analysis utilizing the five-year data of NOAA at a depth of 400 m, the time-series plot is shown as Fig. 3.

In terms of the first-level prediction of temperature data, three exponential smoothing methods are used by AUVs and nodes to predict the temperature data before the end of the Tth cycle. Segmental temperature data for experiment from NOAA was selected, and augmented Dickey–Fuller (ADF) test was used to verify the stationarity of time series. If ADF result is 0, the data sequence is unsteady, and the sequence is stable if result is 1. For temperature data, the results are shown in Table II.

The results show that the original data sequence is nonstationary, and first- and second-order differences are jittery in

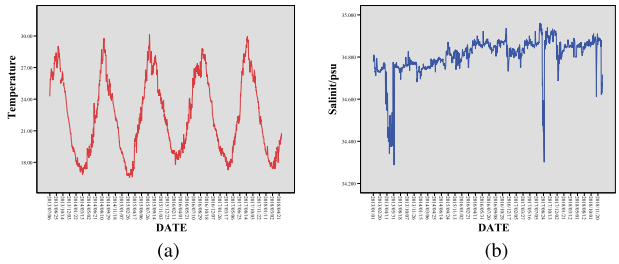


Fig. 3. Time-series plot of underwater data elements. (a) Temperature. (b) Salinity.

TABLE II
ADF RESULT

Data	ADF Result
Original Data	0
First-order difference	1
Second-order difference	1

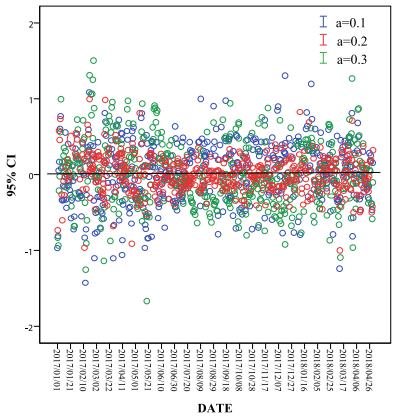


Fig. 4. Error distribution.

some interval segments, but the degree of jitter is not intense. Therefore, a smaller value ($\alpha = 0.1/0.2/0.3$) of smoothing coefficient α should be selected for calculation by trial-and-error method [33]. Through analyzing the ocean data of 2013–2018, different error distribution diagrams are shown in Fig. 4. As can be seen from the figure, the error distribution of blue circle ($\alpha = 0.1$) and green circle ($\alpha = 0.3$) is more dispersive, while the red circle error point ($\alpha = 0.2$) is more densely distributed above and below the reference line ($X = 0$).

When $\alpha = 0.2$, RMSE value is higher and R-square is lower as shown in the Table III. According to the formula (6) and (7), it can be judged that the prediction accuracy is the highest and this value is the most reasonable.

In this article, the historical data of the first three quarters of 2015–2017 is used as the training set. The verification set consists of the real data collected in the fourth quarter. For the data of 2018, only four months are used for training. After the actual value is fitted with the predicted value, the obtained results are shown in Fig. 5.

1) Fig. 5(a)–(c) show that the selected parameters, trend, and periodic analysis are basically correct, which have good prediction effect and high accuracy. The predicted data floats above and below the true value within the error threshold.
2) In 2018, there are fewer training datasets. The experimental results show that the data fluctuate partially, but the overall trend prediction is accurate. However, there are outliers, in the partially enlarged detail in Fig. 5(d). Then the node sends a feedback instruction. As soon as AUV receives the instruction, it will move to the node and correct the error values.

The stability of the sequence should be judged by the model based on autocorrelation function and the partial correlation function. When both are trailing, the data is nonstationary and needs to be differentiated. The differential processing is done by the extended model ARIMA (p, d, q), and the obtained differential sequence is a stationary one. The number of differences is the d value.

The parameters of ARIMA can be estimated by *EKF*, and appropriate ones will be obtained, aiming to verify the error between the predicted value and the actual value. Partial data in 2018 is used for experiments to single out different P and Q values. After many verifications, it is found that the observed value (purple circle) fits best with ARIMA(*1,1,1*) (blue circle) results when $p = 1$ and $q = 1$, as shown in Fig. 6. When p and q are slightly changed, a large error may occur. From the above analysis, it is discovered that the model has a better prediction effect on the values of P and Q.

The above descriptions show when the appropriate parameters are obtained, the model results are more reliable and the accuracy is higher. However, when there are uncorrected errors, problems will definitely arise in subsequent work. Therefore, it is necessary for AUV to make predictions within the communication range. Partial data in 2018 verifies the accuracy of data prediction at various levels. Fig. 7 depicts the graphical error between the observed value and the two-level predicted value. Due to the small quantity of data, the first-level prediction has some certain errors.

a) If the error is not corrected, the first-level prediction will not be corrected in time. If the second-level prediction value is not highly fitted to the first-level prediction value, the former cannot transmit accurate data to the aggregation node, and finally will cause errors.
b) If the error is corrected, it can be seen that the three data lines are basically fitted, and the second-level prediction protocol has a higher accuracy. When AUV receives the correct data, the sink node can accurately predict the future value, extending the life cycle of the node.

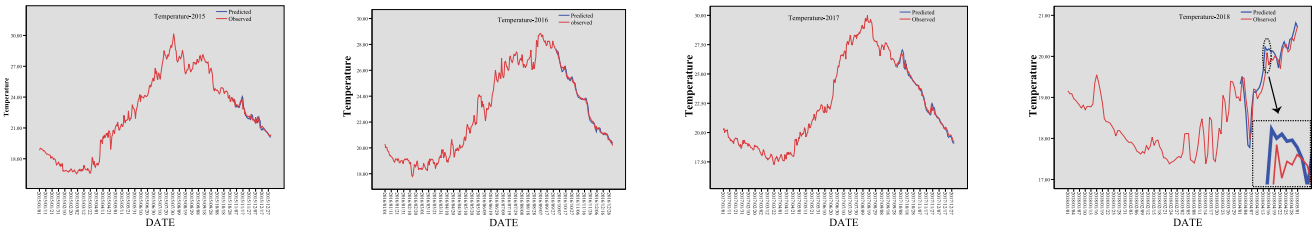
Therefore, it is necessary to introduce the edge layer as a branch, to ensure the accuracy of the data. The end-edge-cloud model can be adjusted to receive diverse data, and parameters of the model are changed to achieve the best prediction effect, so the underwater nodes do not need to receive/send the data any more.

B. Results Analysis

In this article, it is assumed that nodes are randomly distributed in a 500 × 400 × 300 m region. In this simulation, we

TABLE III
RMSE AND R-SQUARE RESULTS

	α=0.1		α=0.2		α=0.3	
Year	R^2	RMSE	R^2	RMSE	R^2	RMSE
2013	0.98766	0.0026	0.9899	0.0013	0.9790	0.0051
2014	0.9877	0.0005	0.9854	0.0007	0.9880	0.0007
2015	0.9891	0.0003	0.9861	0.0026	0.9897	0.0008
2016	0.9883	0.0013	0.9892	0.0013	0.98512	0.003
2017	0.9938	0.0078	0.9913	0.0043	0.9942	0.0077
2018	0.9570	0.025	0.9913	0.0175	0.9942	0.0255



(a) (b) (c) (d)

Fig. 5. Temperature training results. (a) Training results in 2015. (b) Training results in 2016. (c) Training results in 2017. (d) Training results in 2018.

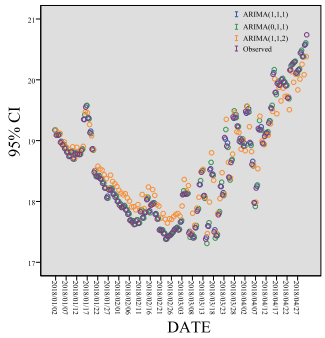
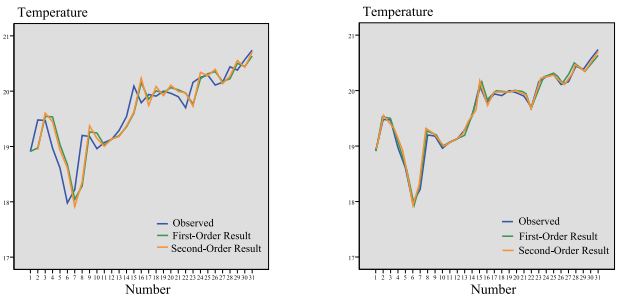


Fig. 6. Fitting analysis results.



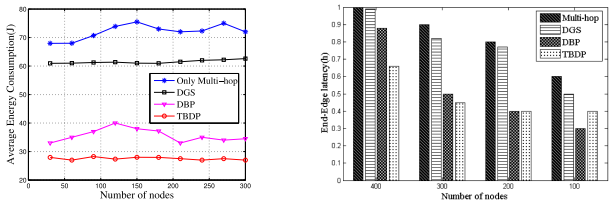
(a) (b)

Fig. 7. Two-level prediction results. (a) Failure to correct errors in time. (b) Timely error correction.

assume that the environment in UWSN is equipped with AUV which has sufficient computing and storage capacity and the sensor node has certain computing capacity as well. The specific parameter settings required for the experiment are shown in Table IV. The results in Fig. 8 show the contrast between unit

TABLE IV
EXPERIMENTAL PARAMETERS

Parameter	value
Network Size($X \times Y \times Z$)	500m×400m×300m
Number of Nodes($\|S\|$)	50-300
Sensed Range	50m-60m
Packet Size(L)	1024bit
Data Packets Generation Rate	100p/r per sensor
Node Initial Energy	100J
Reception Power Consumption	0.8×10^{-3}W
Transmission Power Consumption	1.6×10^{-3}W
Power Consumption(idle)	0.1×10^{-3}W



(a) (b)

Fig. 8. End nodes performance. (a) Average consumption. (b) End-edge latency.

energy consumption (i.e., nodes per rotation contract awarding the average energy consumption) and end-edge latency.

1) Fig. 8(a) shows that with the quantity change of nodes, the average energy consumption of node for a cycle of data collection is different. In the figure, the four protocols fluctuate around a certain value as the number of nodes increase. For instance, TBDP is basically maintained at 27 J. As can be clearly seen from the figure, the blue line segment has the largest energy consumption including sending and forwarding parts that can only

be forwarded by multihop routing protocol. The DBP protocol consumes less energy than the first two models, but once an error occurs, traditional protocol is still needed to transmit the correct data.

2) Fig. 8(b) shows that as the number of nodes increases, the latency of all methods increases on different levels. When the number of nodes is small, it can be seen that the delay of DBP is the smallest, but as the number of nodes increases, the error rate of DBP increases, and the delay increases. DGS and multihop latency is similar to the linear increase. In contrast, TBDP performance is smooth and better.

Generally speaking, the TBDP model plays a vital role in saving node energy compared with existing protocols.

V. Conclusion

With the surge of underwater data and the emergence of various applications, cloud computing analysis, the dominant paradigm, had become inefficient. Meanwhile, UASN had its own flaws on data traffic, energy-limited nodes, and acoustic communication consumption. The cost of communication was high while the rate was pretty low. Therefore, edge computing played an important role in lessening the burden of cloud computing while some computing tasks could be offloaded from the centralized cloud to the edge. Moreover, AUVs were regarded as edge elements to predict future data based on historical information to decrease acoustic communication and the huge energy consumption. Therefore, an end-edge-cloud orchestrated system was proposed to adapt to delay-sensitive and context-aware underwater environment. In this article, we designed TBDP, a novel data collection scheme with AUVs. The bidirectional prediction method ensured the accuracy of the predicted value by means of verifying one side of sensor readings. In addition, the design of two-level model was to meet computing capacity of diverse nodes underwater. Experimental results showed that if the adaptive exponential smoothing algorithm was combined with ARMA algorithm and TBDP, the accuracy of data collection would be maintained, the cost of data collection, the average node energy consumption, and latency would be significantly reduced.

References

[1] S. Basagni, V. Di Valerio, P. Gjanci, and C. Petrioli, "Harnessing hydro: Harvesting-aware data routing for underwater wireless sensor networks," in *Proc. 18th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2018, pp. 271–279.

[2] K. Wang, H. Gao, X. Xu, J. Jiang, and D. Yue, "An energy-efficient reliable data transmission scheme for complex environmental monitoring in underwater acoustic sensor networks," *IEEE Sensors J.*, vol. 16, no. 11, pp. 4051–4062, Jun. 2016.

[3] G. Han, S. Shen, H. Song, T. Yang, and W. Zhang, "A stratification-based data collection scheme in underwater acoustic sensor networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10 671–10 682, Nov. 2018.

[4] Y. Wu, H. Huang, N. Wu, Y. Wang, M. Z. A. Bhuiyan, and T. Wang, "An incentive-based protection and recovery strategy for secure big data in social networks," *Inf. Sci.*, vol. 508, pp. 79–91, Jan. 2020.

[5] Y. Guo, B. Zou, J. Ren, Q. Liu, D. Zhang, and Y. Zhang, "Distributed and efficient object detection via interactions among devices, edge and cloud," *IEEE Trans. Multimedia*, to be published, doi: 10.1109/TMM.2019.2912703.

[6] D. Yang *et al.*, "Assignment of segmented slots enabling reliable real-time transmission in industrial wireless sensor networks," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3966–3977, Jun. 2015.

[7] S. Tilak, N. B. Abu-Ghazaleh, and W. Heinzelman, "A taxonomy of wireless micro-sensor network models," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 6, no. 2, pp. 28–36, 2002.

[8] T. Wang, L. Qiu, G. Xu, A. K. Sangaiah, and A. Liu, "Energy-efficient and trustworthy data collection protocol based on mobile fog computing in Internet of Things," *IEEE Trans. Ind. Inform.*, to be published, doi: 10.1109/TII.2019.2920277.

[9] T. Wang, H. Luo, J. Zheng, and M. Xie, "Crowdsourcing mechanism for trust evaluation in CPCS based on intelligent mobile edge computing," *ACM Trans. Intell. Syst. Technol.*, 2019, to be published, doi: 10.1145/3324926.

[10] Z. Ning, X. Kong, F. Xia, W. Hou, and X. Wang, "Green and sustainable cloud of things: Enabling collaborative edge computing," *IEEE Commun. Mag.*, vol. 57, no. 1, pp. 72–78, Jan. 2019.

[11] X. Peng, J. Ren, L. She, D. Zhang, J. Li, and Y. Zhang, "BOAT: A block-streaming app execution scheme for lightweight IoT devices," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1816–1829, Jun. 2018.

[12] J. Pan, J. Wang, A. Hester, I. AlQerm, Y. Liu, and Y. Zhao, "EdgeChain: An edge-IoT framework and prototype based on blockchain and smart contracts," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4719–4732, Jun. 2018.

[13] J. Ren, H. Guo, C. Xu, and Y. Zhang, "Serving at the edge: A scalable IoT architecture based on transparent computing," *IEEE Netw.*, vol. 31, no. 5, pp. 96–105, 2017.

[14] I. Jawhar, N. Mohamed, J. Al-Jaroodi, and S. Zhang, "An architecture for using autonomous underwater vehicles in wireless sensor networks for underwater pipeline monitoring," *IEEE Trans. Ind. Inform.*, vol. 15, no. 3, pp. 1329–1340, Mar. 2019.

[15] H. U. Yildiz, V. C. Gungor, and B. Tavli, "Packet size optimization for lifetime maximization in underwater acoustic sensor networks," *IEEE Trans. Ind. Inform.*, vol. 15, no. 2, pp. 719–729, Feb. 2019.

[16] P. Xie, J.-H. Cui, and L. Lao, "VBF: Vector-based forwarding protocol for underwater sensor networks," in *Proc. Int. Conf. Res. Netw.*, 2006, pp. 1216–1221.

[17] N. Ilyas *et al.*, "AEDG: AUV-aided efficient data gathering routing protocol for underwater wireless sensor networks," *Procedia Comput. Sci.*, vol. 52, pp. 568–575, 2015.

[18] D. Hwang and D. Kim, "DFR: Directional flooding-based routing protocol for underwater sensor networks," in *Proc. OCEANS*, Quebec City, QC, Canada, Sep. 2008, pp. 1–7.

[19] N. Ilyas *et al.*, "SEDG: Scalable and efficient data gathering routing protocol for underwater WSNs," *Procedia Comput. Sci.*, vol. 52, pp. 584–591, 2015.

[20] Z. Zhou, R. Xing, Y. Duan, Y. Zhu, and J. Xiang, "Event coverage detection and event source determination in underwater wireless sensor networks," *Sensors*, vol. 15, no. 12, pp. 31 620–31 643, 2015.

[21] Z. Zhou, W. Fang, J. Niu, L. Shu, and M. Mukherjee, "Energy-efficient event determination in underwater WSNs leveraging practical data prediction," *IEEE Trans. Ind. Inform.*, vol. 13, no. 3, pp. 1238–1248, Jun. 2017.

[22] C.-F. Cheng and L.-H. Li, "Data gathering problem with the data importance consideration in underwater wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 78, pp. 300–312, 2017.

[23] G. Han, X. Long, C. Zhu, M. Guizani, and W. Zhang, "A high-availability data collection scheme based on multi-auvs for underwater sensor networks," *IEEE Trans. Mobile Comput.*, to be published, doi: 10.1109/TMC.2019.2907854.

[24] J. Ren, S. Yue, D. Zhang, Y. Zhang, and J. Cao, "Joint channel assignment and stochastic energy management for RF-powered OFDMA WSNS," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1578–1592, Feb. 2019.

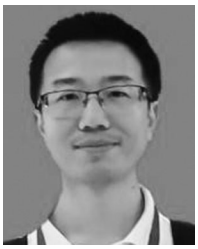
[25] J. Jiang, G. Han, L. Shu, S. Chan, and K. Wang, "A trust model based on cloud theory in underwater acoustic sensor networks," *IEEE Trans. Ind. Inform.*, vol. 13, no. 1, pp. 342–350, Feb. 2017.

[26] T. Wang, H. Ke, X. Zheng, K. Wang, A. K. Sangaiah, and A. Liu, "Big data cleaning based on mobile edge computing in industrial sensor-cloud," *IEEE Trans. Ind. Inform.*, to be published, doi: 10.1109/TII.2019.2938861.

[27] B. Cao, J. Zhao, P. Yang, Z. Lv, X. Liu, and G. Min, "3-D multiobjective deployment of an industrial wireless sensor network for maritime applications utilizing a distributed parallel algorithm," *IEEE Trans. Ind. Inform.*, vol. 14, no. 12, pp. 5487–5495, Dec. 2018.

[28] L. Qi, X. Zhang, W. Dou, C. Hu, C. Yang, and J. Chen, "A two-stage locality-sensitive hashing based approach for privacy-preserving mobile service recommendation in cross-platform edge environment," *Future Gener. Comput. Syst.*, vol. 88, pp. 636–643, 2018.

[29] Y. Chen *et al.*, "Fast neighbor search by using revised KD tree," *Inf. Sci.*, vol. 472, pp. 145–162, 2019.
[30] P. D. Ganjewar, S. Barani, and S. J. Wagh, "HFBLMS: Hierarchical fractional bidirectional least-mean-square prediction method for data reduction in wireless sensor network," *Int. J. Model., Simul., Sci. Comput.*, vol. 9, no. 02, 2018, Art. no. 1850020.
[31] G. B. Tayeh, A. Makhoul, D. Laiymani, and J. Demerjian, "A distributed real-time data prediction and adaptive sensing approach for wireless sensor networks," *Pervasive Mobile Comput.*, vol. 49, pp. 62–75, 2018.
[32] W. Gong, L. Qi, and Y. Xu, "Privacy-aware multidimensional mobile service quality prediction and recommendation in distributed fog environment," *Wireless Commun. Mobile Comput.*, vol. 2018, 2018, Art. no. 3075849.
[33] Y. Su, W. Gao, D. Guan, and W. Su, "Dynamic assessment and forecast of urban water ecological footprint based on exponential smoothing analysis," *J. Cleaner Prod.*, vol. 195, pp. 354–364, 2018.
[34] E. M. De Oliveira and F. L. C. Oliveira, "Forecasting mid-long term electric energy consumption through bagging ARIMA and exponential smoothing methods," *Energy*, vol. 144, pp. 776–788, 2018.
[35] S. M. Hashemi and M. Sanaye-Pasand, "A new predictive approach to wide-area out-of-step protection," *IEEE Trans. Ind. Inform.*, vol. 15, no. 4, pp. 1890–1898, Apr. 2019.
[36] L. Liu, Y. Wang, C. Wang, F. Ding, and T. Hayat, "Maximum likelihood recursive least squares estimation for multivariate equation-error arma systems," *J. Franklin Inst.*, vol. 355, no. 15, pp. 7609–7625, 2018.



Tian Wang received the B.Sc. and M.Sc. degrees in computer science from Central South University, Changsha, China, in 2004 and 2007, respectively. He received the Ph.D. degree in computer science from the City University of Hong Kong, Hong Kong, in 2011.

He is a Professor with the College of Computer Science and Technology, Huaqiao University, Xiamen, China. His research interests include wireless sensor networks, fog computing, and mobile computing.

Dan Zhao received the B.S. degree in Internet of Things engineering from the Henan Normal University of China, Xinxiang, China, in 2016. She is currently working toward the master's degree in computer technology with the National Huaqiao University of China, Quanzhou, China.

Her research interests include underwater sensor networks, mobile computing, and edge computing.

Shaobin Cai received the Ph.D. degree in computer system architecture from the Harbin Institute of Technology, Harbin, China, in 2005. He is a Minjiang Scholar of Fujian Province.

He is currently a Professor of Computer Science and Technology with Huaqiao University, Quanzhou, China. His primary research interests include ad hoc networks, wireless sensor networks, underwater acoustic sensor networks, and the blockchain technology.

Dr. Cai is a member of the Wireless Sensor Network Committee of the Chinese Computer Society.

Weijia Jia received the B.Sc. and M.Sc. degrees from Center South University, China, in 1982 and 1984, and the Master of Applied Science and Ph.D. degrees from the Polytechnic Faculty of Mons, Belgium, in 1992 and 1993, respectively, all in computer science.

He is currently a Professor with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Zhuhai. His research interests include next generation wireless communication, protocols, and heterogeneous networks.

Prof. Jia has served as an Editor or Guest editor for international journals and as Program Committee (PC) Chair or member/Keynote Speaker for various prestigious international conferences. He is a member of the Association for Computing Machinery (ACM) and China Computer Federation (CCF).

Anfeng Liu received the M.Sc. and Ph.D degrees from Central South University, Changsha, China, in 2002 and 2005, respectively, both majored in computer science.

He is currently a Professor with the School of Computer Science and Engineering, Central South University. His major research interests include cyber-physical systems, service network, and wireless sensor network.

Dr. Liu is a member of China Computer Federation (CCF).