Exploiting Deep Reinforcement Learning for Multi-AUV Assisted VoI-Maximum Data Collection in UWSNs

Xuan Gu*, Jiarun Tang*, Xiao Huang†, Jianhua He‡, and Jing Xu*
*School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China
†China Ship Development and Design Center, Wuhan, China
‡School of Computer Science and Electronic Engineering, University of Essex, Colchester, CO4 3SQ, UK
Emails: {guxuan, tangjiarun, xujing}@hust.edu.cn*, huangxiao_88@outlook.com†, j.he@essex.ac.uk‡

***Abstract*—Reliable and timely data collection is an important and challenging problem for underwater wireless sensor networks (UWSNs), partly due to the very slow underwater communication and the difficulty in recharging the sensors. In this paper, we exploit the use of autonomous underwater vehicles (AUVs) for UWSN data collection task. Specifically, we investigate how to maximize the value of information (VoI) for data collection through joint optimization of cluster head (CH) selection and multi-AUV path planning. We formulate the joint optimization problem for the task, taking into account the energy constraints of sensor nodes. To solve the problem, we propose a deep reinforcement learning algorithm based on an encoder-decoder architecture. The entire UWSN system is fed into the encoder network, followed by a composite decoder consisting of an AUV selection decoder and a cluster access sequence decoder to obtain the cluster access sequence for each AUV. Based on the determined sequences, we further utilize the dynamic programming algorithm to achieve optimal CH selection. Finally, we obtain the sequence of AUVs accessing the selected CHs. Simulation results demonstrate that the proposed learning-based approach converges and achieves a higher VoI than the existing benchmark algorithms.**

***Index Terms*—AUV, data collection, VoI, path planning**

I. Introduction

Underwater wireless sensor networks (UWSNs) play a crucial role for ocean monitoring and resource exploitation, with applications such as marine environment monitoring, surveillance of underwater facilities, early disaster warning [1]. One of the major challenges for UWSNs is reliable and efficient data collection. Compared to the terrestrial sensor networks, communication between underwater sensor nodes and data sinks often relies on acoustic signals due to the rapid attenuation of radio frequency signals in underwater environments. However, underwater acoustic communication is known for its low data rate, large delay, and high energy consumption limitations [2]. Additionally, replacing batteries for underwater sensor nodes is extremely difficult. Therefore, there is a strong demand for reliable and efficient data collection methods.

Recent advances in autonomous underwater vehicle (AUV) have led to increasing research interest in utilizing AUVs to assist data collection in UWSNs. A key advantage of using AUVs for data collection is their ability to periodically navigate towards sensor nodes and collect data from them using close-range and high-speed communication links (such as optical or electromagnetic communication) [3]. In this way, the data can be collected fast with minimized energy consumption of the sensor nodes, thereby significantly extending the lifetime of UWSNs. To enhance data collection efficiency, it has been proposed to cluster sensor nodes and select cluster heads (CHs) to collect and aggregate data from other nodes in the clusters [4]. This allows AUVs to collect data from the entire area by visiting a limited number of CHs.

Existing studies have focused on single AUV deployments in small-scale scenarios. However, the increasing scale of UWSNs and the complexity and low-latency requirements of underwater operational tasks necessitate collaborative data collection by multiple AUVs [5]. In this paper, we are motivated to investigate the use of multiple AUVs for reliable and efficient data collection in UWSNs. We specifically consider the path planning of AUVs as it significantly impacts the energy consumption of AUVs and data collection delay [6], [7]. Meanwhile, considering the influence of CH selection, we investigate the problem of jointly optimizing CH selection and multi-AUV path planning.

We first formulate the joint optimization problem for the task, taking into account the energy constraints of sensor nodes. Unlike existing research which primarily focuses on reducing data latency and extending the lifespan of sensor networks, we aim to maximize the value of information (VoI), which emphasizes the timeliness of information and highlights the significant differences in its importance [8]–[10]. Furthermore, the path planning problem can be viewed as a sequence-to-sequence problem, which is particularly suitable for solving using an encoder-decoder architecture. Thus, we propose an encoder-decoder based deep reinforcement learning (EDDRL) algorithm for its solution. The UWSN system serves as input to the encoder network, which is then decoded by a composite decoder comprising an AUV selection decoder and a cluster access sequence decoder. This process allows us to obtain the cluster access sequence for each AUV. Subsequently, CHs are selected by using a dynamic programming method with the predetermined sequences, thereby deriving the selected CHs access sequence for each AUV. The parameters of the proposed algorithm are trained in an unsupervised manner using the REINFORCE algorithm. Simulation results demonstrate that the proposed algorithm converges and achieves a higher VoI compared to other benchmark algorithms.

The remainder of this paper is organized as follows. Section

II introduces the system model and formulates the problem. The proposed algorithm is detailed in Section III. Section IV provides simulation results, and Section V concludes the paper.

II. System Model and Problem Formulation

We consider a scenario of large-scale UWSNs for multi-AUV collaborative data collection. The system comprises a surface station, multiple AUVs, and numerous underwater sensor nodes. The underwater sensor nodes are deployed regionally, and perform continuous monitoring of the specific region for which they are responsible. It is now assumed that the system operates under non-emergency conditions, where the AUVs periodically depart from the surface station, navigate to each CH to collect data according to predetermined paths, and upon completing the data collection, return to the surface station for data offloading.

To improve the efficiency of data collection, all nodes are organized into M clusters, each of which is equipped with a designated CH. For any given cluster $C_m, m \in \{1,2,\ldots,M\}$, the CH is designated as $H_m, H_m \in C_m$, making the set of CHs represented by $H = \{H_1, H_2, \ldots, H_M\}$. Before the AUV arrives, CH H_m collects data from other member nodes through hydroacoustic communication. When the AUV visits H_m, the CH transmits a data packet to the AUV via an optical communication link. As a result, the energy consumption of the member node i and the CH of C_m can be denoted as

$$\begin{aligned} \mathcal{E}_i &= P_t \frac{D_i}{R(d_i)} \\ \mathcal{E}_m &= P_r \sum_{i \in \widetilde{C_m}} \frac{D_i}{R(d_i)} + P_t \frac{I_m}{R_o} \end{aligned} \tag{1}$$

where D_i is the amount of data collected from the underwater environment by the node i in cluster C_m, I_m is the total amount of data in the cluster, denoted by $I_m = \sum_{i \in C_m} D_i$. The receiving power and transmitting power of the sensor node are denoted by P_r and P_t, respectively. The set of member nodes in the cluster C_m is denoted by $\widetilde{C_m}$. The Euclidean distance of node i from the CH H_m is denoted by d_i. $R(d_i)$ represents the hydroacoustic communication rate, while the optical communication rate between the AUV and the CH is denoted by R_o.

We assume that AUV_j needs to access Z_j CHs during a data collection task, and define binary variable $A_j^m[\zeta]$ to indicate whether AUV_j visits the CH H_m for data collection during step ζ. A value of 1 represents visitation, while a value of 0 indicates non-visitation. Therefore, the CHs access sequence for all AUVs can be denoted as $A = \left\{A_j^m[\zeta], 1 \le j \le U, 1 \le m \le M, 1 \le \zeta \le Z_j\right\}$. As mentioned above, the movement of AUVs must satisfy the constraints as follows

$$\sum_{j=1}^{U} \sum_{\zeta=1}^{Z_j} A_j^m[\zeta] = 1, \forall 1 \le m \le M \tag{2a}$$

$$\sum_{j=1}^{U} \sum_{\zeta=1}^{Z_j} \sum_{m=1}^{M} A_j^m[\zeta] = M, \forall 1 \le j \le U \tag{2b}$$

where (2a) restrict each CH to be visited by only one AUV in each round of data collection, and (2b) indicates that all AUVs need to visit all CHs.

The set of trajectory of the AUV_j can be denoted as $\mathcal{L}_j = \{\ell_j[0], \ell_j[1], \ldots, \ell_j[Z_j], \ell_j[0]\}$, where $\ell_j[\zeta]$ represents the coordinates of the AUV at step ζ and $\ell_j[0]$ is the coordinates of the surface station. Let us analyze the change in VoI after AUV collects data from the CH. Assume that the initial VoI of C_m is denoted as E_m, thus the initial VoI of collected data when AUV_j visits the CH at step ζ denoted as $E_j[\zeta] = \sum_{m=1}^{M} A_j^m[\zeta] E_m$, which reflects the importance of the data. Once the data is transmitted to AUV, its VoI decreases with time, and the rate of this decrease is influenced by the subsequent travel time of the AUV and its communication time with the CH. At the moment $T_j^{\zeta+1}$, the VoI remaining of the data collected is represented as

$$V_j^\zeta\left(T_j^{\zeta+1}\right) = \beta E_j[\zeta] + (1-\beta) E_j[\zeta] e^{-\frac{\Delta T_j^{\zeta \to \zeta+1}}{\alpha}} \tag{3}$$

where $\beta \in [0,1]$ is a factor that measures the trade-off between data importance and timeliness, and α is the decay factor, reflecting the rate of decay of the VoI over time. $\Delta T_j^{\zeta \to \zeta+1}$ denotes the difference between the moments when the AUV_j arrives at position $\ell_j[\zeta]$ and position $\ell_j[\zeta+1]$ respectively, which includes the time of the optical communication between the AUV and the CH at position $\ell_j[\zeta]$, as well as the time of the movement of the AUV from position $\ell_j[\zeta]$ to position $\ell_j[\zeta+1]$. Denote the amount of data collected by AUV_j when it visits the CH at step ζ as $I_j[\zeta] = \sum_{m=1}^{M} A_j^m[\zeta] I_m$. Thus $\Delta T_j^{\zeta \to \zeta+1}$ is equal to

$$\Delta T_j^{\zeta \to \zeta+1} = T_j^{\zeta+1} - T_j^{\zeta} = \frac{I_j[\zeta]}{R_o} + \frac{d_j[\zeta]}{v_a} \tag{4}$$

where v_a denotes the movement speed of the AUV, and $d_j[\zeta] = \|\ell_j[\zeta+1] - \ell_j[\zeta]\|$ is the displacement distance of the AUV from position $\ell_j[\zeta]$ to position $\ell_j[\zeta+1]$. When data collection is complete, the total system VoI collected by the surface station can be expressed as

$$V = \sum_{j=1}^{U} \sum_{\zeta=1}^{Z_j} \sum_{i=\zeta}^{Z_j} \left(\beta E_j[\zeta] + (1-\beta) E_j[\zeta] e^{-\frac{\Delta T_j^{i \to i+1}}{\alpha}} \right) \tag{5}$$

Since the retained VoI for data collected by AUVs at each CH depends on the time interval required by their return to the surface station. This time interval is affected by the path planning of AUVs, and the path of each AUV is influenced by the CH selected for each cluster and the CHs access sequence for AUVs. The aim of this paper is to maximize the VoI of the UWSNs by jointly optimizing the selection of CHs H and the CHs access sequence A for all AUVs. The optimization problem is formulated as follows

$$\mathbf{P}_0 : \max_{H,A} V(H,A) \tag{6a}$$

$$\text{s. t. } H_m \in C_m, \forall 1 \le m \le M \tag{6b}$$

$$\mathcal{E}_m \le \mathcal{E}_{\max} \tag{6c}$$

$$\text{(2a) and (2b)} \tag{6d}$$

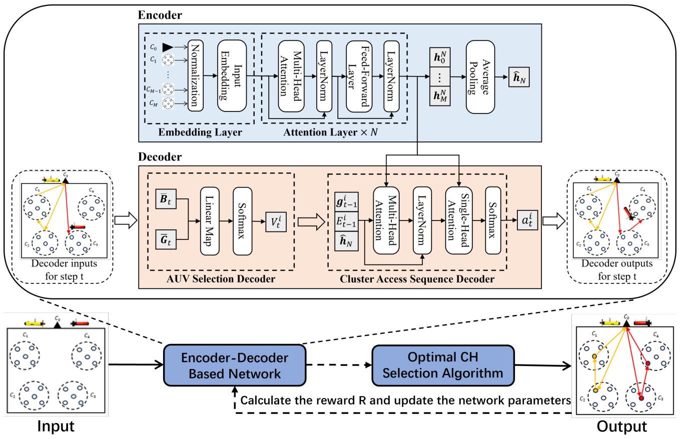


Fig. 1: The proposed algorithm architecture.

where (6b) delineates the restrictions associated with the CHs, (6c) denotes the energy constraints for each sensor node within a single round of data collection, where $\mathcal{E}_{\max}$ represents the maximum allowable energy consumption. Since the aim of $\mathbf{P}_0$ depends on the AUV's travel time and communication duration with the CH, the cluster access sequence of the AUV needs to integrate considerations of CH coordinates, data volume within the cluster, and the initial VoI of the data.

III. Encoder-Decoder Based Deep Reinforcement Learning Algorithm

$\mathbf{P}_0$ is a classic NP-hard problem, which is challenging to obtain the optimal solution within polynomial time due to its inherent complexity. To solve this problem, we propose EDDRL algorithm. Specifically, the algorithm obtains the cluster access sequence of each AUV through a network based on an encoder-decoder architecture. Then, the CHs are selected based on the previously determined sequences through an optimal CH selection algorithm based on dynamic programming. Additionally, the algorithm calculates the system rewards based on acquired VoI and updates the parameters of the network using the REINFORCE algorithm. The overall architecture of the algorithm is depicted in Fig. 1.

A. Encoder-Decoder Based Network

To obtain the cluster access sequence for each AUV, we employ an encoder-decoder based network where the inputs include the coordinates of the surface station, as well as the coordinates, data volume, and data VoI of nodes in multiple clusters, and the output is the cluster access sequence for each AUV. In this paper, the cluster access sequence of AUVs is viewed as a mapping between sequences, where the initial sequence contains multiple clusters including the surface station, and the target sequence is the cluster access sequence for each AUV.

1) Encoder: The encoder is responsible for extracting features from the input data $\boldsymbol{I} = \{\boldsymbol{I}_0, \boldsymbol{I}_1, \ldots, \boldsymbol{I}_M\}$, where $\boldsymbol{I}_0 = \ell_0 \in \mathbb{R}^3$ represents the starting point of AUVs, $\boldsymbol{I}_m$ represents all the node coordinates, total data volume, and initial VoI within cluster C_m. As shown in Fig. 1, the encoder mainly consists of multiple attention layers. First, the input $\boldsymbol{I}$ is embedded as $\boldsymbol{h}_0$. Subsequently, the embedded data $\boldsymbol{h}_0$ is fed into N attention layers for better feature representation, each attention layer is the same as the one in the standard Transformer, consisting of a multi-head attention layer and a feed-forward layer, each of which performs residual concatenation and layer normalization. Assuming that the input of the nth attention layer is $\boldsymbol{h}_{n-1} = \left\{\boldsymbol{h}_0^{n-1}, \boldsymbol{h}_1^{n-1}, \ldots, \boldsymbol{h}_M^{n-1}\right\}$, where $\boldsymbol{h}_m^{n-1}$ represents the feature representation of cluster C_m after going through the previous $n-1$ attention layers. Then the nth attention layer can be represented as

$$\begin{aligned} \boldsymbol{Y}_n &= \mathrm{MHA}\left(\boldsymbol{h}_{n-1}, \boldsymbol{h}_{n-1}, \boldsymbol{h}_{n-1}\right) \\ \overline{\boldsymbol{Y}}_n &= \mathrm{LN}\left(\boldsymbol{h}_{n-1} + \boldsymbol{Y}_n\right) \\ \boldsymbol{h}_n &= \mathrm{LN}\left(\overline{\boldsymbol{Y}}_n + \mathrm{FFN}\left(\overline{\boldsymbol{Y}}_n\right)\right) \end{aligned} \tag{7}$$

where $\mathrm{MHA}(\cdot)$ represents the multi-head attention calculation, $\mathrm{FFN}(\cdot)$ represents the feed-forward operation and $\mathrm{LN}(\cdot)$ represents the layer normalization operation.

After passing through N encoder layers, the final output $\boldsymbol{h}_N \in \mathbb{R}^{(M+1)\times \mathrm{dim}}$ of the encoder layers is obtained. Thus, the cluster feature representation for each cluster is denoted as $\boldsymbol{h}_m^N \in \mathbb{R}^{\mathrm{dim}} (0 \le m \le M)$. The average of all cluster features, represented as $\widehat{\boldsymbol{h}}_N = \frac{1}{M+1}\sum_{m=0}^{M} \boldsymbol{h}_m^N$, serves as the overall graph feature of the UWSN. This value will be used in the decoder stage to construct the context vector for each AUV.

2) Decoder: The decoder inputs the cluster feature vectors from the encoder and the state vectors of the AUV before decoding step t, and outputs the AUV that needs to be moved and the next cluster that the AUV needs to visit. To enable the network to flexibly allocate tasks among multiple AUVs based on the environmental state and avoid conflicts arising from different AUVs visiting the same cluster, the decoder in this paper consists of an AUV selection decoder and a cluster access sequence decoder. The decoding step is to firstly select an AUV to be moved by the AUV selection decoder, and then the cluster access sequence decoder chooses the next cluster that the selected AUV needs to visit.

The AUV selection decoder leverage both the current features of each AUV and the historical route features to make informed decisions regarding which AUV to move next. Specifically, in the decoding phase at step t, the historical route features of AUV_i are denoted as

$$\begin{aligned} \boldsymbol{G}_t^i &= \left[\boldsymbol{g}_0^i, \boldsymbol{g}_1^i, \ldots, \boldsymbol{g}_{t-1}^i\right] \\ \boldsymbol{G}_t &= \left[\max\left(\boldsymbol{G}_t^1\right), \max\left(\boldsymbol{G}_t^2\right), \ldots, \max\left(\boldsymbol{G}_t^U\right)\right] \\ \overline{\boldsymbol{G}}_t &= \mathrm{FFN}\left(\boldsymbol{W}_1 \boldsymbol{G}_t + \boldsymbol{f}_1\right) \end{aligned} \tag{8}$$

where $\boldsymbol{G}_t^i \in \mathbb{R}^{t\times \mathrm{dim}}$ represents the arrangement of the cluster features visited by AUV_i before step t, $\boldsymbol{g}_j^i \in \mathbb{R}^{dim}$ represents the feature vector of the cluster visited by AUV_i at step j, $\boldsymbol{W}_1$ and $\boldsymbol{f}_1$ are learnable parameters. $\mathrm{MAX}(\cdot)$ represents the maximum pooling operation. Similarly, the current features of AUVs can be constructed as follows

$$\begin{aligned} \boldsymbol{B}_t^i &= \left[\boldsymbol{g}_{t-1}^i, E_{t-1}^i\right] \\ \boldsymbol{B}_t &= \left[\boldsymbol{B}_t^1, \boldsymbol{B}_t^2, \ldots, \boldsymbol{B}_t^U\right] \\ \overline{\boldsymbol{B}}_t &= \mathrm{FFN}\left(\boldsymbol{W}_2 \boldsymbol{B}_t + \boldsymbol{f}_2\right) \end{aligned} \tag{9}$$

where E_{t-1}^i represents the VoI collected by AUV before step t, $\boldsymbol{W}_2$ and $\boldsymbol{f}_2$ are learnable parameters. Finally, the historical route features $\overline{\boldsymbol{G}}_t$ and the current features $\overline{\boldsymbol{B}}_t$ of AUVs are projected linearly and then passed through a softmax function to obtain the probability distribution $\boldsymbol{P}_t$ for the selection of each AUV

$$\boldsymbol{P}_t = \mathrm{softmax}\left(\boldsymbol{W}_3\left[\overline{\boldsymbol{B}}_t, \overline{\boldsymbol{G}}_t\right] + \boldsymbol{f}_3\right) \tag{10}$$

where $\boldsymbol{W}_3$ and $\boldsymbol{f}_3$ are learnable parameters. The AUV selection result V_t^i for step t can be determined based on the probability distribution $\boldsymbol{P}_t$. Then the cluster accessed by the AUV at step $t+1$ is determined by the cluster access sequence decoder. Firstly, the characteristics of the cluster in which the AUV is currently located, the VoI of the collected data and the encoding characteristics of the clusters need to be taken into account to construct the eigenvector of AUV_i at step t, denoted as $\boldsymbol{y}_t^i$. A multi-head attention mechanism is used to aggregate the attention based features of AUV_i to all clusters, then the attention weight of AUV_i for selecting each cluster is computed through the single-head attention. The whole process can be expressed as

$$\begin{aligned} \boldsymbol{y}_t^i &= \left(\boldsymbol{W}_4\left[\boldsymbol{g}_{t-1}^i, E_{t-1}^i\right] + \boldsymbol{f}_4\right) + \widehat{\boldsymbol{h}}_N \\ \overline{\boldsymbol{y}}_t^i &= \mathrm{LN}\left(\boldsymbol{y}_t^i + \mathrm{MHA}\left(\boldsymbol{y}_t^i, \boldsymbol{h}_N, \boldsymbol{h}_N\right)\right) \\ u_{i,m}^t &= \begin{cases} \frac{\boldsymbol{Q}_i(\boldsymbol{K}_m)^T}{\sqrt{\dim_k}}, b_m = 0 \\ -\infty, b_m = 1 \end{cases} \end{aligned} \tag{11}$$

where $\boldsymbol{Q}_i = \overline{\boldsymbol{y}}_t^i \boldsymbol{W}_Q \in \mathbb{R}^{\dim}, \boldsymbol{K}_m = \boldsymbol{h}_m^N \boldsymbol{W}_K \in \mathbb{R}^{\dim}$ are query vectors and key vectors, respectively. $\boldsymbol{W}_Q$, $\boldsymbol{W}_K$, $\boldsymbol{W}_4$ and $\boldsymbol{f}_4$ are learnable parameters, b_m denotes whether the cluster C_m is visited by AUV or not. The attention weights of AUV_i to the cluster C_m are defined as $-\infty$ if $\boldsymbol{C}_m$ has already been visited by AUV in order to prevent the repeated visits. From this, the attention weight of AUV_i to all clusters is obtained, which is denoted as $\boldsymbol{u}_i^t = \left\{u_{i,1}^t, u_{i,2}^t, \ldots, u_{i,M}^t\right\}$, then the learning process of $u_{i,m}^t (1 \leq m \leq M)$ is stabilized by the tanh function, and finally the probability of AUV_i visiting each cluster at step t is calculated by the softmax function as

$$p_{i,m}^t = \frac{e^{C \cdot \tanh\left(u_{i,m}^t\right)}}{\sum_{j=1}^{M} e^{C \cdot \tanh\left(u_{i,j}^t\right)}}, 1 \leq m \leq M \tag{12}$$

The next cluster will be visited can be determined based on the probability distribution $p_{i,m}^t$, upon determining the next cluster to be visited, AUV_i updates the environmental state. Subsequently, leveraging this updated state, it identifies the AUV scheduled to move in the $(t+1)$th step and the corresponding cluster for visitation. This decoding process iterates until all clusters have been traversed.

B. Optimal CH Selection Algorithm

The optimal CH selection algorithm, grounded in dynamic programming, takes as input the access sequence of clusters by AUVs, and the output corresponds to the CH associated with the cluster visited by the AUV. The objective of the algorithm is to derive a path that maximizes the VoI retained by the AUV.

We first analyze the relationship between the VoI that AUV has when accessing any adjacent clusters C_ζ and $C_{\zeta+1}$. Assume that cluster C_ζ has an initial VoI of E_ζ, $V_n[\zeta]$ denotes the VoI of data retained by the AUV when the AUV visits the nth sensor node of cluster C_ζ, then the AUV moves to the mth sensor node of cluster $C_{\zeta+1}$ for data collection, the data VoI that the AUV has is denoted as $V_{n\to m}[\zeta+1]$, $V_n[\zeta]$ and $V_{n\to m}[\zeta+1]$ can be expressed as

$$\begin{aligned} V_n[\zeta] &= \sum_{i=1}^{\zeta-1} \beta E_i + \sum_{i=1}^{\zeta-1} (1-\beta) E_i e^{-\frac{\Delta T_{i\to\zeta}^n}{\alpha}} \\ V_{n\to m}[\zeta+1] &= \sum_{i=1}^{\zeta} \beta E_i + \sum_{i=1}^{\zeta} (1-\beta) E_i e^{-\frac{\left(\Delta T_{i\to\zeta}^n + \Delta T_\zeta^{n\to m}\right)}{\alpha}} \end{aligned} \tag{13}$$

where $\Delta T_{i\to\zeta}^n$ denotes the time difference between when AUV starts collecting data from cluster C_i and reaches the nth sensor node of cluster C_ζ. It is evident that selecting different CHs will make $\Delta T_{i\to\zeta}^n$ different, subsequently influencing $V_n[\zeta]$, $\Delta T_\zeta^{n\to m}$ denotes the time required for the AUV to collect data from the nth sensor node in cluster C_ζ and move to the mth sensor node in cluster $C_{\zeta+1}$. The first term in $V_n[\zeta]$ and $V_{n\to m}[\zeta+1]$ remains constant over time, while the second term diminishes over time and is affected by the CH selection. We define $V_{n,var}[\zeta]$ as decaying VoI of $V_n[\zeta]$, which can be expressed as

$$V_{n,var}[\zeta] = \sum_{i=1}^{\zeta-1} E_i e^{-\frac{\Delta T_{i\zeta\zeta}^n}{\alpha}} \tag{14}$$

Therefore, during the selection of the CH, it is only necessary to ensure the maximization of decaying VoI to guarantee the maximization of the final obtained VoI. Similarly, $V_{n\to m,var}[\zeta+1]$ can be defined as decaying VoI of $V_{n\to m}[\zeta+1]$, and its state transition relationship with $V_{n,var}[\zeta]$ can be defined as follows according to (13) and (14)

$$V_{n\to m,var}[\zeta+1] = e^{-\frac{\Delta T_\zeta^{n\to m}}{\alpha}} \left(V_{n,var}[\zeta] + E_\zeta\right) \tag{15}$$

It is evident that when sensor nodes m and n are determined, maximizing $V_{n\to m,var}[\zeta+1]$ requires maximizing $V_{n,var}[\zeta]$. We define $V_{n\to m,var}^*[\zeta+1], V_{n,\mathrm{var}}^*[\zeta]$ are the maximum values corresponding to $V_{n\to m,var}[\zeta+1], V_{n,\mathrm{var}}[\zeta]$ respectively. Furthermore defining $V_{m,var}^*[\zeta+1]$ as the maximum value of the decay over time portion of the data VoI retained by the AUV when it reaches the mth sensor node of cluster $C_{\zeta+1}$, which is expressed as

$$V_{m,var}^*[\zeta+1] = \max_{n\in C_\zeta} e^{-\frac{\Delta T_\zeta^{n\to m}}{\alpha}} \left(V_{n,var}^*[\zeta] + E_\zeta\right) \tag{16}$$

The equation above represents the state transition equation for the decayed VoI of the AUV between cluster C_ζ and

Algorithm 1 Optimal CH Selection Algorithm Based on Dynamic Programming

```
Initialize $dp \in \mathbb{R}^{(z_i+2)\times N_{\max}}$, $p \in \mathbb{R}^{(z_i+1)\times N_{\max}}$, $\zeta = 0$
while $\zeta \leq z_i$ do
    for $m$ in $C_{\zeta+1}$ do
        if $x_{\zeta+1,n} = 1$ then
            Update $dp[\zeta+1][m]$ and $p[\zeta][m]$ according to (17)
        else
            $dp[\zeta+1][m] = 0$
            $p\,[\zeta][m] = -1$
        end if
    end for
    $\zeta \leftarrow \zeta + 1$
end while
Initialize $pre = p\,[z_i]\,[0]$, CH set $\boldsymbol{H} = \{pre\}$
while $z_i > 1$ do
    $pre = p\,[z_i - 1]\,[pre]$
    Insert $pre$ in the first place of $\boldsymbol{H}$
    $z_i \leftarrow z_i - 1$
end while
```

cluster $C_{\zeta+1}$. Since the VoI remains fixed without time decay and is independent of CH selection, the objective of CH selection is to maximize $V_{var}^{*}\,[z_i+1]$, which represents the maximum data VoI when the AUV returns to the surface station. Based on (16), the optimal solution $V_{var}^{*}\,[z_i+1]$ can be obtained through dynamic programming. We define array $dp[\zeta][n] = V_{n,var}^{*}[\zeta]$ to represent the maximum retained decayed VoI when the AUV visits the nth sensor node in cluster C_ζ for data collection. To obtain the selected CH for each cluster, we define an array $\mathrm{p}[\zeta][m]$ to store the sensor node in the previous cluster corresponding to $dp[\zeta+1][m]$. In this case, we have the equations as follows

$$\begin{aligned} \mathrm{dp}[\zeta+1][m] &= \max_{n\in \boldsymbol{C}_\zeta} e^{-\frac{\Delta T_{\zeta}^{n\to m}}{\alpha}}\left(dp[\zeta][n] + E_\zeta\right) \\ \mathrm{p}[\zeta][m] &= \arg\max_{n\in \boldsymbol{C}_\zeta} e^{-\frac{\Delta T_{\zeta}^{n\to m}}{\alpha}}\left(dp[\zeta][n] + E_\zeta\right) \end{aligned} \tag{17}$$

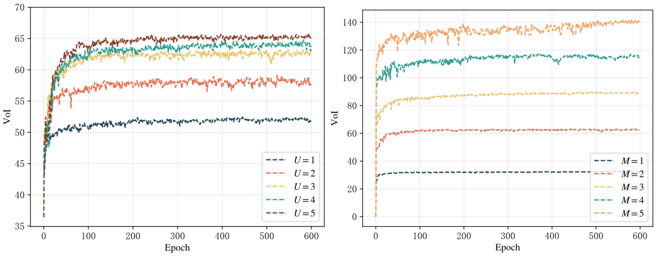
Additionally, the selected CH needs to ensure that the energy consumption of the nodes within the cluster satisfies constraint (1). For any node in cluster C_ζ serving as the CH, we define binary variable $x_{\zeta,n} \in \{0,1\}$ to indicate whether the energy consumption of the nodes within the cluster satisfies the constraint when node n serves as the CH. The optimal CH selection algorithm is detailed in Algorithm 1.

C. Model Training Based on REINFORCE

The parameters of the encoder-decoder based network need to be trained to maximize the system VoI. In this section, we consider training the network parameters using a Monte Carlo-based REINFORCE algorithm. Given an instance s of the UWSN, the network parameters are denoted by $\boldsymbol{\theta}$, and the network outputs the result of the cluster access sequence for each AUV, which is represented as $\boldsymbol{\pi}$. With $P_{\boldsymbol{\theta}}(\boldsymbol{\pi} \mid s)$ representing the probability distribution of network output actions. According to the REINFORCE algorithm, the gradient of our objective $\mathcal{L}(\boldsymbol{\theta})$ is expressed as

$$\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}) = \tfrac{1}{B}\textstyle\sum_{i=0}^{B}\left[\left(V\left(\boldsymbol{\pi}_i \mid s_i\right) - V_{bl}\left(s_i\right)\right)\nabla_{\boldsymbol{\theta}}\log P_{\boldsymbol{\theta}}\left(\boldsymbol{\pi} \mid s_i\right)\right] \tag{18}$$

where $V_{bl}\,(s_i)$ is the reward obtained from the baseline network, which is used to estimate the average performance of the encoder-decoder based network in determining the AUV cluster access sequence. Before training, the network model parameters $\boldsymbol{\theta}$ and the baseline model parameters $\boldsymbol{\theta}_{bl}$ are identical. After training, the parameters of the network model are updated, and if the improvement of $\boldsymbol{\theta}$ is significant according to a paired t-test (5%), it is copied to $\boldsymbol{\theta}_{bl}$.



(a) AUV number (b) Cluster number

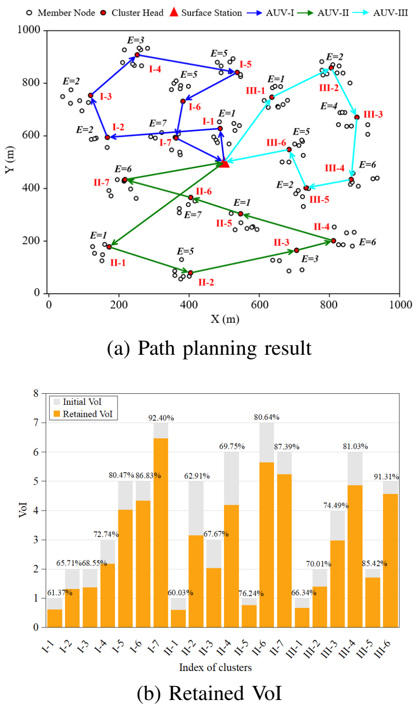
Fig. 2: The convergence of the proposed algorithm under different parameters.

IV. Performance Evaluation

In the simulation, we consider deploying multiple AUVs with a speed of 5 $\mathrm{m/s}$ for data collection in an underwater environment of $1000\ \mathrm{m} \times 1000\ \mathrm{m} \times 100\ \mathrm{m}$, twenty clusters are deployed within, each containing 5 to 10 sensor nodes that are randomly deployed. The initial VoI for data in each cluster is randomly generated between $[1,7]$, with β set to 0.5 and decay factor α set to 200. The amount of data perceived by sensor nodes in each round is randomly generated between $40-50$ kb. The transmission power and reception power of sensor nodes are set to $300\mathrm{mW}$ and $100\mathrm{mW}$ respectively. The maximum energy consumption constraint for nodes during each round is set to 1.2J. To minimize the random effects of simulation parameters, the results of all algorithms were averaged from 20 experiment repetitions.

We first focus on the convergence of multi-AUV path planning networks with different parameters. Fig. 2 demonstrates that the proposed algorithm achieves stable convergence. As depicted in Fig. 2a, as the number of AUVs increases, the convergence of the network decelerates. This is due to more exploration is required to fully learn the task allocation among AUVs. Meanwhile, the VoI retained by AUVs amplifies with the increase in the number of AUVs. This trend stems from the reduced average distance traveled by each AUV. Fig. 2b shows the variation of system VoI with epoch when the number of clusters increases. It can be seen that as the number of clusters increases, the VoI retained by the AUVs increases, but also the network convergence speed decreases as more exploration is required to learn the path planning of AUVs.

Fig. 3 displays an example of the AUV movement paths planned by our algorithm and the percentage of VoI retained for each cluster in an UWSN. As shown in Fig. 3a, each AUV tends to visit a similar number of clusters, and the clusters with



(a) Path planning result

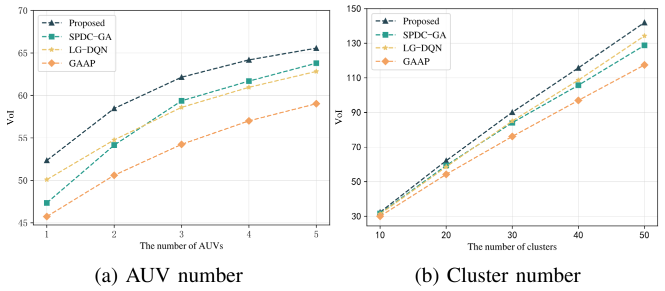
(b) Retained VoI

Fig. 3: Example AUV path planning in the 2D plane with the percentage of VoI retained with 3 AUVs.

higher data importance tend to be visited later. The trade-off between the data importance and timeliness is also shown. The selected CHs effectively shorten the travel distance of AUVs, reducing the loss of VoI. Fig. 3b shows the retained VoI of each cluster along the AUV movement path. All clusters preserve more than 50% of VoI upon the AUV's return to the surface station. Furthermore, for the last few clusters accessed, each one retains over 80% of VoI. This supports accessing clusters with higher data importance later in the sequence.

We then verify the effect of different parameters on the VoI of the system. For comparison, we consider three benchmark schemes: GAAP [8], LG-DQN [10], and SPDC-GA [11]. As depicted in Fig. 4a, the VoI collected by each algorithm increases with the number of AUVs. Notably, our algorithm consistently outperforms others, yielding the highest data VoI. This is due to the efficient learning of AUV cluster access sequences and the selection of optimal CHs in accordance with the sequences, thereby enhancing retained data VoI while adhering to node energy constraints. Fig. 4b demonstrates the relationship between the VoI obtained by each algorithm for different number of clusters. The VoI collected by each algorithm exhibits an increasing trend with the number of clusters, and our algorithm consistently achieves the highest data VoI across different cluster numbers. This is because our algorithm uses the optimal result as the final output, which improves the overall data VoI.

V. Conclusion



(a) AUV number

(b) Cluster number

Fig. 4: Comparison of collected VoI by different algorithms.

In this paper, we investigated the issue of data collection in UWSNs assisted by multiple AUVs. We formulated this problem as a challenging combinatorial optimization problem, aiming to maximize the VoI of the collected data. By employing EDDRL algorithm, we successfully addressed the joint optimization of CH selection and multi-AUV path planning. Simulation results demonstrate that the proposed algorithm converges faster and achieves higher VoI.

Acknowledgment

This research was supported by the European Union's Horizon 2020 programme under grant No 824019 and 101022280, Horizon European program under grant No 101086228, EPSRC with RC Grant No EP/Y027787/1 and UKRI under grant No EP/Y028317/1.

References

[1] T. Qiu, Z. Zhao, T. Zhang, C. Chen, and C. P. Chen, "Underwater internet of things in smart ocean: System architecture and open issues," *IEEE Trans. Industr. Inform.*, vol. 16, no. 7, pp. 4297–4307, 2019.

[2] X. Wei, H. Guo, X. Wang, X. Wang, and M. Qiu, "Reliable data collection techniques in underwater wireless sensor networks: A survey," *IEEE Commun. Surv. Tut.*, vol. 24, no. 1, pp. 404–431, 2021.

[3] S. Song, J. Liu, J. Guo, B. Lin, Q. Ye, and J. Cui, "Efficient data collection scheme for multi-modal underwater sensor networks based on deep reinforcement learning," *IEEE Trans. Veh. Technol.*, 2022.

[4] X. Zhuo, M. Liu, Y. Wei, G. Yu, F. Qu, and R. Sun, "Auv-aided energy-efficient data collection in underwater acoustic sensor networks," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 10010–10022, 2020.

[5] G. Han, A. Gong, H. Wang, M. Martínez-García, and Y. Peng, "Multi-auv collaborative data collection algorithm based on q-learning in underwater acoustic sensor networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 9, pp. 9294–9305, 2021.

[6] M. Xi, J. Yang, J. Wen, H. Liu, Y. Li, and H. H. Song, "Comprehensive ocean information-enabled auv path planning via reinforcement learning," *IEEE Internet Things J.*, vol. 9, no. 18, pp. 17440–17451, 2022.

[7] M. Huang, K. Zhang, Z. Zeng, T. Wang, and Y. Liu, "An auv-assisted data gathering scheme based on clustering and matrix completion for smart ocean," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 9904–9918, 2020.

[8] P. Gjanci, C. Petrioli, S. Basagni, C. A. Phillips, L. Bölöni, and D. Turgut, "Path finding for maximum value of information in multi-modal underwater wireless sensor networks," *IEEE Trans. Mob. Comput.*, vol. 17, no. 2, pp. 404–418, 2017.

[9] R. Duan, J. Du, C. Jiang, and Y. Ren, "Value-based hierarchical information collection for auv-enabled internet of underwater things," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 9870–9883, 2020.

[10] J. Wang, S. Liu, W. Shi, G. Han, and S. Yan, "A multi-auv collaborative ocean data collection method based on lg-dqn and data value," *IEEE Internet Things J.*, 2023.

[11] Y. He, G. Han, Z. Tang, M. Martínez-García, and Y. Peng, "State prediction-based data collection algorithm in underwater acoustic sensor networks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 4, pp. 2830–2842, 2021.