RESEARCH ARTICLE

Energy-Efficient Underwater Data Collection: A Q-Learning Based Approach

Haiyan Zhao[1,2], Jing Yan[1,2], Tao Wu[3], Aihong Li[1,2] and Xiaoyuan Luo[1]

Abstract

Underwater data collection is an importance part in the process of network monitoring, network management and intrusion detection. However, the limited-energy of nodes is a major challenge to collect underwater data. The solution of this problem are not only in the hands of network topology but in the hands of path of autonomous underwater vehicle (AUV). With the problem in hand, an energy-efficient data collection scheme is designed for mobile underwater network. Especially, the data collection scheme is divided into two phases, i.e., routing algorithm design for sensor nodes and path planing for AUV. With consideration of limited-energy and network robustness, Q-learning based dynamic routing algorithm is designed in the first phase to optimize the routing selection of nodes, through which a potential-game based optimal rigid graph method is proposed to balance the trade-off between the energy consumption and the network robustness. With the collected data, Q-learning based path planning strategy is proposed for AUV in the second phase to find the desired path to gather the data from data collector, then a mode-free tracking controller is developed to track the desired path accurately. Finally, the performance analysis and simulation results reveal that the proposed approach can guarantee energy-efficient and improve network stability.

1 Introduction

Underwater cyber-physical system (UCPS) is a new complex system with high efficiency communication and effective control. It is widely used in various underwater engineering and research fields, such as resource protection,

---

Article Highlights

- The energy consumption, network robustness, and environmental obstacles are all considered for underwater data collection.
- A Q-learning based dynamic routing algorithm is proposed to balance energy consumption and the network robustness.
- A Q-learning based path planning algorithm is developed to guide the movement of AUV.

✉ Jing Yan
  jyan@ysu.edu.cn

1   Department of Automation, Yanshan University, Qinhuangdao 066004, China

2   Key Laboratory of Ocean Observation Technology Ministry of Natural Resources, National Marine Technology Center, Tianjin 300110, China

3   Department of Frontier and Innovation Research, Wuhan Second Ship Design and Research Institute, Wuhan 430205, China

ocean environment monitoring, and early warning system of natural disaster (Gong et al. 2020; Lin et al. 2019; Wang et al. 2015). In order to support these applications, the data acquisition of underwater environment becomes a primary task to be solved. To handle this problem, internet of underwater things (IoUT), as the core components of UCPS, has attracted the interest of researches, which has advantages of low cost, rapid deployment, and self-organization (Cashmore et al. 2018; Li et al. 2019; Yan et al. 2020a).

Different from ground environment, the underwater environment has many special characteristics, which poses a substantial challenge to the underwater data acquisition. Firstly, the robustness of networks is more demanding since IoUT is a sparse network. To improve the network robustness, a common method is to improve network connectivity. However, a good connectivity can increase the energy consumption of network, which leads to shorten network lifetime. Secondly, the power of nodes in IoUT is difficult to recharge since the nodes are powered by batteries (Lin et al. 2016; Yan et al. 2021a, 2021b). Thirdly, the high-frequency radio waves are strongly absorbed in underwater environment, which cannot used in underwater scenario. Till now, the acoustic wave as the only available signal for underwater communication, but the transmission

rate of acoustic wave is 1.5×10^3 m/s, which is much slower than radio wave (Li et al. 2016b; Su et al. 2019; Yan et al. 2020b). Due to these challenges, the data collection schemes proposed for terrestrial sensor network are no longer suitable for IoUT.

Up to now, underwater data collection schemes can be divided into two categories: data collection by multi-hop transmission and data collection by autonomous underwater vehicle (AUV). To be specific, multi-hop transmission mainly transmits the data to the receiver (or sink node) on the water surface through a series of routing mechanisms. However, the nodes that near to the receiver consume energy quickly. Thereby, there is an imbalance of energy consumption in multi-hop transmission, which is easy to lead to the emergence of energy voids (Ren et al. 2016). For instance, Xie et al. (2006) firstly designed a vector-based forwarding (VBF) protocol, which can provide robust, scalable and energy-efficient routing. Besides that, a power-aware data collection algorithm was developed (Dhurandher et al. 2009). However, these literature were not suitable for sparse networks due to substantial energy consumed by this protocol via flooding. To reduce energy consumption, Wu et al. (2021) provided a Hierarchical Adaptive Energy-efficient Clustering Routing (HAECR) strategy to prolong the network lifetime. In Ali et al. (2014), a novel routing strategy named Layer by layer Angle-based flooding (L2-ABF) was provided under the constraints of movement of nodes, delays and energy consumption. Inspired by Xie et al. (2006), an adaptive hop-by-hop vector-based forwarding routing (AHH-VBF) protocol was designed (Yu et al. 2015). Some other multi-hop transmission based data collection schemes were presented (Jin et al. 2017; Wei et al. 2013). Although these schemes can achieve energy efficient, the robustness of the network is not considered. For the second category, the data collection task is completed by assistance of AUV. In Khan and Cho (2014; 2016), multiple AUVs assisted data collection method was proposed. Besides that, Han et al. (2019) developed a prediction-based delay optimization data collection method to solve the issue of data collection delay. In Duan et al. (2020), the value of information was utilized as a metric to measure the quality of information. In view of this, a reliable hierarchical information collection system was constructed for data collection. In addition, Cai et al. (2019) applied the mobile edge elements into data collection algorithm, whose results were helpful to solve the problem of unbalanced energy consumption. In Han et al. (2018), a stratification-based data collection algorithm was proposed for underwater data collection with constraints of water delamination and limited-energy. Although these algorithms help to improve the energy efficient, how to improve the robustness of network has not been well studied. We notice that topology is a good way to balance network connectivity and energy consumption. Referring to Shames

and Summers (2015), we found that the rigid characteristics of the network are closely related to its algebraic properties. In Luo et al. (2018), the potential-game based optimally rigid topology control algorithm was proposed to improve the network performance. The rigid graph based topology optimization algorithm was employed to balance the trade-off between energy consumption and network connectivity in Yan et al. (2016). Although the works in Luo et al. (2018) and Yan et al. (2016) demonstrate a good trade-off between energy consumption and connectivity maintenance, the relationship between topology optimization and routing strategy has not been studied. With regards to limited-energy and network connectivity, how to adopt the topology optimization technology to balance the trade-off between the energy efficient and network connectivity for routing algorithm is still an open issue.

With the assumption of the data is collected by data collector, some path planning schemes have been proposed for AUV. For instance, the path planning problem of multiple robots was studied for data collection in Bhadauria et al. (2011). In McMahon and Plaku (2017), the sampling-based motion planning were incorporated into constraint-based solvers to achieve autonomous data collection. Nevertheless, the methods in Bhadauria et al. (2011) and McMahon and Plaku (2017) only consider the effect of geographical location on AUV data collection, the validity of the data is not considered. Since the data generated by the nodes has associated values that decay over time, it is necessary to consider the value information of the collected data. Based on this, the timeliness of the data was used to plan the path of AUV in Gjanci et al. (2018), Liu et al. (2022), then maximize the effective value of the total data collection. Our previous work (Yan et al., 2018) developed a dynamic value-based path planning method for AUV to maximize the value of information. However, the obstacle avoidance is not addressed. With consideration of geographical location of nodes, timeliness of the data and obstacle, how to design the path planning and tracking strategy of AUV is another problem to be solved.

With these problems in hand, an AUV-assisted energy-efficient data collection scheme is proposed in this paper. Two phases data acquisition algorithm is designed, i. e., routing algorithm design for sensor nodes and path planing for AUV. In the first phase, the optimal rigid graph was incorporated into dynamic routing algorithm for sensor nodes to transmit the data to data collector. Of note, the proposed method can balance the trade-off between the energy consumption and the network robustness. Subsequently, the Q-learning based AUV path planning strategy is developed in the second phase to periodically accesses the data collector and periodically resurfaces the water to unload the collected data to the control center. In particular, the main contributions of this paper are as follows:

1) Routing algorithm design for sensor nodes. The po-

tential game based optimal rigid graph is designed to balance the trade-off between the energy consumption and network connectivity, and then a Q-learning based dynamic routing algorithm is provided for sensor nodes to relay the collected data to data collector. Compared with the existing works (Li et al. 2016a; Yan et al. 2018), the routing algorithm in this paper can prolong the network lifetime and improve the robustness of network.

2) Path planning and tracking for AUV. A Q-learning based path planning algorithm is developed for AUV to guide the movement of AUV, and then a proportional-integral-derivative (PID) based tracking algorithm is proposed to track the desired path. Compared with the existing work (Han et al. 2017), the path planning algorithm in this paper can improve the information value of the total data collection with a time limit and avoid the influence of environmental obstacles.

2 System model and problem formulation

2.1 System model

As depicted in Figure 1, a network including sensor nodes, data collectors and AUV is designed for data collection.

• Sensor Nodes. The role of sensor nodes is to perform underwater monitoring tasks, whose clocks are synchronized and the coordinates are accurately known. It should be emphasized that, sensor nodes can move passively due to the effect of current.

• Data Collectors. Data collectors are static nodes, whose function is to gather data from sensor nodes through some routing mechanisms. Particularly, the locations of data collectors are fixed with the anchor wire constraints.

• AUV. An AUV is deployed to visit data collectors, through which the data can be retrieved by high speed visible light communications. Meanwhile, AUV periodically surfaces to offload the retrieved data to control center via high speed radio communications. Without loss of generality, we assume that the batteries equipped on AUV are sufficient.

Similar to Yan et al. (2018), the underwater monitoring area is divided into M subareas. It is noted that N sensor nodes and one data collector are deployed in each subarea. For an arbitrary subarea, sensor nodes collect data from the surrounding environment and send the collected data to data collector through single-hop or multi-hop communication. Without loss of generality, we assume that a set E of events $E_1, \cdots, E_{|E|}$ occur in the underwater monitoring area, where $|E|$ is the total number of events. When sensor node $i \in \{1, \cdots, N\}$ has sensed an event E_k at time $t_{k,i}$, the value of information (VoI) $C_{E_k,i}$ for the sensed data on event E_k can be acquired by sensor i, where $1 \leq k \leq |E|$. Specially, the VoI $C_{E_k,i}$ at time t is defined as
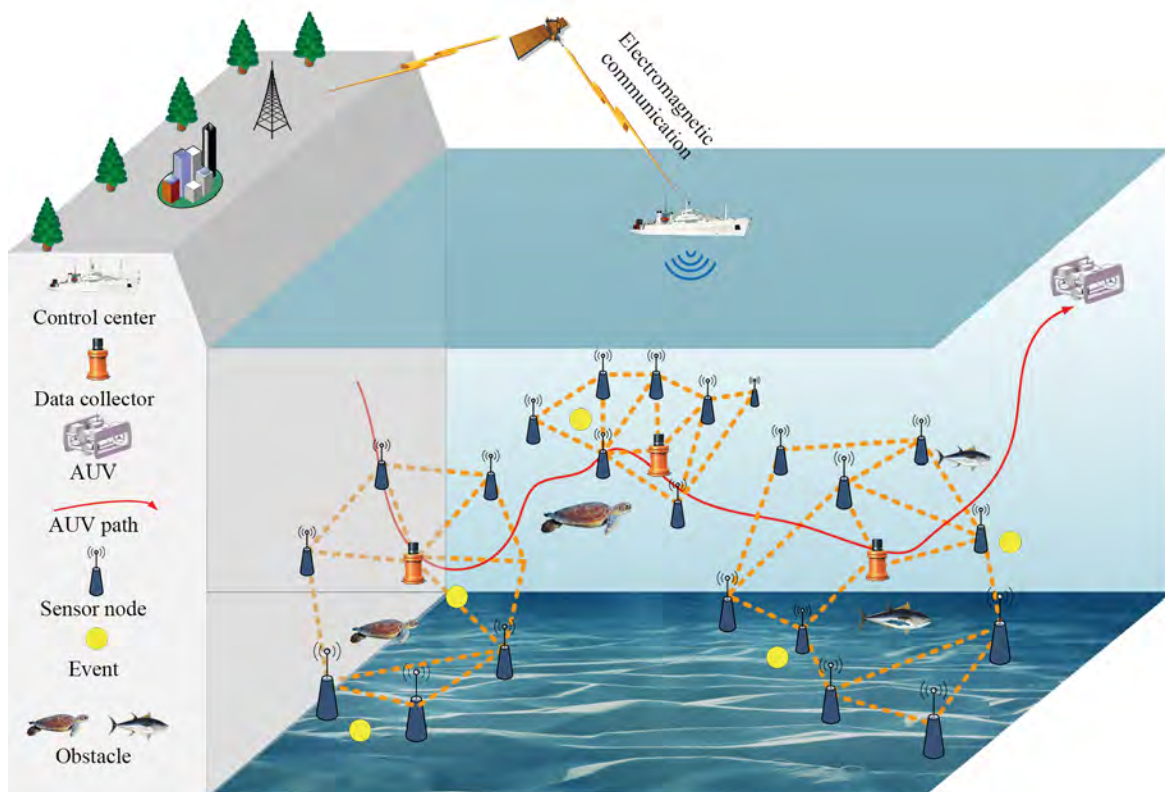
Figure 1   Description of the network architecture, where an AUV is employed to retrieve data from multiple data collectors

$$C_{E_k,i}(t) = \beta_k F_{E_k} + (1 - \beta_k)e^{-(t-t_{k,i})}, \text{ for } t \geq t_{k,i} \quad (1)$$

where $F_{E_k} \in \mathcal{R}^+$ and $e^{-(t-t_{k,i})}$ denote the event importance and timeliness of E_k, respectively. $0 \leq \beta_k \leq 1$ represents the information weight, whose role is to balance the trade-off between importance and timeliness. It is noted that, the event importance can be designed and modified according to the monitoring levels. Meanwhile, the event timeliness is a monotonically decreasing function, which is to captures the temporal decay of the sensed data. This is intuitive as later retrieve of a data segment can lead to higher losses.

Different from the terrestrial networks, the sensor nodes in IoUT move passively since the effect of current. Based on this, a meandering current mobility model (Xia et al. 2017) is utilized to describe the movement of sensor nodes. Accordingly, the movement of sensor node $i \in \{1, \cdots, N\}$ can be updated as

$$x_i(t+1) = x_i(t) - \rho\frac{\partial\psi(x_i, y_i, t)}{\partial y_i}$$
$$y_i(t+1) = y_i(t) - \rho\frac{\partial\psi(x_i, y_i, t)}{\partial x_i} \quad (2)$$
$$z_i(t+1) = z_{\text{fix},i} + \vartheta_i$$

where $(x_i, y_i, z_i) \in \mathcal{R}^3$ denotes the position of sensor node i, ψ
$(x_i, y_i, t) = -\tanh\left[\dfrac{y_i - B(t)\sin(\kappa(x_i - ct))}{\sqrt{1 + \kappa^2 B^2(t)\cos^2(\kappa(x_i - ct))}}\right]$, $\rho \in \mathcal{R}^+$ is
an iteration scalar, $z_{\text{fix},i} \in \mathcal{R}$ is the predefined fixed depth of sensor node i, $\vartheta_i \in \mathcal{R}$ follows White Gaussian noise (WGN), $\kappa \in \mathcal{R}$ is the number of meanders in the unit length, and $c \in \mathcal{R}$ denotes the phase speed. Particularly, $B(t) = A + \epsilon \cos(\omega t)$ modulates the width of meanders, wherein $A \in \mathcal{R}$ resolves the average meander width, $\epsilon \in \mathcal{R}$ represents the amplitude of the modulation, and $\omega \in \mathcal{R}$ denotes its frequency. In a data gathering cycle, we assume that the change of water current is not frequent and the sensor nodes cannot move out of the predefined subareas.

Referring to Sehgal et al. (2011), the energy consumption when transmitting one packet over distance d can be calculated as

$$E_{\text{rx}} = 4\pi d^2 10^{\frac{\text{SL} - 20\log\vec{l} - \alpha\vec{l}\times10^{-3} - \text{TL}}{10}} T_{\text{tx}} \quad (3)$$

where d is the transmission distance between one sensor node and its next-hop node, $\text{SL} \in \mathcal{R}^+$ denotes the sonar source level, $\vec{l}$ denotes the transmission loss range, α represents the absorption coefficient in dB/km, TL denotes the transmission loss, and T_{tx} is the transmission time of a single packet.

It is noted that, the body-fixed reference frame (BRF) and inertial reference frame (IRF) are adopted to describe the AUV dynamics. Due to the intrinsic stability of rolling and pitching of open-frame vehicles (Caccia and Veruggio 2000; Kim et al. 2016), this paper presents a simplified model including four degrees of freedom (4-DOF), i.e., neglecting pitch and roll motions. Based on this, the dynamics of AUV is formulated as

$$u(t+1) = u(t) + \frac{\rho}{m_u}(F_u + m_v vr - k_u u - k_u^* u|u|)$$
$$v(t+1) = v(t) + \frac{\rho}{m_v}(F_v - m_u ur - k_v v - k_v^* v|v|)$$
$$\quad (4)$$
$$w(t+1) = w(t) + \frac{\rho}{m_w}(F_w + W - k_w w - k_w^* w|w|)$$
$$r(t+1) = r(t) + \frac{\rho}{I_r}(T_r - (m_v - m_u)uv - k_r r - k_r^* r|r|)$$

where u, v, and w denote the linear velocities in surge, sway and heave, respectively. r denotes the angle velocity in yaw. Besides, m_u, m_v, and m_w denote the masses in surge, sway and heave, respectively. I_r is the moment of the inertia in yaw. F_u, F_v and F_w are the exerted forces in surge, sway and heave, respectively. T_r is the exerted torque in yaw and W is the buoyancy force in heave. k_u, k_v, k_w, and k_r are the linear damping scales in surge, sway, heave and yaw, respectively. $k_u^* u|u|$, $k_v^* v|v|$, $k_w^* w|w|$ and $k_r^* r|r|$ are the quadratic damping constants in surge, sway, heave and yaw, respectively. In view of (4), the kinematics of AUV can be described as

$$x(t+1) = x(t) + \rho(u(t)\cos\varphi(t) - v(t)\sin\varphi(t))$$
$$y(t+1) = y(t) + \rho(u(t)\sin\varphi(t) + v(t)\cos\varphi(t))$$
$$z(t+1) = z(t) + \rho w(t)$$
$$\varphi(t+1) = \varphi(t) + \rho r(t)$$
$$\quad (5)$$

where x, y, and z are the positions on X, Y and Z axes, respectively. φ denotes the angle in yaw.

2.2 Problem formulation

Problem 1 (Routing strategy design for sensor nodes) Since the complexity of underwater environment, such as coral reefs and fish stocks, it affects data transmission and requires higher robustness to networks. Meanwhile, the battery power installed on underwater sensor node is limited and cannot be recharged. In order to avoid the energy holes and prolong the network lifetime, the optimal rigid graph is incorporated into dynamic routing algorithm to relay the data from sensor nodes to data collector. This problem is reduced to optimize the routing selection of sensor nodes with the consideration of energy holes and network robustness.

Problem 2 (Path planning and tracking for AUV) In mobile IoUT, data collectors gather data from sensor nodes, but the collected data is decaying with time. To maximize the VoI of data delivered to control center, we aim to de-

sign a Q-learning based path planning strategy for AUV to find the desired path, which can avoid environmental obstacles effectively. Of note, the AUV dynamics are highly complex, and the exact dynamic parameters of AUV are not easy to acquire due to the influence of ocean currents. To solve this issue, a model-free tracking tracking controller is sought to be provided, such that AUV can accurately track the desired path.

3 Energy-efficient data collection approach

In this section, the energy-efficient data collection approach is developed. As shown in Figure 2, the approach is divided into two phases, i.e., 1) routing strategy for sensor nodes; 2) path planning and tracking for AUV. The detailed process of the two phases is described below.
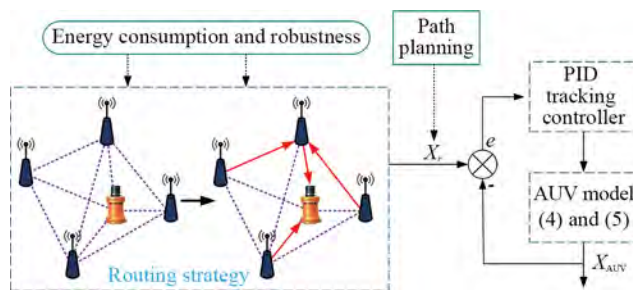

Figure 2   Framework of energy-efficient data collection approach

3.1 Routing strategy for sensor nodes

Without loss of generality, the routing strategy of sensor node is divided into two stages, i.e., optimal rigid graph design and dynamic routing algorithm design. In the following, the routing strategy for sensor nodes is given.

3.1.1 Optimal rigid graph design

For ease of description, an arbitrary subarea S_1 is taken as an example. Without loss of generality, subarea S_1 can be modelled as a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, in which $\mathcal{V}=\{1, 2, \cdots, N, N+1\}$ is the set of sensor nodes and data collector in subarea S_1, and $\mathcal{E}=\{(i, j)\in\mathcal{V}\times\mathcal{V}:i\neq j\}$ is the edge set. Of note, each edge in the graph is expressed by a different pair of vertices (i, j). If any vertex pair (i, j) of graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ satisfies $\forall (i, j)\in\mathcal{E}\Rightarrow(j, i)\in\mathcal{E}$, then graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is called an undirected graph, otherwise it is called directed graph. Especially, graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is called rigid graph if vertex is removed does not destroy the stability of graph. Otherwise, it is called a flexible graph. It is noted that, it is called minimum rigid graph if the rigid graph becomes flexible when any an edge is removed from rigid graph. For a clearer description, examples of a flexible graph, a rigid graph, and a minimum rigid graph are given in Figure 3. Accordingly, the following definition and lemma are given.

(a) Flexible graph   (b) Rigid graph   (c) Minimum rigid graph
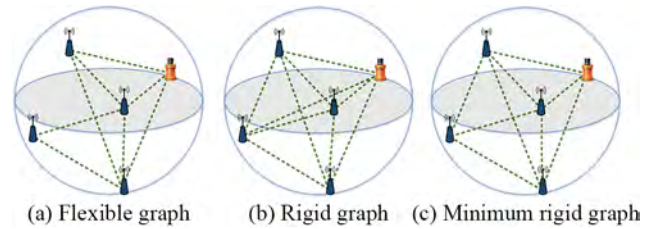
Figure 3   Example of flexible and rigid graphs

Definition 1 (Zhang et al. 2015) Defining $\mathcal{R}_{(\mathcal{G})}\in\mathcal{R}^{|\mathcal{E}|\times|3\mathcal{V}|}$ is the rigidity matrix of a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with $N+1$ vertices in 3D space. The graph is called a infinitesimally rigid graph if and only if the rank of $\mathcal{R}_{(\mathcal{G})}$ is $3(N+1)-6$, i.e, rank$(\mathcal{R}_{(\mathcal{G})})=3(N+1)-6$.

Lemma 1 (Luo et al. 2013) $\mathcal{G}_i(i=1, 2, \cdots, N+1)$ is a minimum rigid graph, which composes of node i and its neighborhood. $\mathcal{G}(\mathcal{V}, \mathcal{E})= \bigcup_{i=1, \cdots, N+1}\mathcal{G}_i$. Denote the generated optimal rigid graph as $\mathcal{G}_{\mathrm{opr}}(\mathcal{V}, \mathcal{E}_{\mathrm{opr}})$. Then, the following statements hold: a) $e\notin\mathcal{E}_{\mathrm{opr}}$, if $e=(j, k)\in\mathcal{E}$, j, $k \in \mathcal{G}_i$, and $e \notin \mathcal{G}_i$; b) The optimal rigid graph $\mathcal{G}_{\mathrm{opr}}(\mathcal{V}, \mathcal{E}_{\mathrm{opr}})$ can be obtained by deleting those edges satisfying a).

The aim of this section is to balance the trade-off between the network stability and energy consumption by optimizing network topology. Of note, each node (i.e., data collector or sensor node) in the network is independent and affects each other. Based on this, the game theory is introduced to optimize the network. Without loss of generality, a strategy game consists of three elements, i.e., Player $\mathcal{I}$, Strategic Space $\mathcal{A}$, and Utility Function $\mathcal{U}$. Thus, the strategy game is expressed as $\Gamma\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}$. In the following, the three elements of strategy game are introduced in detail: 1) Player $\mathcal{I}\in\{1, 2, \cdots, i, \cdots, N+1\}$, where $N+1$ is the total number of players in game; 2) Strategic Space $\mathcal{A}:\mathcal{A}= \Pi_{i\in\mathcal{I}} \mathcal{A}_i$ is the set of all possible combinations of policies for all players to choose. $a=(a_i, a_{-i})$ is the strategic of all players, where $a_i\in\mathcal{A}_i$ is the strategy selection by player i and a_{-i} is the strategic selection of players other than player i; 3) Utility Function $\mathcal{U}:\mathcal{A}\rightarrow\mathbb{R}$ for $i \in \mathcal{I}$ is mappings. Based on this, the following definitions are given.

Definition 2 (Nash Equilibrium (NE)) (Chen et al. 2019; Lee et al. 2018) The strategy $a^*=(a_i^*, a_{-i}^*)$ is said to be Nash Equilibrium, if $\mathcal{U}_i(a_i^*, a_{-i}^*)\geqslant\mathcal{U}_i(a_i, a_{-i}^*)$ always holds, where a_i^* is the best strategy of player i to other players, a_{-i}^* is the best strategy of players other than player i.

Definition 3 (Ordinal Potential-Game (OPG) and Ordinal Potential Function (OPF)) (Chen et al. 2019; Lee et al. 2018) For game $\Gamma\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}$, if there exist a function $\mathcal{U}(a)$: $\mathcal{A}\rightarrow\mathbb{R}$, $\forall i\in\mathcal{I}$, $\forall a_{-i}\in \mathcal{A}_{-i}$, for $\forall a_i^a\in \mathcal{A}_i$ and $a_i^b\in \mathcal{A}_i$, the following equation holds

$$\begin{aligned} &\mathrm{sgn}\,(U\,(a_i^b, a_{-i}) - U\,(a_i^a, a_{-i}))\\ &= \mathrm{sgn}\,(\mathcal{U}_i(a_i^b, a_{-i}) - \mathcal{U}_i(a_i^a, a_{-i})) \end{aligned} \tag{6}$$

Then, the game $\Gamma\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}$ is called OPG, and the function

$U(a)$ is called OPF of the game $\Gamma\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}$.

Based on the above definitions, the following two aspects are considered to construct the utility function.

1) Energy consumption of network: Similar to Yan et al. (2018), the relative energy consumption is adopted to describe the energy consumption. Define the required energy at time t as RE(t), which is calculated by (3). The available energy at time t is denoted as AE(t). Based on this, the relative energy consumption of sensor node i is defined as $E_i(t) = \dfrac{\text{AE}_i(t)}{\text{RE}_i(t)}$. For nodes i and j, $i \neq j$ and $i, j \in \{1, 2, \cdots, N+1\}$, we assume that the node i sends the data packet to its neighbor node j, and the available energy of nodes i and j is greater than the required energy. Then, the energy-based model is described as

$$f_1 = \max \sum_i \sum_{j \in N_i} \{ E_i(t) + E_j(t) \} \tag{7}$$

where N_i is the neighborhood node set of node i.

Without loss of generality, we assume that the same parameters (i.e., SL and α) are used in the monitoring area. Then, one has RE$_i(t)$=RE$_j(t)$=RE$_{ij}(t)$. Thus, (7) can be rearranged as $f_1 = \max \sum_i \sum_{j \in N_i} W_{ij}(t)$, where $W_{ij}(t) = (\text{AE}_i(t) + \text{AE}_j(t))/\text{RE}_{ij}(t)$ is the weight between nodes i and j. In view of this, the rigidity matrix $\mathcal{R}_{(\mathcal{G})}$ can be defined as

$$\begin{bmatrix} \cdots & \ddots & \vdots & & \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \dfrac{(p_i^{\mathrm{T}} - p_j^{\mathrm{T}})W_{ij}}{\| p_i^{\mathrm{T}} - p_j^{\mathrm{T}} \|} & \cdots & \dfrac{(p_j^{\mathrm{T}} - p_i^{\mathrm{T}})W_{ij}}{\| p_j^{\mathrm{T}} - p_i^{\mathrm{T}} \|} & 0 & \cdots & 0 \\ \cdots & \ddots & \vdots & & \vdots & & \vdots & & \vdots & \ddots & \vdots \end{bmatrix} \tag{8}$$

where $p_i = \left[p_i^1, p_i^2, p_i^3 \right]^{\mathrm{T}}$ and $p_j = \left[p_j^1, p_j^2, p_j^3 \right]^{\mathrm{T}}$ are the coordinates of nodes i and j, respectively. Of note, each row $[0 \cdots 0 \ \dfrac{(p_i^{\mathrm{T}} - p_j^{\mathrm{T}})W_{ij}}{\| p_i^{\mathrm{T}} - p_j^{\mathrm{T}} \|} \cdots \dfrac{(p_j^{\mathrm{T}} - p_i^{\mathrm{T}})W_{ij}}{\| p_j^{\mathrm{T}} - p_i^{\mathrm{T}} \|} \ 0 \cdots 0]$ corresponds to one edge $(i, j) \in \mathcal{E}$.

2) Stability of network: To ensure the stability of the network, definition 1 must be satisfied. In view of this, the following function is defined

$$h_i(a_i, a_{-i}) = \begin{cases} 1, \ \text{rank}\,(\mathcal{R}_{(\mathcal{G})}) = 3(N+1) - 6 \\ 0, \ \text{otherwise} \end{cases} \tag{9}$$

Subsequently, the vertex rigidity gramian $X_{(\mathcal{G})}$ is given to reflect the algebraic quality of network, i.e., $X_{(\mathcal{G})} = \mathcal{R}_{(\mathcal{G})}^{\mathrm{T}} \mathcal{R}_{(\mathcal{G})} \in \mathcal{R}^{|3\mathcal{V}| \times |3\mathcal{V}|}$. It is noticed that the trace of the vertex rigidity gramian is the sum of the eigenvalues of the rigidity matrix. Therefore, the following optimization problem is given

$$f_2 = \max \text{trace}\,(X_{(\mathcal{G})}) \tag{10}$$

Therefore, the utility function of game $\Gamma\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}$ can be constructed as

$$\mathcal{U}(a_i, a_{-i}) = h_i(a_i, a_{-i}) \text{trace}\,(X_{(\mathcal{G}, a_i)}) \tag{11}$$

where $X_{(\mathcal{G}, a_i)}$ is the vertex rigidity gramian with the strategy a_i.

In view of (11) and Lemma 1, the potential game based optimal rigid graph can be generated, and the corresponding pseudo codes are described in Algorithm 1. It should be emphasized that the Algorithm 1 contains two principles, i. e., 1) keeping the network is a minimum rigid graph, and 2) maximizing the trace of the vertex rigidity gramian.

Algorithm 1 Potential game based optimal rigid graph

Input: A graph with $N+1$ vertices that are labelled as $1, 2, \cdots, N+1$, the position of all nodes $p = \{p_1^1, p_1^2, p_1^3, \cdots, p_i^1, p_i^2, p_i^3, \cdots, p_{N+1}^1, p_{N+1}^2, p_{N+1}^3\}$, the optional strategies spaces $\mathcal{A}_i$;
Output: The optimal rigid graph with the strategy $a_i^* = \{a_1^*, a_2^*, \cdots, a_{N+1}^*\}$;
1: The game implements based on the node ID, $\dot{a}_i = a_i^c$, a_i^c is the optional strategies for $\forall i \in \mathcal{V}$;
2: $\dot{\mathcal{A}} = \{\dot{a}_1, \dot{a}_2, \cdots, \dot{a}_{N+1}\}$;
3: **for** $i \in \mathcal{V}$ **do**
4:      **if** $|\dot{\mathcal{A}}_i| \leq 3(N+1) - 6$ **then**
5:          $\dot{a}_i = \arg\max_{a_i \in \mathcal{A}_i} \mathcal{U}_i(a_i, a_{-i})$;
6:          $|\dot{\mathcal{A}}_i| = |\dot{\mathcal{A}}_i| + 1$;
7:      **end if**
8: **end for**

3.1.2 Dynamic routing algorithm design

With the generated optimal rigid graph in the Section 3.1.1, the communication topology of network can be obtained. To transmit the data from sensor nodes to data collector, the Q-learning based dynamic routing algorithm is designed. As we know, Q-learning is a learning algorithm, the agent (i.e., sensor node) select the optimal action through the interact with an external environment. The learning process of network can be modeled as a Markov decision process with limited state and action space, it can be denoted as $M = (S, A, R)$, where S is the set of nodes (i.e., state set), A is the set of optional node (i.e., action set), and R is the reward function. Of note, each state in this section has $N+1$ optional action. To evaluate the performance of actions, the reward function $R(s_l^1, a_l^1)$ at lth episode step is designed as

$$\mathcal{R}(s_l^1, a_l^1) = \begin{cases} \mathcal{R}_{\max}, \ s_{l+1}^1 \text{ is data collector} \\ -\mathcal{R}_{\max}, \ \text{no communication of two states} \\ -d_{l,l+1}, \ \text{otherwise} \end{cases} \tag{12}$$

where $\mathcal{R}_{\max}$ is the maximum bonus value, which implies that the next state is the data collector. $-\mathcal{R}_{\max}$ is a negative value, indicating that there is no communication between the two states. $d_{l,l+1} = \| p_l - p_{l+1} \|$ is the distance between the nodes l and $l+1$. Besides that, s_l^1, s_{l+1}^1 and a_l^1 are the

current state, the next state and the current action, respectively.

Algorithm 2 Q-learning based dynamic routing algorithm

1: Set the reward function matrix $\mathcal{R}$ according to optimal rigid graph in Algorithm 1;
2: Initialization: $\boldsymbol{Q} = \boldsymbol{0}$, α and γ;
3: **for** $i = 1:N$ **do**
4:     **for** each episode **do**
5:         Randomly select an initial state s_l^1;
6:         **if** s_l^1 is not the data collector **then**
7:             Use action a_l^1 to get the next state s_{l+1}^1;
8:             Calculate $Q(s_l^1, a_l^1)$ by (13);
9:             $s_l^1 \leftarrow s_{l+1}^1$;
10:             $l=l+1$;
11:         **end if**
12:     **end for**
13:     $P_{\text{opt}}(i)=\arg \max(Q)$;
14: **end for**
15: Return optimal routing P_{opt} of sensor nodes.

According to $R(s_l^1, a_l^1)$, the Q function can be updated by the following equation, i.e.,

$$Q(s_l^1, a_l^1) = (1 - \alpha)Q(s_l^1, a_l^1) + \alpha(\mathcal{R}(s_l^1, a_l^1) \\ +\gamma \max(Q(s_{l+1}^1, a_{l+1}^1))) \quad (13)$$

where α is the learning rate, $\gamma \in [0, 1)$ is the discount factor, and a_{l+1}^1 denotes the next action.

Therefore, the Q-learning based dynamic routing algorithm is given in Algorithm 2. It is noted that the sensor node selects the node with the maximum Q value as the next hop to forward data based on the Q table. The sensor node can transmit the data to data collector by shortest distance. Even if one transmission path is interrupted, the sensor node can transmit data to data collector successful. Therefore, it can guarantee to prolong network lifetime and improve the network reliability.

3.2 AUV path planning strategy

With the routing strategy for sensor nodes in Section 3.1, the data collector can collect the data from sensor nodes successfully. Define a binary variable $k_{k,i}$ with $i \in \{1, \cdots, N\}$, $k_{k,i}=1$ if sensor i sense the event E_k, otherwise $k_{k,i}=0$. Similarly, a binary variable $l_{j,i}$ with $j \in \{1, \cdots, M\}$ is defined to express whether data collector j can obtain the VoI of sensor i, i.e., $l_{j,i}=1$ if the data collector j can receive the VoI of sensor i, otherwise $l_{j,i}=0$. Accordingly, the VoI of sensor i and VoI of data collector j from sensors can be calculated as

$$\mathcal{C}_{E,i}(t) = \sum_{k=1}^{|E|} k_{k,i} \mathcal{C}_{E_k,i}(t) \quad (14)$$

$$\mathcal{RC}_j(t) = \sum_{i=1}^{N} l_{j,i} \mathcal{C}_{E,i}(t) \quad (15)$$

With the above discussions, we employ the AUV to collect the information from the data collectors. AUV access the data collector periodically and unload the collected data to the control center. The access period time of AUV is defined as T. It is noted that the time of vertical movement is important for data collection at the next data collection cycle. Based on this, we assume that the depth of the data collector is h and the vertical velocity of AUV is v, then the time that AUV vertical movement to data collector is $t_{\text{vertical}}=\frac{h}{v}$. Therefore, the data collection time of AUV is updated as $\bar{T}=T-2t_{\text{vertical}}$. Besides that, we assume that the communication range of data collector is the maximum accessible distance under the fixed transmission power of AUV. Each data collector broadcasts a data packet to AUV via acoustic communication, which contains the position information of the data collectors and the VoI of the collected events. Accordingly, the income function of data collector j is designed as

$$I_j(t) = \mathcal{RC}_j(t) - \bar{a}D_j(t) \quad (16)$$

where $t \in [0, T]$, $D_j(t)$ is the distance between current position of AUV and data collector j. The purpose of subtraction of $D_j(t)$ is to reduce the distance of access. $\bar{a} \in \mathcal{R}^+$ is a scalar. In view of (16), the income function of AUV can be expressed as

$$I_{\text{AUV}}^*(t) = \max \{ I_1(t), I_2(t), \cdots, I_M(t) \}, t_{\text{in}} < t < \bar{T} \quad (17)$$

where t_{in} is the time that AUV start to collect data.

Algorithm 3 Q-Learning based AUV Path Planning

Input: The VoI of data collector RC_j and the position information $D_j(t)$ are received by AUV at time $t = t_{\text{in}}$, where $j \in 1, 2, \cdots, M$;
Output: The dynamic path planing of AUV;
1: Initialization: $\boldsymbol{Q}_A=\boldsymbol{0}$, α_1 and γ_1;
2: **while** $t_{\text{in}} < t < t_{\text{in}}+\bar{T}$ **do**
3:     Destination is determined by (17);
4:     **for** each episode **do**
5:         Select an initial state s_l^2 randomly;
6:         **if** s_l^2 is not destination **then**
7:             Choose action a_l^2 from s_l^2 using policy derived from $\boldsymbol{Q}_A$;
8:             Take action a_l^2, observe $\boldsymbol{R}_A$ and s_{l+1}^2;
9:             Calculate $Q_A(s_l^2, a_l^2)$ by (19);
10:             $s_l^2 \leftarrow s_{l+1}^2$;
11:             $l=l+1$;
12:         **end if**
13:     **end for**
14:     The income function of destination is reset as zero;
15: **end while**

Based on (17), one knows that AUV dynamically select the data collector with the largest income function as the destination at the next time. Due to the obstacles of the underwater environment, the Q-learning based path planning

strategy is adopted to plan the optimal path of AUV for data collection. At the beginning, the AUV submerge to the depth of the data collector with the largest income function. Then, the current position of AUV is set as the initial state and the data collector with the largest income function is set as the destination. Of note, the area is uniformly divided into $\mathcal{M}=n \times n$ lattices, in which each lattice corresponds to a state and its corresponding Q value is zero. Based on this, there are $\mathcal{M}$ states. Different from Section 3.1, each state in this section has four optional action, i.e., action set includes moving toward east, west, north and south. To evaluate the performance of actions, the reward function $R_A(s_l^2, a_l^2)$ is defined as

$$\mathcal{R}_A(s_l^2, a_l^2) = \begin{cases} \mathcal{R}_{A, \max}, & s_{l+1}^2 \text{ is destination} \\ -\mathcal{R}_{A, \max}, & s_{l+1}^2 \text{ is obstacle} \\ 0, & \text{otherwise} \end{cases} \quad (18)$$

where $\mathcal{R}_{A, \max}$ is a maximum positive value. In addition, s_l^2, s_{l+1}^2 and a_l^2 are the current state, the next state and the current action of AUV, respectively.

Based on this, the Q function is given as

$$Q_A(s_l^2, a_l^2) = (1 - \alpha_1)Q_A(s_l^2, a_l^2) + \alpha_1(\mathcal{R}_A(s_l^2, a_l^2) + \gamma_1 \max(Q_A(s_{l+1}^2, a_{l+1}^2))) \quad (19)$$

where α_1 is the learning rate, γ_1 is the discount factor, and a_{l+1}^2 is the next action of AUV.

Similarly, the pseudo codes of Q-learning based path planning strategy of AUV is presented in Algorithm 3. It is noted that the information value of the data collector, the effect of geographical location and environment obstacles are all considered.

According to Algorithm 3, the dynamic path planing of AUV can be obtained, i.e., the desired trajectory of AUV is acquired. Since the accurately dynamic parameters of AUV is difficult to obtain, the model-free tracking controller is developed to drive the AUV to track the desired trajectory. Considering that the PID controller is simplicity, easy implementation, excellent performance and meeting the demand of tracking, the PID tracking controller is employed in this paper for tracking. Without loss of generality, the trajectory of AUV is defined as $X_{\text{AUV}}=[x, y, z]^{\text{T}}$ and the desired trajectory of AUV is denoted as $X_r=[x_r, y_r, z_r]^{\text{T}}$. Thus, the position tracking error is computed as $e=X_r-X_{\text{AUV}}$. Define $F=[F_u, F_v, F_w]^{\text{T}}$. To ensure $X_{\text{AUV}} \rightarrow X_r$ within a limited time interval, the PID tracking controller is designed as

$$F(t) = F(t-1) + \Delta F(t-1) = F(t-1) + K_p \Delta e(t) + K_I e(t) + K_d(\Delta e(t) - \Delta e(t-1)) \quad (20)$$

where K_p is proportional gain, K_I is integral gain and K_d is derivative gain. Besides that, $\Delta e(t)=e(t)-e(t-1)$ denotes tracking velocity error.

Obviously, the PID tracking controller does not require

model information of AUV. In view of (Zhao and Guo 2017), the PID tracking controller can guarantee that the AUV can accurately track the desired trajectory by adjusting the parameters of K_p, K_I and K_d.

4 Performance analysis

Theorem 1 The potential game based optimal rigid graph is 3-connected. The average node degree of each node is 6.
Proof According to Algorithm 1, the generated graph is the minimum rigid graph. From the properties of the minimum rigid graph, it can be seen that each node in the optimal rigid graph has at least three communicating neighbor nodes for rigidity requirement. Thereby, the potential game based optimal rigid graph is 3-connected.

Clearly, the number of edges of optimal rigid graph is $\mathcal{E}=3(N+1)-6$. Referring to (Yan et al. 2018), we know that the sum of the degree of all nodes is twice the number of edges. Thus, the degree of all nodes is computed as $2(3(N+1)-6)$. Define the average degree of each node is AD, then one has $\text{AD} = \dfrac{2(3(N+1)-6)}{N+1}$. It is noted that the average degree of each node converges to 6 with the increasing of the nodes number, i.e., $\lim_{N+1 \to \infty} \text{AD}=6$. Thereby, the average degree of each node is 6.

Theorem 2 Under the condition of the minimum rigidity, the optimal rigid graph in this paper has the shortest communication link and the largest weight sum of the energy balance under the condition of maximizing algebraic characteristic.
Proof Take subarea R_i as an example. According to definition 1, we know that the number of edges of the minimum rigid graph is $3(N+1)-6$. The proposed optimal rigid graph can ensure that the network has the least communication links under the condition of ensuring connectivity and stability.

Clearly, the energy consumption of the sensor nodes is used to receive data and transmit data. According to (3), one knows that the energy consumed by transmitting data is mainly related to distance between receiver and transmitter. We assume that the number of data packets is received by each node is same, then the energy consumption problem of sensor nodes is mainly determined by the energy consumed by transmitting data.

According to Algorithm 1, the weight of energy balance is maximized with maximizing the weight of rigid matrix. Thus, the transmission energy consumption of the sensor node is minimized. Due to the transmission energy consumption is mainly related to the transmission distance, the communication link of sensor node is minimized. Clearly, this proof also suitable for other subarea.

Theorem 3 The game model $\Gamma\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}$ for optimal rigid graph is an OPG. The function $U(a):\vec{A} \rightarrow \mathbb{R}$ is an OPF of $\Gamma\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}$.
Proof The ordinal potential function is defined as $U(a_i, a_{-i})=$

$\sum_{i\in\mathcal{V}}\mathcal{U}(a_i,\,a_{-i})$ and let $\Delta\mathcal{U}_i{=}\mathcal{U}_i(a_i^b,\,a_{-i}){-}\mathcal{U}_i(a_i^a,\,a_{-i})$, where a_i^a and a_i^b are two different strategies of node i. Then, the difference between the utility functions of the two different strategies can be calculated as

$$
\begin{aligned}
\Delta U &= U\left(a_i^b, a_{-i}\right) - U\left(a_i^a, a_{-i}\right) \\
&= \sum_{i \in \mathcal{V}}(h_i(a_i^b, a_{-i})\mathrm{trace}\,(X_{(\mathcal{G}, a_i^b)}) \\
&\quad - h_i(a_i^a, a_{-i})\mathrm{trace}\,(X_{(\mathcal{G}, a_i^a)})) \\
&= [\, h_i(a_i^b, a_{-i})\mathrm{trace}\,(X_{(\mathcal{G}, a_i^b)}) \\
&\quad + \sum_{j \in \mathcal{V}, j \neq i} h_j(a_i^b, a_{-i})\mathrm{trace}\,(X_{(\mathcal{G}, a_i^b)}) \,] \\
&\quad - [\, h_i(a_i^a, a_{-i})\mathrm{trace}\,(X_{(\mathcal{G}, a_i^a)}) \\
&\quad + \sum_{j \in \mathcal{V}, j \neq i} h_j(a_i^a, a_i)\mathrm{trace}\,(X_{(\mathcal{G}, a_i^a)}) \,] \\
&= \Delta\mathcal{U}_i + \sum_{j \in \mathcal{V}, j \neq i} h_j(a_i^b, a_{-i})\mathrm{trace}\,(X_{(\mathcal{G}, a_i^b)}) \\
&\quad - \sum_{j \in \mathcal{V}, j \neq i} h_j(a_i^a, a_{-i})\mathrm{trace}\,(X_{(\mathcal{G}, a_i^a)})
\end{aligned}
\tag{21}
$$

Then, the signs of ΔU and $\Delta\mathcal{U}_i$ are analyzed as follows, i.e.,

$$
\begin{aligned}
&\mathrm{sgn}\,(\Delta U) = \\
&\mathrm{sgn}\,(\Delta\mathcal{U}_i)
\begin{cases}
= 0, h_i(a_i^b, a_{-i}) = h_i(a_i^a, a_{-i}) = 0 \\
> 0, h_i(a_i^b, a_{-i}) = 1, h_i(a_i^a, a_{-i}) = 0 \\
< 0, h_i(a_i^b, a_{-i}) = 0, h_i(a_i^a, a_{-i}) = 1
\end{cases}
\end{aligned}
\tag{22}
$$

If $h_i(a_i^b, a_{-i}){=}h_i(a_i^a, a_{-i}){=}1$ holds, then one has

$$
\mathrm{sgn}\,(\Delta U) = \mathrm{sgn}\,(\Delta\mathcal{U}_i)
\begin{cases}
= 0, a_i^b = a_i^a \\
> 0, a_i^b > a_i^a \\
< 0, a_i^b < a_i^a
\end{cases}
\tag{23}
$$

According to Definition 2 and Definition 3, one can obtained that the game model $\Gamma\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}$ for optimal rigid graph is an OPG. The function $U(a):\mathcal{A}\to\mathbb{R}$ is an OPF of $\Gamma\{\mathcal{I}, \mathcal{A}, \mathcal{U}\}$. The proof is completed.

Theorem 4 Q-learning-based dynamic routing algorithms do not have data loops in the process of data fusion.

Proof According to (12), the reward function is set to the maximum reward value $R_{\max}$ when the next state is the data collector, the reward function is set to a negative value $-R_{\max}$ when the two nodes are not communication, and the reward function is set to $-d_{i,j}$ when the nodes are communication but the next state is not the data collector. According to (13), we know that the Q value is higher when the communication link is close to the data collector. Therefore, the consistency of the transmission direction can be guaranteed.
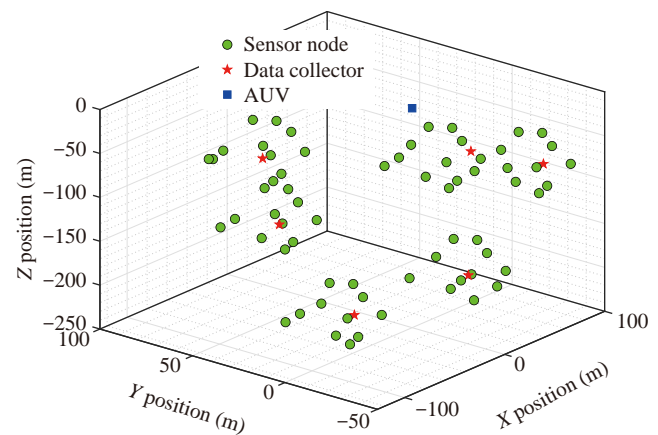
5 Simulation results

In this section, simulation results are presented to prove the effectiveness of the proposed algorithm. Without loss of generality, a monitoring area of size of 250 m×250 m×

250 m is divided into M subareas, N sensor nodes and one data collector are deployed in each subarea. In each subarea, the data is collected by sensor nodes from surrounding environment. Subsequently, sensor node sends data to the data collector by single-hop or multi-hop manner. Of note, the number of sensor nodes and subareas are mainly determined by monitoring performance requirements. The more the number of sensor nodes and subareas, the higher the accuracy of data collection, but the higher the monitoring cost. Especially, the parameters used in simulation are given in Table 1. Accordingly, the initial deployment of the network is shown in Figure 4.

Table 1 Parameters used in the simulation

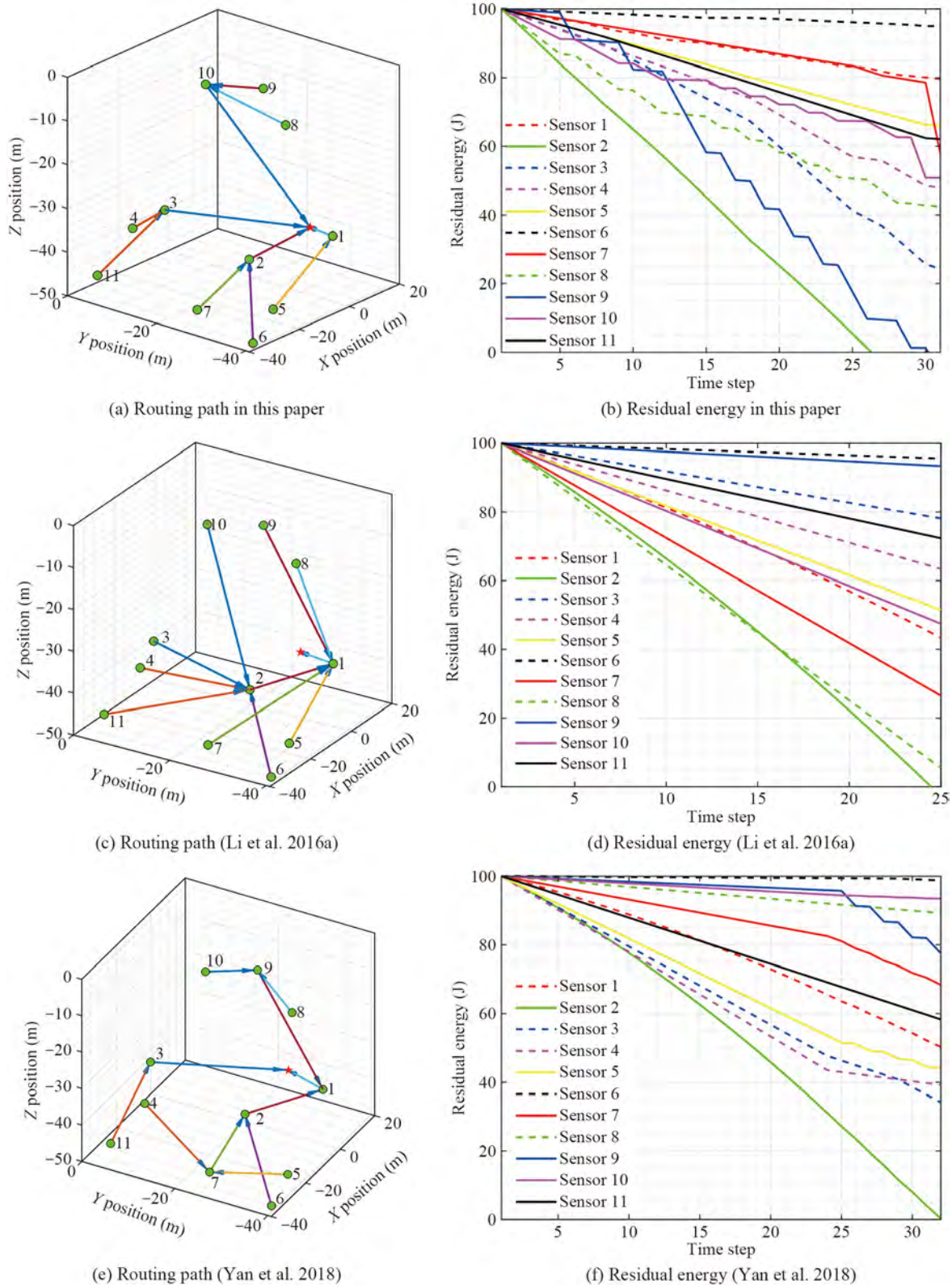| Parameter	Value
m_u	39.15
I_r	3.53
k_w	6.56
$k_v^* v \lvert v\rvert$	32.83
W	−5
c	0.12
ρ	0.1
m_v	63.96
k_u	1.6
k_r	4.54
$k_w^* w \lvert w\rvert$	29.68
Initial energy	100
ε	0.3
M	6
m_w	63.96
k_v	13.18
$k_u^* u \lvert u\rvert$	22.94
$k_r^* r \lvert r\rvert$	22.16
A	1.2
ω	0.4
N	11

Figure 4   Initial deployment of network

5.1 Simulation for routing path of sensor nodes

The results of routing path of sensor nodes are analyzed in this section. Since the routing path of each subarea is similar, only the results of arbitrary subarea S_1 are investigated, i.e., the results of other subarea are omitted in this paper.

(a) Routing path in this paper

(b) Residual energy in this paper

(c) Routing path (Li et al. 2016a)

(d) Residual energy (Li et al. 2016a)

(e) Routing path (Yan et al. 2018)

(f) Residual energy (Yan et al. 2018)

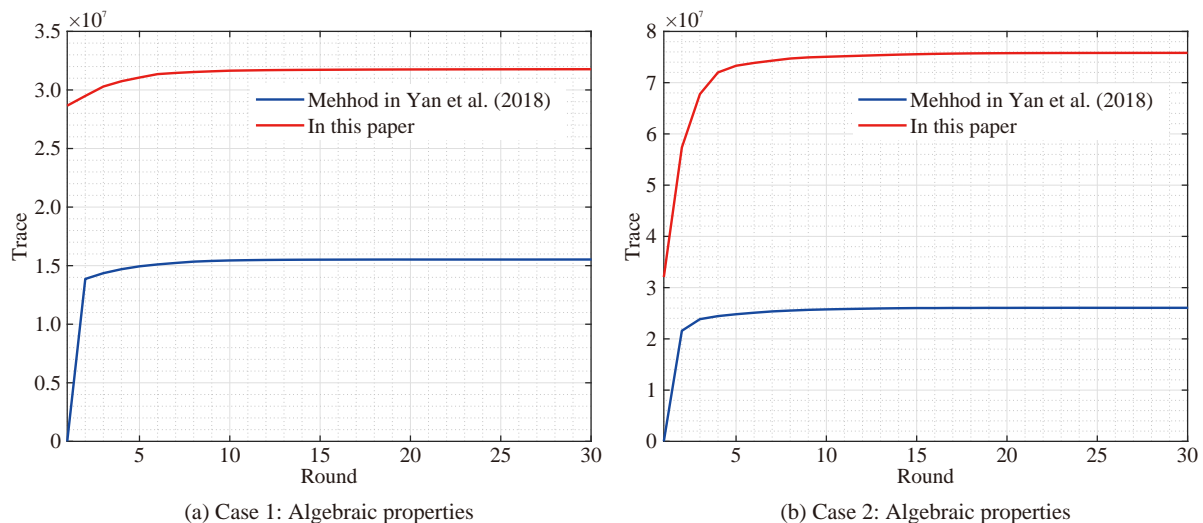**Figure 5** Results for routing path under case 1

To verify the performance of dynamic routing algorithm, two cases are considered: 1) the static underwater environment; and 2) the dynamic underwater environment. Under case 1, the nodes in network are stationary. Based on this, the routing path of sensor nodes in this paper is shown in Figure 5(a). Correspondingly, the residual energy of sensor nodes under the method in this paper is described in Figure 5(b). In Li et al. (2016a), the sensor nodes forward the data to data collector through the neighborhood nodes, which is not the optimal communication topology. Under the same assumption, the routing path and residual energy of sensor nodes are shown in Figure 5(c) and Figure 5(d), respectively. In addition, a local routing decision strategy was proposed to complete data collection from sensor nodes in Yan et al. (2018). Accordingly, the routing path and residual energy of sensor nodes are shown in Figure 5(e) and Figure 5(f), respectively. From Figures 5(a)−(d), one knows that the method in this paper can prolong the lifetime of network and the robustness of network can be guaranteed. Referring to (Shames and Summers 2015), one knows that the rigid graph with the larger eigenvalues of determinant has better algebraic properties and the stability of the network is improved accordingly. Inspired by this, the trace of $X_{(G)}$ under the methods in (Yan et al. 2018) and this paper is shown in Figure 6(a). From Figure 6(a), Figures 5(a)−(b), Figures 5(e)−(f), we can see that the methods in (Yan et al. 2018) and this paper are both effective for data collection and the stability of network is improved in this paper.

In case 2, the sensor nodes can be moved under the influence of ocean current. Based on this, the routing path and residual energy of sensor nodes under the method in this paper are shown in Figure 7(a) and Figure 7(b), respectively. Under the same assumption, the routing path and residual energy of sensor nodes under the method in Li et al. (2016a) are shown in Figure 7(c) and Figure 7(d), respectively. Correspondingly, the routing path and residu-
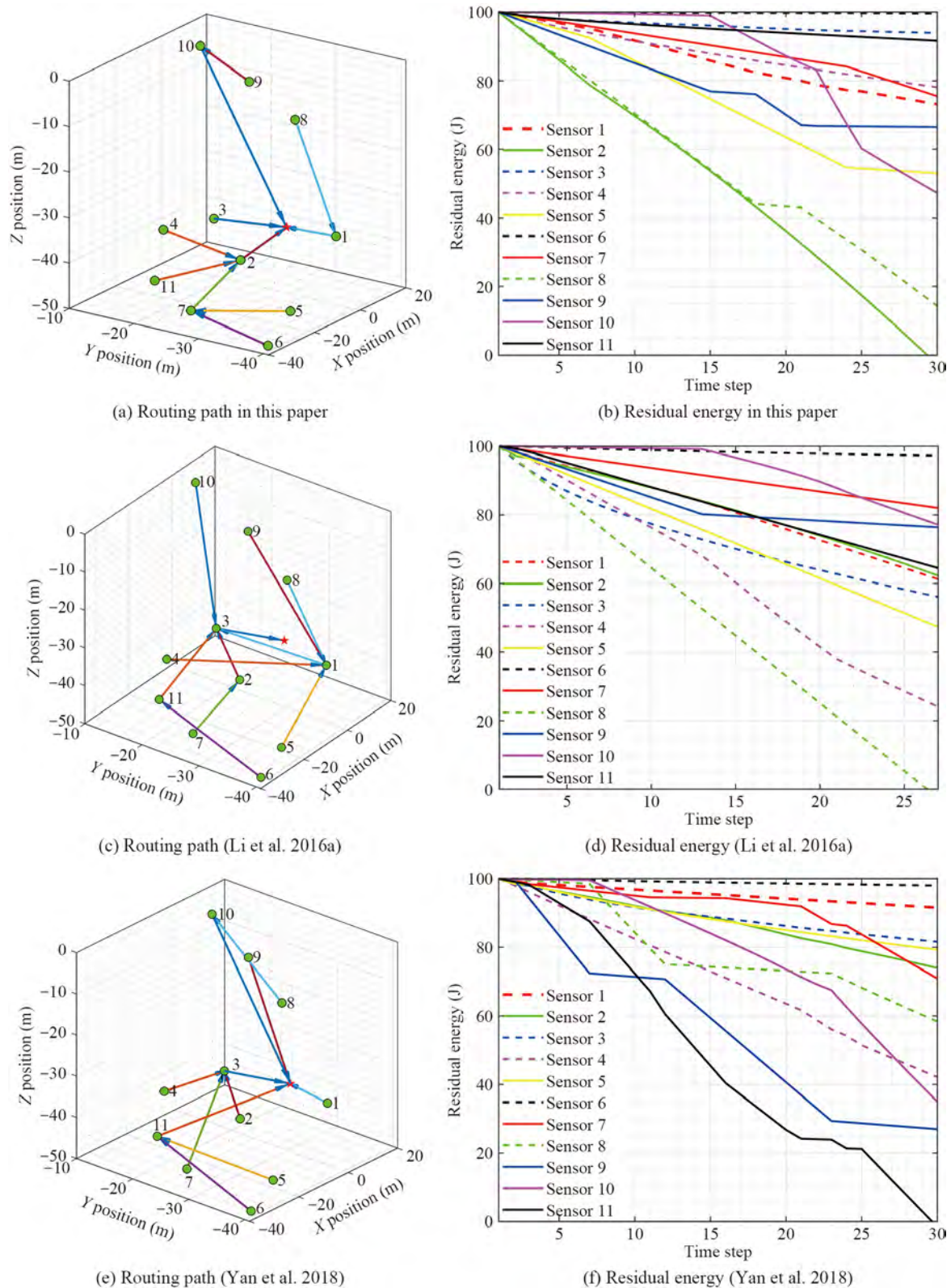
al energy of sensor nodes under the method in Yan et al. (2018) are shown in Figure 7(e) and Figure 7(f), respectively. Besides that, the trace of $X_{(G)}$ under two methods is shown in Figure 6(b). Obviously, the network lifetime can be prolonged by compare with the ones in Li et al. (2016a) and the stability of network can be improved by compare with the ones in Yan et al. (2018). Therefore, the proposed routing path method is meaningful.

5.2 Simulation for path planning and tracking of AUV

In order to verify the effective of the path planning in this paper, the VoI collected by data collectors are set as: $RC_1 = 0$, $RC_2 = 2.4$, $RC_3 = 1.8$, $RC_4 = 3.5$, $RC_5 = 3.8 + 0.3t$, and $RC_6 = 0.05$, where $t \in [0, 160]$. Based on the VoI, the AUV analyzes the VoI of data collectors and path planning through Algorithm 3. Accordingly, the path planning of AUV (i.e., desired trajectory of AUV) with the method in this paper is shown in Figure 8(a). In Han et al. (2017), the traveling salesman problem (TSP) strategy was developed to plan the trajectory of AUV. Under the same assumption, the path planning of AUV with the method in Han et al. (2017) is described in Figure 8(b). From Figures 8(a)−(b), we can seen that the path planning of AUV under the method in this paper is more appropriate comparing with the method in Han et al. (2017), since the VoI of data collectors 1 and 6 are small, they do not need to be accessed by AUV. Of note, accessing valueless data collectors can increase the offload time for the collected data to the control center on the sea surface. To verify the effectiveness of the proposed algorithm, the environmental obstacles are set in paths 2 and 3, i.e., the pink area in Figure 8(c). Accordingly, the path planning of AUV from data collector 2 to data collector 3 is shown in Figure 8(c), where the optimal path of AUV is represented by red origin. Clearly, the Q-learn-

(a) Case 1: Algebraic properties

(b) Case 2: Algebraic properties

Figure 6 Algebraic properties

(a) Routing path in this paper

(b) Residual energy in this paper

(c) Routing path (Li et al. 2016a)

(d) Residual energy (Li et al. 2016a)

(e) Routing path (Yan et al. 2018)

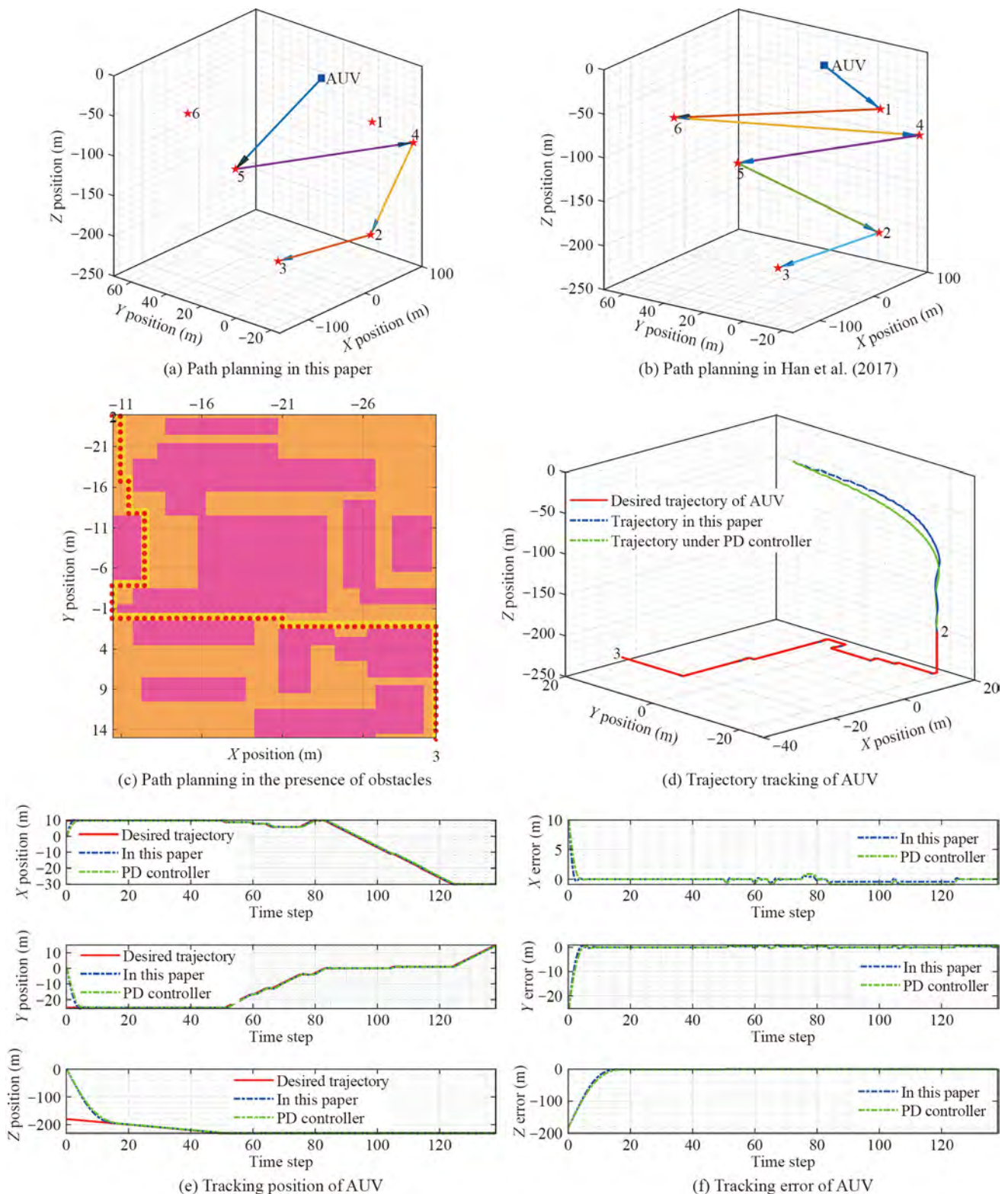(f) Residual energy (Yan et al. 2018)

Figure 7   Results for routing path under case 2

ing based AUV path planning algorithm in this paper can effectively avoid the impact of environmental obstacles.

Based on the desired trajectory of AUV, the PID tracking controller is employed for tracking. In the simulation, we only give the part of 2→3. Based on this, the tracking trajectory with the method in this paper is shown in Figure 8(d).

(a) Path planning in this paper

(b) Path planning in Han et al. (2017)

(c) Path planning in the presence of obstacles

(d) Trajectory tracking of AUV

(e) Tracking position of AUV

(f) Tracking error of AUV

Figure 8   Simulation results for path planning and trajectory tracking of AUV

Correspondingly, the tracking position and tracking error are shown in Figure 8(e) and Figure 8(f), respectively. According to Thenozhi and Yu (2014), we found that the PD controller can achieve tracking but has a steady-state error, which can be reduced by introducing an integral term. Therefore, the PID controller can achieve a zero steady-state error. Based on this, the tracking trajectory, tracking position, and tracking error under the PD tracking controller are

shown in Figures 8(d)−(f). Obviously, PID controller has better performance than PD controller.

6 Conclusion

With consideration of energy consumption, network robustness, and environmental obstacles, the energy-efficient data collection issue for IoUT has been studied in this paper. At the beginning, the potential game based optimal rigid graph strategy is proposed to balance the tradeoff between the energy consumption and the network stability. Based on optimal communication topology, the Q-learning dynamic routing algorithm for sensor nodes is developed to collect data from sensor node to data collector. Then, Q-learning based AUV path planning algorithm is presented for AUV to plan trajectory to collect data from data collector, through which the PID tracking controller is utilized to track desired trajectory. Finally, simulation results show that the proposed algorithm can ensure the energy efficient and network stability by compare with some existing works, and the PID tracking controller can track desired trajectory well. In the future, we will further expand the verification fields of the experimental platform.

References

Ali T, Jung L, Faye I (2014) End-to-end delay and energy efficient routing protocol for underwater wireless sensor networks. Wireless Personal Communications 79(1): 339-361. https://doi.org/10.1007/s11277-014-1859-z

Bhadauria D, Tekdas O, Isler V (2011) Robotic data mules for collecting data over sparse sensor fields. Journal of Field Robotics 28(3): 388-404. https://doi.org/10.1002/rob.20384

Caccia M, Veruggio G (2000) Guidance and control of a reconfigurable unmanned underwater vehicle. Control Engineering Practice 8(1): 21-37. https://doi.org/10.1016/S0967-0661(99)00125-2

Cai S, Zhu Y, Wang T, Xu G, Liu A, Liu X (2019) Data collection in underwater sensor networks based on mobile edge computing. IEEE Access 7: 65357-65367. https://doi.org/10.1109/ACCESS.2019.2918213

Cashmore M, Fox M, Long D, Magazzeni D, Ridder B (2018) Opportunistic planning in autonomous underwater missions. IEEE Transactions on Automation Science and Engineering 15(2): 519-530. https://doi.org/10.1109/TASE.2016.2636662

Chen J, Xu Y, Wu Q, Zhang Y, Chen X, Qi N (2019) Interference-aware online distributed channel selection for multicluster fanet: A potential game approach. IEEE Transactions on Vehicular

Technology 68(4): 3792-3804. https://doi.org/10.1109/TVT.2019.2902177

Dhurandher S, Khairwal S, Obaidat M, Misra S (2009) Efficient data acquisition in underwater wireless sensor ad hoc networks. IEEE Wireless Communications 16(6): 70-78. https://doi.org/10.1109/MWC.2009.5361181

Duan R, Du J, Jiang C, Ren Y (2020) Value-based hierarchical information collection for auv-enabled internet of underwater things. IEEE Internet of Things Journal 7(10): 9870-9883. https://doi.org/10.1109/JIOT.2020.2994909

Gjanci P, Petrioli C, Basagni S, Phillips C, Bölöni L, Turgut D (2018) Path finding for maximum value of information in multi-modal underwater wireless sensor networks. IEEE Transactions on Mobile Computing 17(2): 404-418. https://doi.org/10.1109/TMC.2017.2706689

Gong Z, Li C, Jiang F (2020) A machine learning-based approach for auto-detection and localization of targets in underwater acoustic array networks. IEEE Transactions on Vehicular Technology 69(12): 15857-15866. https://doi.org/10.1109/TVT.2020.3036350

Han G, Li S, Zhu C, Jiang J, Zhang W (2017) Probabilistic neighborhood-based data collection algorithms for 3D underwater acoustic sensor networks. Sensors 17(2): 1-16. https://doi.org/10.3390/s17020316

Han G, Shen S, Song H, Yang T, Zhang W (2018) A stratification-based data collection scheme in underwater acoustic sensor networks. IEEE Transactions on Vehicular Technology 67(11): 10671-10682. https://doi.org/10.1109/TVT.2018.2867021

Han G, Shen S, Wang H, Jiang J, Guizani M (2019) Prediction-based delay optimization data collection algorithm for underwater acoustic sensor networks. IEEE Transactions on Vehicular Technology 68(7): 6926-6936. https://doi.org/10.1109/TVT.2019.2914586

Jin Z, Ma Y, Su Y, Li S, Fu X (2017) A Q-learning-based delay-aware routing algorithm to extend the lifetime of underwater sensor networks. Sensors 17(7): 1-15. https://doi.org/10.3390/s17071660

Khan J, Cho H (2014) A data gathering protocol using AUV in underwater sensor networks. Proceedings of OCEANS, Taipei, 1-6. https://doi.org/10.3390/s150819331

Khan J, Cho H (2016) Data-gathering scheme using auvs in large-scale underwater sensor networks: A multihop approach. Sensors 16(10): 1-20. https://doi.org/10.3390/s16101626

Kim J, Joe H, Yu S, Lee J, Kim, M (2016) Time-delay controller design for position control of autonomous underwater vehicle under disturbances. IEEE Transactions on Industrial Electronics 63(2): 1052-1061. https://doi.org/10.1109/TIE.2015.2477270

Lee S, Park S, Choi H (2018) Potential game-based non-myopic sensor network planning for multi-target tracking. IEEE Access 6: 79245-79257. https://doi.org/10.1109/ACCESS.2018.2885027

Li C, Xu Y, Diao B, Wang Q, An Z (2016a) DBR-MAC: A depth-based routing aware mac protocol for data collection in underwater acoustic sensor networks. IEEE Sensors Journal 16(10): 3904-3913. https://doi.org/10.1109/JSEN.2016.2530815

Li G, Peng S, Wang C, Niu J, Yuan Y (2019) An energy-efficient data collection scheme using denoising autoencoder in wireless sensor networks. Tsinghua Science and Technology 24(1): 86-96. https://doi.org/10.26599/TST.2018.9010002

Li N, Martínez J, Meneses Chaus J, Eckert M (2016b) A survey on underwater acoustic sensor network routing protocols. Sensors 16(3): 1-28. https://doi.org/10.3390/s16030414

Lin C, Han G, Guizani M, Bi Y, Du J (2019) A scheme for delay-sensitive spatiotemporal routing in sdn-enabled underwater acoustic sensor networks. IEEE Transactions on Vehicular Technology 68(9): 9280-9292. https://doi.org/10.1109/TVT.2019.2931312

Lin C, Deng D, Wang S (2016) Extending the lifetime of dynamic underwater acoustic sensor networks using multi-population harmony search algorithm. IEEE Sensors Journal 16(11): 4034-4042. https://doi.org/10.1109/JSEN.2015.2440416

Liu Z, Meng X, Liu Y, Yang Y, Wang Y (2022) AUV-aided hybrid data collection scheme based on value of information for internet of underwater things. IEEE Internet of Things Journal 9(9): 6944-6955. https://doi.org/10.1109/JIOT.2021.3115800

Luo X, Li X, Wang J, Guan X (2018) Potential-game based optimally rigid topology control in wireless sensor networks. IEEE Access 6: 16599-16609. https://doi.org/10.1109/ACCESS.2018.2814079

Luo X, Yan Y, Li S, Guan X (2013) Topology control based on optimally rigid graph in wireless sensor networks. Computer Networks 57(4): 1037-1047. https://doi.org/10.1016/j.comnet.2012.12.002

McMahon J, Plaku E (2017) Autonomous data collection with limited time for underwater vehicles. IEEE Robotics and Automation Letters 2(1): 112-119. https://doi.org/10.1109/LRA.2016.2553175

Ren J, Zhang Y, Zhang K, Liu A, Chen J, Shen X (2016) Lifetime and energy hole evolution analysis in data-gathering wireless sensor networks. IEEE Transactions on Industrial Informatics 12(2): 788-800. https://doi.org/10.1109/TII.2015.2411231

Sehgal A, David C, Schönwälder J (2011) Energy consumption analysis of underwater acoustic sensor networks. Proceedings of OCEANS, Waikoloa, USA, 1-6. https://doi.org/10.23919/OCEANS.2011.6107287

Shames I, Summers T (2015) Rigid network design via submodular set function optimization. IEEE Transactions on Network Science and Engineering 2(3): 84-96. https://doi.org/10.1109/TNSE.2015.2480247

Su R, Zhang D, Li C, Gong Z, Venkatesan R, Jiang F (2019) Localization and data collection in auv-aided underwater sensor networks: Challenges and opportunities. IEEE Network 33(6): 86-93. https://doi.org/10.1109/MNET.2019.1800425

Thenozhi S, Yu W (2014) Stability analysis of active vibration control of building structures using PD/PID control. Engineering Structures 81: 208-218. https://doi.org/10.1016/j.engstruct.2014.09.042

Wang Z, Song H, Watkins D, Ong K, Xue P, Yang Q, Shi X (2015) Cyber-physical systems for water sustainability: challenges and opportunities. IEEE Communications Magazine 53(5): 216-222. https://doi.org/10.1109/MCOM.2015.7105668

Wei B, Luo Y, Jin Z, Wei J, Su Y (2013) ES-VBF: an energy saving routing protocol. Proceedings of the 2012 International Conference on Information Technology and Software Engineering, Beijing, 87-97. https://doi.org/10.1007/978-3-642-34528-9-10

Wu J, Sun X, Wu J, Han G (2021) Routing strategy of reducing energy consumption for underwater data collection. Intelligent and Converged Networks 2(3): 163-176. https://doi.org/10.23919/ICN.2021.0012

Xia N, Ou Y, Wang S, Zheng R, Du H, Xu C (2017) Localizability judgment in uwsns based on skeleton and rigidity theory. IEEE Transactions on Mobile Computing 16(4): 980-989. https://doi.org/10.1109/TMC.2016.2586051

Xie P, Cui J, Lao L (2006) VBF: Vector-based forwarding protocol for underwater sensor networks. Proceeding of International Conference on Research in Networking, Springer, Berlin, Heidelberg, 1216-1221. https://doi.org/10.1007/11753810-111

Yan J, Chen C, Luo X, Yang X, Hua C, Guan X (2016) Distributed formation control for teleoperating cyber-physical system under time delay and actuator saturation constrains. Information Sciences 370: 680-694. https://doi.org/10.1016/j.ins.2016.02.019

Yan J, Zhao H, Meng Y, Guan X (2021a) Localization in underwater sensor networks. ser. Wireless Networks, Singapore Springer. https://doi.org/10.1007/978-981-16-4831-1

Yan J, Yang X, Zhao H, Luo X, Guan X (2021b) Autonomous underwater vehicles: Localization, tracking, and formation. ser. Cognitive Intelligence and Robotics, Singapore Springer. https://doi.org/10.1007/978-981-16-6096-2

Yan J, Yang X, Luo X, Chen C (2018) Energy-efficient data collection over auv-assisted underwater acoustic sensor network. IEEE Systems Journal 12(4): 3519-3530. https://doi.org/10.1109/JSYST.2017.2789283

Yan J, Zhao H, Luo X, Wang Y, Chen C, Guan X (2020a) Asynchronous localization of underwater target using consensus-based unscented kalman filtering. IEEE Journal of Oceanic Engineering 45(4): 1466-1481. https://doi.org/10.1109/JOE.2019.2923826

Yan J, Zhao H, Pu B, Luo X, Chen C, Guan X (2020b) Energy-efficient target tracking with uasns: A consensus-based bayesian approach. IEEE Transactions on Automation Science and Engineering 17(3): 1361-1375. https://doi.org/10.1109/TASE.2019.2950702

Yu H, Yao N, Liu J (2015) An adaptive routing protocol in underwater sparse acoustic sensor networks. Ad Hoc Networks 34: 121-143. https://doi.org/10.1016/j.adhoc.2014.09.016

Zhang P, de Queiroz M, Cai X (2015) Three-dimensional dynamic formation control of multi-agent systems using rigid graphs. Journal of Dynamic Systems, Measurement, and Control 137(11): 1-7. https://doi.org/10.1115/1.4030973

Zhao C, Guo L (2017) PID controller design for second order nonlinear uncertain systems. Science China Information Sciences 60(2): 1-13. https://doi.org/10.1007/s11432-016-0879-3