A Federated Deep Reinforcement Learning-Based Trust Model in Underwater Acoustic Sensor Networks

Yu He, Guangjie Han, *Fellow, IEEE,* Aohan Li, *Member, IEEE,* Tarik Taleb, *Senior Member, IEEE,* Chenyang Wang, *Member, IEEE,* and Hao Yu

***Abstract*—Underwater acoustic sensor networks (UASNs) have been widely deployed in many areas, such as marine ranching, naval applications, and marine disaster warning systems. The security of UASNs, particularly insider threats, is of growing concern. Internal attacks carried out via compromised normal nodes are more damaging and stealthy than external attacks, such as signal stealing, data decryption, and identity forgery. As a security mechanism for internal threat detection based on interaction data, trust models have proven to enhance the security of UASNs. However, traditional trust models lack sufficient scalability when faced with movable underwater devices, heterogeneous network environments, and variable attack patterns. Therefore, in this paper, a novel trust model based on federated deep reinforcement learning is proposed for UASNs. First, the evidence acquisition mechanism, including communication, energy, and data evidence, is improved based on existing ones to better accommodate the topological dynamics of UASNs. Second, acquired trust evidence is fed into the corresponding deep reinforcement learning-based local trust model to accomplish trust prediction and model training. Finally, a federated learning-based update method periodically aggregates and updates the parameters of the local models. The experimental results prove that the proposed scheme exhibits satisfactory performance in terms of improving trust prediction accuracy and energy efficiency.**

***Index Terms*—Underwater acoustic sensor networks, trust model, deep reinforcement learning, and federated learning.**

I. Introduction

UNDERWATER acoustic sensor networks (UASNs) are innovative paradigms widely applied in underwater environment monitoring, disaster warning systems, military defense, and other underwater-based scenarios [1]–[3]. The open nature of the underwater environment makes it easy for adversaries to infiltrate devices in UASNs, compromising and transforming them into malicious nodes lurking in the network [4]. Depending on the attack program implanted, these malicious nodes can execute attacks on different frequencies, modes, and targets. The legitimacy of such compromised nodes makes it difficult for traditional authentication and data encryption methods to counter their attacks. The trust model, a mechanism for predicting the potential for future interactions based on historical interactions, has been proven to be effective against such internal threats. Typically, trust models in UASNs utilize historical communication behavior between underwater nodes or differences in each other's attributes to predict the trustworthiness of the target node [5]. Furthermore, trustworthiness is used to ensure the reliability of critical devices during the operation of UASNs, such as selecting data forwarding nodes, electing cluster head nodes, and filtering data fusion targets.

Although several studies have contributed to the development of trust models in UASNs, there are several issues that have not yet been effectively addressed. For example, with the rapid development of the underwater robotics industry, new devices, such as autonomous underwater vehicles (AUVs), underwater gliders, and wave gliders (shown in Fig. 1), are gradually increasing in underwater networks. Homogeneous UASNs comprising sensor nodes are gradually evolving into heterogeneous networks entailing different types of devices with distinctive capabilities in communication, computing, and mobility. The impact of the heterogeneity of UASNs on trust management reflects primarily two aspects [6]–[10]:

1) Device heterogeneity within the local area. Conventional trust models typically possess fixed parameters and thresholds. In the local network context, when there is a significant difference between the inherent attributes of trust evaluation objects, traditional trust models lack the necessary adaptability, resulting in a loss of evaluation precision. As one of the indicators of trust evaluation, the communication success rate is typically compared to a predetermined threshold to determine the target's communication credibility. There is a substantial difference between the communication capabilities of underwater sensor nodes and AUVs; consequently, the credibility thresholds for evaluating the communication capabilities of the two should also be distinct. In heterogeneous UASN scenarios, it is difficult to set appropriate thresholds and model parameters for different evaluation objects using conventional trust models, which lack adaptive capabilities. Deep reinforcement learning (DRL) is poised to revolutionize the field of artificial intelligence and represents a step toward the development of autonomous systems with a higher-level understanding of their sur-

Yu He, Guangjie Han are with Changzhou Key Laboratory of Internet of Things Technology for Intelligent River and Lake, Hohai University, Changzhou 213022, China. (e-mail: heyuhhu2018@outlook.com, hanguangjie@gmail.com).

Aohan Li is with the Graduate School of Informatics and Engineering, The University of Electro-Communications, Tokyo, Japan. (e-mail: aohanli@ieee.org).

Tarik Taleb, Hao Yu are with the Center for Wireless Communications, Oulu University, Oulu 90570, Finland. (e-mail: tarik.taleb@oulu.fi, hao.yu@oulu.fi).

Chenyang Wang is with College of Intelligence and Computing, Tianjin University, Tianjin, 300072 China (e-mail: chenyangwang@tju.edu.cn).

roundings [11]. Consequently, this study proposes a DRL-based trust modeling method that employs the robust parameter learning capability of DRL to enhance the adaptability of the trust model to heterogeneous UASNs.

2) Network heterogeneity across regions. The UASNs are gradually evolving toward complex functions and multi-regional integration, resulting in heterogeneity in tasks, environments, and equipment across sub-networks. The traditional trust model has two issues when confronted with the trust evaluation requirements of cross-regional devices: insufficient accuracy of local model evaluation and high cross-regional cost of interaction experience. Federated learning (FL) enables cross-regional model training while preserving the localization of training data [12]. This study therefore incorporates FL into the trust model update process and proposes an FL-based cross-region trust update method. This method achieves effective control of the interregional transmission cost of the interactive experience while utilizing the interactive experience across multiple subnetworks.

Therefore, in this study, we propose a trust model based on joint DRL and FL for heterogeneous UASNs. On the premise of enhancing traditional evidence generation methods to adapt to heterogeneous scenarios, this work employs DRL for trust modeling to improve the adaptability to complex underwater environments. Additionally, the distributed training and centralized update of the model are realized via FL, which further improves the evaluation accuracy of the trust model in the context of spatiotemporal changes. The main contributions of this study are summarized as follows.

1) A more general adversary model is proposed, which incorporates the comprehensive influence of active attack, underwater acoustic communication, network topology, and other factors on trust management in the underwater environment.
2) The traditional method for quantifying trust evidence is enhanced, the impact of malicious attacks on the evidence quantification process is mitigated, and the evaluation precision of the trust model is indirectly enhanced.
3) A DRL-based trust modeling method is proposed in order to increase the adaptability of the trust model to heterogeneous UASNs. This method mitigates the cold start problem of the trust model via the model pre-training mechanism and achieves the effective training of the trust model via the ingenious design of parameters such as the state, action, and reward function in DRL.
4) A FL-based cross-region trust update architecture is proposed, which enhances the trust model's adaptability and scalability by comprehensively utilizing local experience and controlling the transmission cost of interactive experience.

The remainder of this paper is organized as follows. In Sec. II, an overview of previous related literature on trust models is provided. In Sec. III, the system model and assumptions are introduced. Then, detailed descriptions of the proposed scheme and simulation results are provided in Secs. IV and V, respectively. Finally, the conclusions of this study are drawn in Sec. VI.

II. Related work

UASNs are gradually gaining increasing attention from researchers with the recent developments in hydroacoustic communication and underwater networking technologies. Several studies have focused on trust security in UASNs and have made key contributions to the development of underwater trust models. The following section provides a review of relevant research related to trust security in UASNs, and proposes our solutions to address the current issues.

Trust models have been widely used in different scenarios, such as terrestrial wireless sensor networks [13]–[15], social networks [16], [17], vehicle networking [18], [19], and cloud computing [20], [21]. Despite the rapid development of trust models, there are still several issues that need to be addressed when using trust models in UASNs. Han et al. [22] first studied trust management in UASNs and proposed an attack-resistant trust model based on multidimensional trust metrics (ARTMM). The ARTMM proposes three types of trust assessment methods, namely, link trust, data trust, and node trust, with the corresponding trust update mechanisms. The calculation process considers packet loss rate, link utilization, packet variability, node energy, and other metrics. However, the details of the scheme are subjective in terms of variables and weights, which makes it less scalable.

Bolster et al. [23] proposed a multi-metric trust management framework based on grey theory for UASNs to cope with the limitations of single-metric trust prediction. The scheme considers a variety of metrics, such as transmitted and received throughput, delay, received signal strength, transmitted power, and packet loss rate. Additionally, the grey relational analysis was utilized to normalize and combine the disparate metrics into a grey relational coefficient, which is ultimately used as the basis for trust judgments. Despite the advantages of this scheme over probabilistic methods, the weighting means adopted in the metric synthesis are still subjective. As a result, it is difficult to adapt the scheme to complex underwater applications.

Bolster et al. [24] proposed a machine learning-based approach to generate metric weight vectors. The study employed two types of metrics: communication metrics, which include delay, transmitted power, throughput, offered load, and packet loss rate; and physical metrics, which consider the variation in the distance between nodes, the deviation in the direction between nodes, and the speed of nodes. To achieve a more flexible and adaptable metric weight vector than subjective configuration approaches, a random forest regression model was trained, improving the accuracy of detecting misbehaviors.

Jiang et al. proposed cloud theory-based trust models for UASNs in [25] and [26] to bolster the adaptability to underwater characteristics, such as unreliable acoustic channels. Cloud theory is a method that effectively characterizes the ambiguity and randomness of trust, and it was used in both studies to calculate the trust values of nodes. First, three features – mean, entropy, and hyper entropy – of the cloud model are calculated for each type of trust evidence. Subsequently, the same feature

for varying trust evidence is weighted and summed. Finally, direct and recommendation features are combined to obtain a feature vector, which is the final representation of the trust prediction. The core of the solutions lies in the mapping of model features based on the backward cloud generator; however, the interaction data in UASNs tend to be sparse, leading to large deviations in the obtained model features, thereby limiting the accuracy of trust prediction.

Du et al. [27] proposed a trust cloud migration scheme to address the uneven energy consumption caused by frequent updates on the trust cloud model. The proposed trust cloud migration scheme is suitable for multi-hop and homogeneous UASNs. The core of data migration lies in determining the migration destination node. To address the problem, the scheme designed a destination node determination algorithm based on simulated annealing. To be more specific, the algorithm selects the destination cluster based on the average residual energy of the node clusters and then utilizes the node density accessibility to determine the migration destination node. Accordingly, their scheme effectively mitigates the energy imbalance caused by high-density trust updates; however, it assumes that the sink node knows the remaining energy of all clusters, which is not easily satisfied.

Furthermore, Du et al. proposed two customized trust models for UASNs called ITrust [28] and LTrust [29]. ITrust focuses on solving the trust instability caused by the noise in underwater environments. The solution quantifies the effect of environmental noise as a metric called environment trust and then combines it with communication trust, data trust, and energy trust as attributes of the sample, ultimately using the isolated forest algorithm in ML to classify good and bad behavior. LTrust is concerned with the perturbation of trust relationships caused by network topological variability. Accordingly, the scheme quantifies the topological relationships between neighboring nodes as node importance, which is embedded into node trust. Consequently, the trust dataset is composed of four attributes: node trust, communication trust, environment trust, and recommendation trust. The final trust assessment of node behavior is realized using the ML algorithm – LSTM. Both ITrust and LTrust adopt supervised learning for the final trust classification. However, the underwater environment is complex, resulting in trust prediction models based on specific training sets that tend to have poor generalization capabilities. The problem was also discussed in our previous work [30].

In our previous work TUMRL [31], we introduced reinforcement learning to dynamically update evidence weights by considering the learning ability of reinforcement learning for interactive environments. In TUMRL, the weight assignments of each type of evidence are defined as states, and the value domains to which the evidence belongs are defined as actions. When performing trust evaluation, the trustor first maps the trustee's actions to the evidence, that is, acquiring the action. The trustor then executes a state transfer based on the acquired action to modify the weight assignment for trust evidence. The Q-learning algorithm continuously optimizes the process to ensure that the evidence weights selected by the trustor continually adapt to a variety of attack modes and environments.

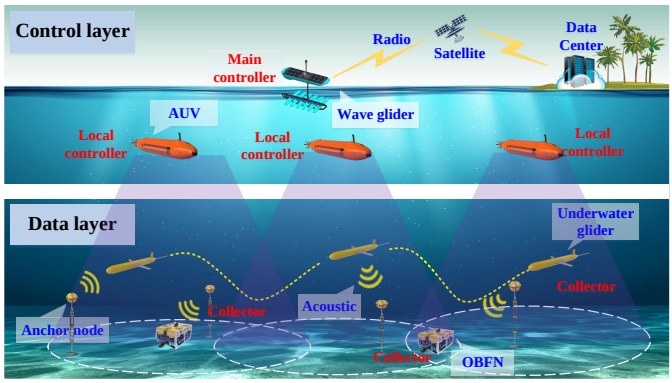


Fig. 1: Structure of a typical heterogeneous underwater acoustic sensor network.

The TUMRL scheme leverages reinforcement learning technology to enhance performance in the face of variable attack patterns and dynamic underwater environments. Nonetheless, it overlooks the impact of heterogeneity within the networks, resulting in poor scalability when dealing with large-scale, heterogeneous UASNs. Building upon this research, this paper further considers the combined influence of equipment heterogeneity within sub-networks and environmental heterogeneity between sub-networks, and introduces the cutting-edge technology of DRL to enhance the adaptive ability of the trust model.

This paper also introduces FL for global trust updating to increase the capabilities of cross-domain trust management [32]. Numerous studies have investigated the security challenges provided by the aggregation of local model parameters in FL, with some adopting a trust mechanism to check the reliability of model parameters from local clients [33]–[36]. The scheme described in [33] utilized blockchain-based trust management to maintain the credibility of participating devices in order to guarantee the integrity of model contributions. Bugshan et al. [34] introduced differential privacy and encoding-based sharing techniques to ensure the reliability of local model generation and sharing. To combat the cold start problem in recommendation systems, Wahab et al. [35] proposed a trust-based FL approach. The trust management in the FL parameter aggregation technique differs from the study presented in this article. This article focuses on security during the data collection phase, as opposed to the subsequent training phase.

III. System Model and Assumption

A. Network Model

As illustrated in Fig. 1, this analysis considers a decentralized trust management architecture for heterogeneous underwater acoustic sensor networks. The entire network is logically divided into a **control layer** and a **data layer** with three types of entities: **main controller**, **local controller**, and **collector**.

The control layer comprises a wave glider on the surface and multiple AUVs. As the sole main controller, the wave glider is responsible for updating and scheduling the trust model while receiving commands from the terrestrial data

center via a satellite relay to adjust the tasks of the network. Each AUV, as the local controller, receives commands from the main controller, and these commands are used to manage trust modeling for data layer devices in the corresponding area. It is assumed that both the main and local controllers comprise sufficient computing, communication, and storage capabilities to support the trust management system.

The data layer comprises various heterogeneous underwater devices, including underwater gliders, underwater anchor nodes, and ocean bottom flight nodes (OBFNs), collectively, they are referred to as collectors, primarily because they collect interaction information about each other while collaboratively performing network tasks. The data layer chooses these three types of components because they represent underwater equipment with different mobilities. Underwater gliders represent highly mobile collectors that can move between areas covered by different local controllers. The OBFNs stand for the collectors with limited mobility that can only move locally rather than across regions. Underwater anchor nodes represent stationary collectors. First, the interaction information of the data layer is processed into trust evidence and passed to the local controller. Second, the local controller models the trust model based on the trust evidence. Ultimately, the main controller receives the model parameters of different local controllers for trust updates and feeds back the latest model parameters.

B. Adversary Model

Owing to the openness of the network, some devices may be compromised to become malicious nodes and launch internal attacks, such as packet dropping, denial of service (DoS), and data tampering. Here, it is assumed that only data layer devices will be compromised as malicious nodes, primarily due to the fact that control layer devices are vastly superior to data layer devices and are located close to the water surface for regular inspection. In addition, it is assumed that the parameter transfer between the main controller and the local controllers is entirely credible, meaning that attacks on the FL process, such as data poisoning attacks, model poisoning attacks, and free-riding attacks [37], are not taken into account. Considering that anomalies in network functionality caused by active attacks can be captured by security techniques, such as trust models, we assume that a malicious node launches an attack at each time slot based on a certain probability to acquire long-term benefits. Additionally, given the instability of hydroacoustic communication, and the transient anomalies of the sensors themselves, we assume that there is some probability for the normal nodes to perform malicious actions. In summary, all devices in the network have a probability of performing normal behavior, called **absolute trustworthiness**, except that the absolute trustworthiness of a malicious node is much less than that of a normal node. Therefore, the goal of this study is to make the **predicted trustworthiness** output from the trust model as close to absolute trustworthiness as possible.

IV. Trust Model based on Federated Deep Reinforcement Learning

In this section, the proposed scheme FedTM is described in detail. Figure 2 depicts the workflow of FedTM, which includes three modules: 1) evidence generation, 2) trust modeling, and 3) model update. Each collector of the data layer is equipped with an evidence generation module to quantify the performance of collectors into specific evidence in terms of communication, energy consumption, and data discrepancy. Following that, the evidence is input into the local controller equipped with the trust modeling module for training in DRL. Ultimately, the model parameters of the policy networks from different local controllers are input into the update module equipped on the main controller, and the main controller feeds back the updated parameters to each local controller following the FL-based parameter updating. The above steps are repeated iteratively; that is, the trust model can accurately maintain the evaluation of the network performance.

A. Trust Evidence Generation

Based on our previous work [30], although UASNs face diverse attacks, the consequences of these attacks are primarily reflected in three aspects: communication failure, increased energy consumption, and data packet tampering. Additionally, these abnormalities may be affected by the instability of underwater acoustic communication, which often has characteristics of the multipath effect, time-varying effect, narrow usable bandwidth, serious signal attenuation, etc. Therefore, three types of trust evidence, namely communication evidence, energy evidence, and data evidence, are defined to demonstrate the impact of malicious attacks and unstable underwater acoustic channels.

1) Communication Evidence ($\mathcal{C}$): Communication evidence is typically expressed as $\mathcal{C} = \frac{s}{s+f}$, where s and f represent the number of successful and failed communications, respectively. However, this definition fails to address the problem caused by sparse evidence. For example, when $s = 1$ and $f = 0$, $\mathcal{C}$ is equal to 1. Even if the authentic $\mathcal{C}$ equals 0.01, the above situation may occur because the only monitored communication is successful. This leads to an obvious difference between the predicted $\mathcal{C}$ $(= 1)$ and the authentic $\mathcal{C}$ $(= 0.01)$. To address the above problems, we adopt beta distribution $Beta\left(x; \alpha, \beta\right)$ as a prior assumption for $\mathcal{C}$. The probability density function (PDF) of the beta distribution can be expressed as:

$$f\left(x; \alpha, \beta\right) = \frac{\Gamma\left(\alpha+\beta\right)}{\Gamma\left(\alpha\right)\Gamma\left(\beta\right)} x^{\alpha-1}(1-x)^{\beta-1}, \tag{1}$$

where Γ represents the gamma function and $\Gamma\left(n\right) = (n-1)!$ for any positive integer n.

Figure 3 displays the PDF of the beta distribution for different values of α and β, and the abscissa x corresponds to the unknown $\mathcal{C}$. Here, $\alpha = \beta = k$ $(k > 1)$, and the PDF value of $x = 0.5$ increases with k, that is, the probability of $\mathcal{C} = 0.5$ is rising. Hence, the initial prior assumption is set as $Beta\left(x; \alpha = k, \beta = k\right)$ $(k > 1)$. Subsequently, the parameters

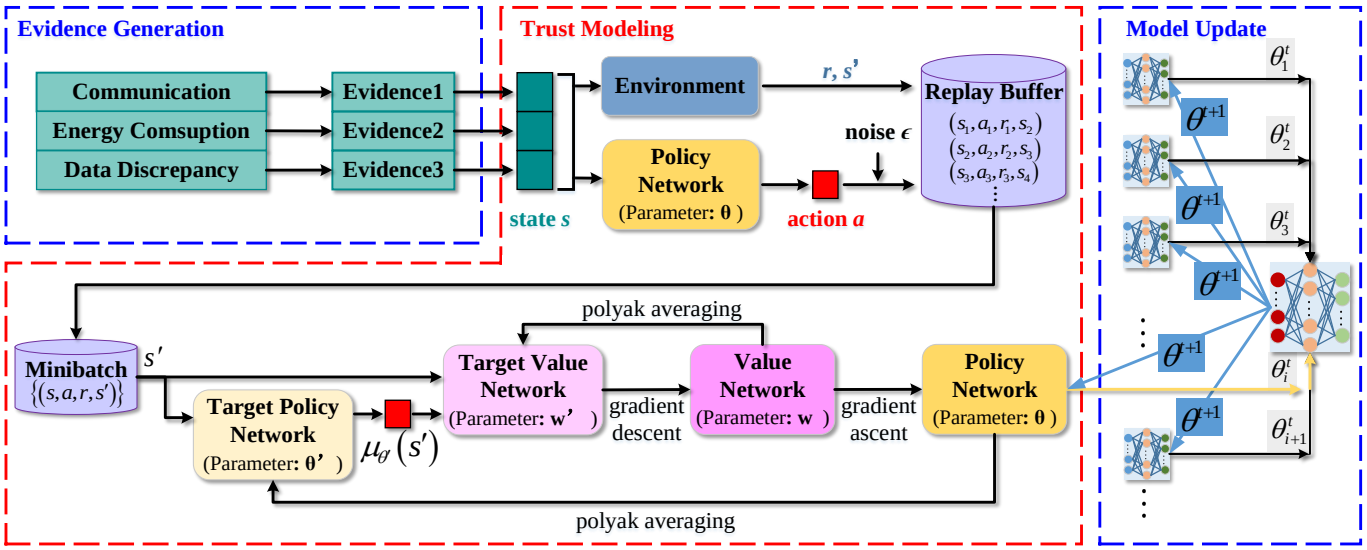


Fig. 2: Workflow of FedTM.

are updated as $\alpha_t = \alpha_{t-1} + s$, $\beta_t = \beta_{t-1} + f$. Finally, the communication evidence at time t is calculated as

$$\mathcal{C}_t = \mathbb{E}\left(Beta\left(x; \alpha_t, \beta_t\right)\right) = \frac{\alpha_t}{\alpha_t + \beta_t}. \tag{2}$$

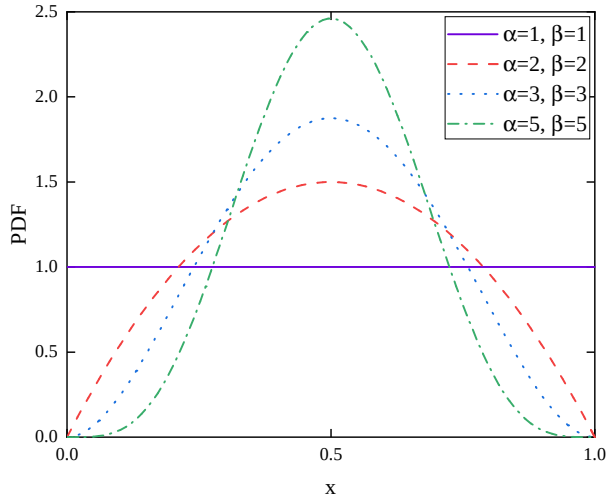


Fig. 3: Probability density function of the beta distribution with different parameters.

2) Energy Evidence ($\mathcal{E}$): Energy evidence $\mathcal{E}$ characterizes the credibility of the trustee in terms of energy, which is positively correlated with its remaining energy and negatively correlated with its abnormal degree of energy consumption. It should be noted that due to the heterogeneity of energy consumption rate levels between underwater equipment, the abnormal degree should reflect the difference in the changing trend of energy consumption rate rather than the levels. Cosine similarity meets this requirement and is therefore chosen to measure the differences in the changing trends of energy consumption rates between two nodes. Assume that E_0 and E_t represent the initial energy and current remaining energy of the trustee, respectively; then, $\mathcal{E} \propto \frac{E_t}{E_0}$. Furthermore, the sequences of energy consumption rates of the trustor and the trustee are assumed to be $\boldsymbol{x_t} = [\upsilon_{t-(m-1)}, \upsilon_{t-(m-1)}, \cdots, \upsilon_{t-1}, \upsilon_t]$ and $\boldsymbol{y_t} = [\varphi_{t-(m-1)}, \varphi_{t-(m-1)}, \cdots, \varphi_{t-1}, \varphi_t]$, respectively, where both υ_t and φ_t represent the energy consumption rate of time t and m indicates the length of the time window used for recording. Following that, the standardized vectors, $\boldsymbol{x_t^*}$ and $\boldsymbol{y_t^*}$, are obtained by independently performing the Z-score standardization on each element of the vector $\boldsymbol{x_t}$ and vector $\boldsymbol{y_t}$:

$$z^* = \frac{z - \mu}{\sigma}, \tag{3}$$

where z and z^* indicate the original element and corresponding normalized element in vectors $\boldsymbol{x_t}$ and $\boldsymbol{y_t}$, and μ and σ symbolize the mean and standard deviation of the elements in the corresponding vector. Subsequently, the cosine similarity is used to measure the difference between vectors $\boldsymbol{x_t^*}$ and $\boldsymbol{y_t^*}$:

$$\cos\left(\boldsymbol{x_t^*}, \boldsymbol{y_t^*}\right) = \frac{\sum_{i=0}^{m-1} \upsilon_{t-i}\varphi_{t-i}}{\sqrt{\sum_{i=0}^{m-1} \upsilon_{t-i}^2}\sqrt{\sum_{i=0}^{m-1} \varphi_{t-i}^2}}, \tag{4}$$

where $\cos\left(\boldsymbol{x_t^*}, \boldsymbol{y_t^*}\right) \in [0, 1]$.

The greater the value of $\cos\left(\boldsymbol{x_t^*}, \boldsymbol{y_t^*}\right)$, the closer the energy consumption rate of the trustor and trustee, that is, the lower the abnormal degree of the trustee's energy consumption rate. Thus, $\mathcal{E} \propto \cos\left(\boldsymbol{x_t^*}, \boldsymbol{y_t^*}\right)$. In summary, energy evidence is defined as

$$\mathcal{E}_t = \frac{E_t}{E_0} \cos\left(\boldsymbol{x_t^*}, \boldsymbol{y_t^*}\right). \tag{5}$$

3) Data Evidence ($\mathcal{D}$): Since the nodes tend to employ multi-hop forwarding to deliver messages, the data between nodes in adjacent underwater areas have spatiotemporal correlations. Therefore, the similarity between the data of the trustee and its neighbors is used to characterize data evidence $\mathcal{D}$, which is improved from our previous work [30]. Our previous

work ignored the influence of outliers in the neighbor data. Therefore, the *Boxplot* method is used to remove the outliers in the neighbor data before obtaining the data evidence.

Figure 4 illustrates how the *Boxplot* method filters the data. The data, such as marine temperature, pressure, salinity, etc., are obtained from the neighbors of the trustee, and sorted in the dark area in the middle of the figure. In the *Boxplot* method, data smaller than the lower limit $(\mathrm{Q1}-1.5*\mathrm{IQR})$ and larger than the upper limit $(\mathrm{Q3}+1.5*\mathrm{IQR})$, are regarded as outliers, where Q1 and Q3 represent the first quartile and the third quartile, respectively, while $\mathrm{IQR}=|\mathrm{Q1}-\mathrm{Q3}|$. Let $\{v_1, v_2, \cdots, v_m\}$ denote the filtered data, and x represent the same type of data from the trustee at time t. Accordingly, data evidence of the trustee at time t is defined as Eq. 6.

$$\mathcal{D}_t = \begin{cases} 2\left(1-\int_{-\infty}^{x} f(v)\,dv\right) & x \geq \mu \\ 2\int_{-\infty}^{x} f(v)\,dv & x < \mu \end{cases}, \tag{6}$$

where $f(v)=\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(v-\mu)^2}{2\sigma^2}\right)$ and μ and σ symbolize the mean and standard deviation of the filtered data, respectively.

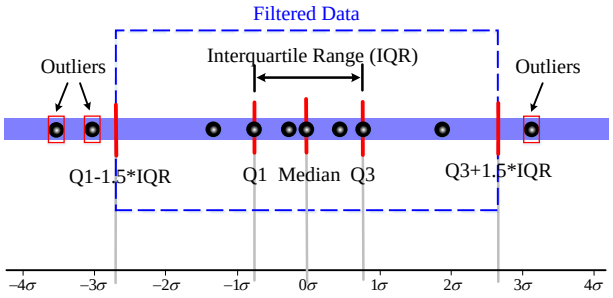


Fig. 4: Different parts of a boxplot.

B. Trust Modeling Based on Deep Reinforcement Learning

Upon obtaining evidence, traditional schemes usually generate trust models via weighted summation of evidence or ML classification. However, traditional schemes find it difficult to adapt to the dynamically changing underwater environments and network topologies owing to the lack of prior knowledge and the sparseness of trust evidence. The DRL method adopted in this scheme can continuously update the model parameters by using the interaction between the agent and the environment to obtain a strategy that adapts to the dynamic environment. This section introduces the trust modeling process based on DRL. Following network initialization, the main controller implements model pre-training based on a virtual interactive environment owing to the lack of interaction experience between network entities. Subsequently, the pre-trained model parameters are delivered to the local controllers as the initialization parameters of their local models, which speeds up the convergence. Each local controller further trains the model based on actual interactions between collectors in their respective regions.

The reinforcement learning parameters used in the proposed scheme are presented below:

1) **Agent & Interactive Environment**: In the proposed scheme, the agent is not a specific entity, instead it is the trustor in each trust evaluation process. The entities around the trustor, including the trustee and neighbors, represent the interactive environment. Based on the current **state** $\boldsymbol{s}$, the agent uses the **policy** $\boldsymbol{\pi}$ to evoke an **action** $\boldsymbol{a}$ before transferring to a new state $\boldsymbol{s'}$, while the interactive environment gives a corresponding **reward** $\boldsymbol{r}$.
2) **State** $\boldsymbol{s}$: The trust model aims to predict the credibility of the object by utilizing the interaction between entities, so the trust evidence introduced in Sec. IV-A is the state. Each state is a triple, denoted as $s=(\mathcal{C},\mathcal{E},\mathcal{D})$, where $\mathcal{C},\mathcal{E},\mathcal{D}\in[0,1]$.
3) **Action** $\boldsymbol{a}$: The action is defined as the predicted trust of the trustor on the behavior of the trustee, $a\in[0,1]$.
4) **Reward** $\boldsymbol{r}$: The reward represents a value that is fed back to the agent via the interactive environment once the agent performs an action. In the proposed scheme, the reward is negatively correlated with the deviation of the predicted trust from the actual trust, and is defined as:

$$r=-\sum_{i=0}^{2} w_i(a-s[i])^2, \tag{7}$$

where $s[i]$ represents the i^{th} attribute of the state s (i.e., the i^{th} evidence); $w_i=1-\frac{s[i]}{\sum_0^2 s[i]}$, which represents the weight of the i^{th} evidence, that is, the smaller the evidences (i.e., evidence $\mathcal{C}$ is smaller than the evidence $\mathcal{E}$ and $\mathcal{D}$), the more likely there is this type of attack (i.e., the communication attack), so a greater weight is given.
5) **Policy** $\boldsymbol{\pi}$: The policy guides the agent to take corresponding actions based on its own state. According to previous definitions of the state and action, the policy is precisely the trust model in the proposed scheme.

Based on the above definitions, the deep deterministic policy gradient (DDPG) algorithm is adopted for model training because of the continuous state and action spaces. As shown in Fig. 5, the specific training process based on DDPG can be outlined as follows.

1) The online policy network outputs an action based on the current state (s_t) and adds the action into a stochastic process $(\mathcal{N}_t)$ to increase the exploration.

$$a_t=\mathcal{N}_t\left(\boldsymbol{\mu}\left(s_t;\boldsymbol{\theta}\right),v\right), \tag{8}$$

where v is updated by shrinking the value during each learning process.
2) The interactive environment executes the action a_t, returns a reward r_t and the new state s_{t+1}.
3) The transition (s_t,a_t,r_t,s_{t+1}) is stored in the replay buffer as the dataset for training the online policy network.
4) N transitions (s_i,a_i,r_i,s_{i+1}) are randomly sampled from the replay buffer as a mini-batch training data for the online policy network and online value network.
5) The gradient of the online value network is computed as

$$y_i=r_i+\gamma\boldsymbol{Q}\left(s_{i+1},\boldsymbol{\mu}\left(s_{i+1};\boldsymbol{\theta'}\right);\boldsymbol{w'}\right), \tag{9}$$

where γ symbolizes the discount factor.

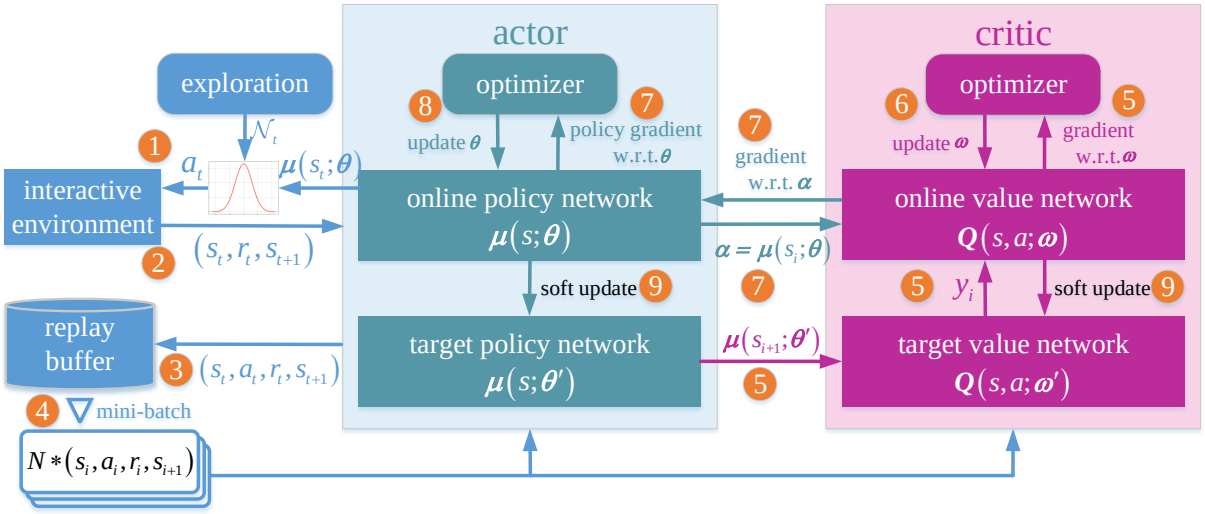


Fig. 5: Training framework of the DDPG algorithm.

6) The online value network is updated based on Adam optimizer [38].
7) The policy gradient of the online policy network is estimated using the Monte Carlo method:

$$\nabla_{\boldsymbol{\theta}} J \approx \frac{1}{N} \sum_{i} \nabla_a Q\left(s, a; w\right)|_{s=s_i, a=a_i} \cdot \nabla_\theta \mu\left(s; \theta\right)|_{s=s_i} \tag{10}$$

8) The online policy network is updated based on Adam optimizer.
9) Soft update for the target network parameters $\boldsymbol{\theta}'$ and $\boldsymbol{w}'$:

$$\begin{cases} \boldsymbol{\theta}' \leftarrow \tau \boldsymbol{\theta} + (1-\tau)\,\boldsymbol{\theta}' \\ \boldsymbol{w}' \leftarrow \tau \boldsymbol{w} + (1-\tau)\,\boldsymbol{w}' \end{cases} \tag{11}$$

Owing to a lack of sufficient interactions between underwater entities, directly using the initial trust model for training in real environments will lead to long-term instability in the prediction accuracy. Therefore, based on the above-mentioned DDPG framework, the main controller initially implements model pre-training and then distributes the pre-trained parameters to local controllers for further training.

Pre-training: Following network initialization, the main controller implements model pre-training based on a virtual interactive environment. As illustrated in Fig. 6, the workflow of the virtual interactive environment comprises five modules. As mentioned before, the input a_t of the interactive environment represents the trust score of the trustor for the trustee, and as a result, **module 1** decides whether the trustor interacts with the trustee with a_t as the interaction probability. Furthermore, **module 2** randomly generates the trustworthiness of the trustee, which determines whether the behavior of the trustee is normal or malicious. Subsequently, **module 3** simulates the interaction between the trustor and the trustee to update attributes, such as the number of successful communications, remaining energy, and data accuracy. Thereafter, the updated attributes are input to **module 4** to obtain the trust evidence and reward. Finally, **module 5** outputs the transition (s_t, r_t, s_{t+1}). Combined with the DDPG framework and the virtual interactive environment, the main controller obtains a set of parameters $(\boldsymbol{\theta}, \boldsymbol{\theta}', \boldsymbol{\omega}, \boldsymbol{\omega}')$ that converges the model. Finally, the mean of the parameters $(\bar{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}', \bar{\boldsymbol{\omega}}, \bar{\boldsymbol{\omega}}')$ obtained following multiple convergences is delivered to the local controllers as pre-training parameters.

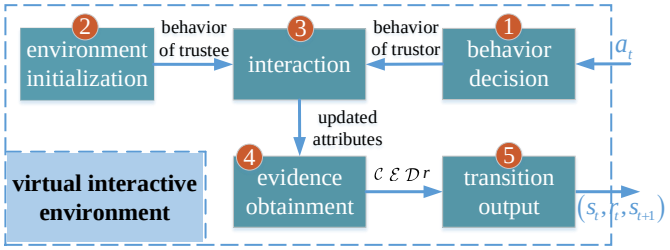


Fig. 6: Workflow of the virtual interactive environment.

Local training: Each local controller initializes its own training model with the received pre-trained parameters $(\bar{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}', \bar{\boldsymbol{\omega}}, \bar{\boldsymbol{\omega}}')$. Subsequently, the trust prediction model $\boldsymbol{\mu}\left(s; \bar{\boldsymbol{\theta}}\right)$ is distributed to the collectors within its respective area. Following that, the collectors interact with each other according to the trust prediction model, and periodically deliver the transitions $\{(s_i, a_i, r_i, s_{i+1})\}$ to the local controller. The local controller then stores the transitions from different collectors to the local replay buffer and adopts the same mini-batch method to train the local model.

C. Trust Model Update Based on Federated Learning

The local training utilizes the actual interaction experience between regional entities to update the model, thereby improving the prediction accuracy of the model in the actual environment. However, fluctuations in the underwater environment and changes in network topology can affect historical experience-based models, rendering them unable to accurately adjudicate current conditions. For example, the trust model trained based on local interaction experience may be unable to accurately adjudicate the trustworthiness of entities from other regions when movable entities cruise between regions subject to different local controllers, where the acoustic conditions

tend to differ. FL makes agents collaboratively learn a shared predictive model, while all training data remain on the individual underwater device (e.g., local controllers), decoupling model update from the data stored [39]. Therefore, we adopt a FL framework to globally update the trust model.

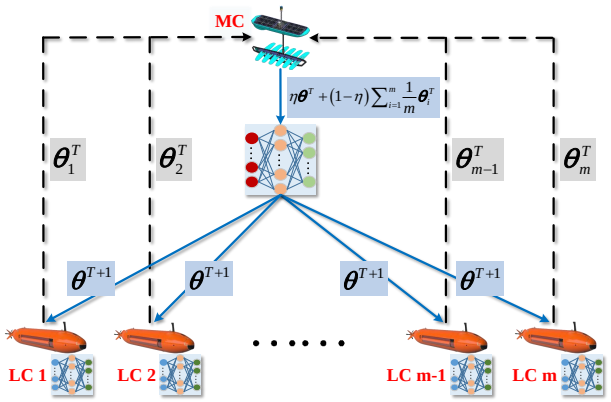


Fig. 7: Trust model update based on federated learning.

As indicated in Fig. 7, we consider that the current network contains a unique main controller **MC** and m local controllers $\{\boldsymbol{LC_1}, \boldsymbol{LC_2}, \cdots, \boldsymbol{LC_{m-1}}, \boldsymbol{LC_m}\}$. Each time T $(T > t)$, local controllers send their latest model parameters $\left\{\boldsymbol{\theta}_1^T, \boldsymbol{\theta}_2^T, \cdots, \boldsymbol{\theta}_m^T, \boldsymbol{\theta}_{m-1}^T\right\}$ to the **MC**, it then updates the global model parameters as follows:

$$\boldsymbol{\theta}^{T+1} = \eta\boldsymbol{\theta}^T + (1-\eta)\sum_{i=1}^{m}\frac{1}{m}\boldsymbol{\theta}_i^T, \tag{12}$$

where the variable η indicates a soft update coefficient as τ in Eq. 11, which tends to be a value close to 1.

Finally, the **MC** sends the updated variable $\boldsymbol{\theta}^{T+1}$ to all local controllers to replace their local parameters. The local controllers continue updating the local parameters according to the interaction experience from the underwater environment and repeat the above updating process with period T.

V. Simulation Results and Analysis

A. Simulation Settings

In this study, the proposed FedTM, and other established trust models, were simulated using TensorFlow 2.8.0 to evaluate and compare their performances. First, we tested the impact of some key hyperparameters on the performance of the proposed FedTM, including the learning rate, step size, and size of the minibatch. Thereafter, the performance of the FedTM was compared with other related work: the WEIGHT, ARTMM [22], and TUMRL [31]. The WEIGHT algorithm is not a published method, but takes the same way of evidence calculation as the scheme proposed in this study while utilizing weighted sum of evidences instead of neural network predictions. The ARTMM algorithm is a classic trust model for UASNs. Its underlying method for quantifying trust evidence has been an inspiration for several research work including ours. The main idea behind TUMRL, our previous work, is to use reinforcement learning techniques to dynamically change the weighting of trust evidence to adapt to a dynamic environment. The performance of the schemes was compared in terms of average prediction error and energy efficiency. The experiment uses the trust model's output, the trust score, as the interaction probability to lower the frequency of interactions between nodes with low trust scores, enhancing the security of data routing. The simulation-related parameter settings are listed in Table I.

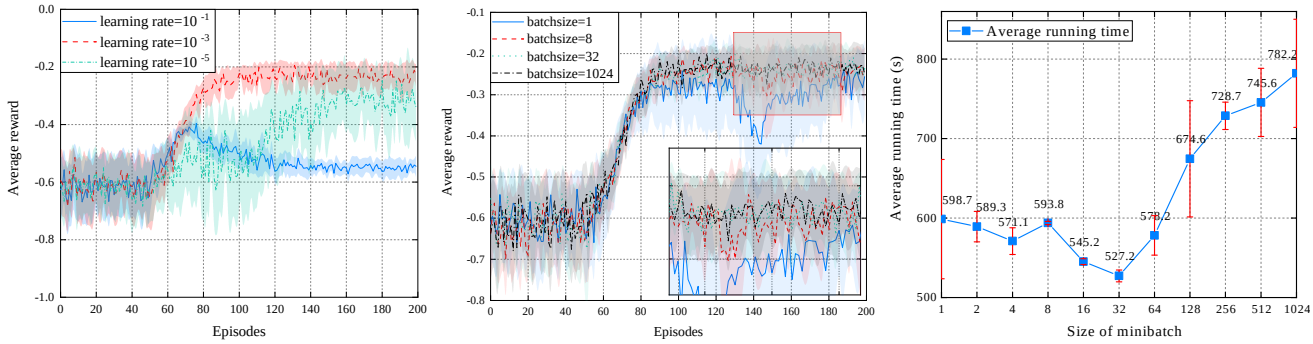
TABLE I: Simulation Parameters

Parameter	Value
Network size	$500 \times 500 \times 500\,\mathrm{m}^3$
Network sub-region size	$250 \times 250 \times 500\,\mathrm{m}^3$
Number of local controllers	4
Number of collectors	100
Communication radius of collectors	150 m
Default initial energy of collectors	1000 J
Update cycle of the local model	1
Update cycle of the global model	5
Mean of normal absolute trustworthiness	0.95
Mean of malicious absolute trustworthiness	0.1

B. Performance of FedTM

The policy network and value network in the DDPG algorithm use gradient ascent and gradient descent for parameter updates, respectively. Meanwhile, the learning rate is related to the convergence speed of gradient updates, which in turn affects the performance of the scheme. Therefore, we first evaluated the relationship between the convergence of the FedTM and the learning rate of different orders of magnitude. The convergence of DRL algorithms is generally measured by the average reward, which is also used in this study to evaluate convergence. The average reward refers to the ratio of the cumulative reward obtained in each episode to the number of exploration steps. As illustrated in Fig. 8(a), the agent solely explores and does not learn within the initial 50 episodes to add noisy experience to the replay buffer. The agent starts learning from the 51^{th} episode and gradually reduces the noise added in the action to achieve convergence. Figure 8(a) illustrates that the convergence is better than other orders of magnitude when the learning rate is of the order of 10^{-3}. When the learning rate is too large (10^{-1}), it is difficult for the algorithm to converge to the optimal solution, and when the learning rate is too small (10^{-5}), the algorithm requires more time to converge. Therefore, in subsequent experiments, the learning rate was set to 10^{-3}.

Figures 8(b) and 8(c) depict the effect of another key hyperparameter, the size of minibatch, on the convergence of the algorithm. The former suggests that the convergence stability of the algorithm enhances as the size of the minibatch increases, while the latter shows that the convergence speed is insensitive to the changes in batch size. When the batch size is small, the number of samples taken from the replay buffer for training the neural network is smaller each time, which results in the gradient descent being more susceptible to individual samples, ultimately making convergence unstable. Therefore, the size of the minibatch should be as large as possible. Figure 8(c) further illustrates the running time of the algorithm at different sizes of the minibatch. Overall, the



(a) Convergence vs. learning rate (b) Convergence vs. mini-batch (c) Running time vs. mini-batch

Fig. 8: The influence of key hyperparameters (learning rate, mini-batch size) on the convergence performance of the proposed FedTM.

average running time increases as the size of the minibatch rises, suggesting that larger minibatch sizes are not better. Therefore, the size of the minibatch is set to 32 by default in subsequent experiments.

C. Comparison with Related Work

1) Comparison of Average Prediction Error: The FedTM algorithm was compared with the WEIGHT, ARTMM, and TUMRL algorithms in terms of average prediction error. The average prediction error (ape) is defined as the mean value of the difference between the true trustworthiness and the predicted trust value of all nodes, and it is calculated as follows.

$$ape = \frac{1}{NM_i}\sum_{i=1}^{N}\sum_{j=1}^{M_i}|t_{ji} - T_i|, \tag{13}$$

where N refers to the number of nodes in the network involved in trust evaluation, M_i denotes the number of neighbors of node i, t_{ji} represents the trust prediction of neighbor j for i, and Ti indicates the true trust value of node i. It is important to note that we define the average prediction error as a performance evaluation criterion on the premise that the objective of this article is to accurately predict the true trustworthiness of a node, which is the result of the combined effect of node maliciousness, selfishness, and communication instability.

Figure 9 illustrates how the average prediction error varies with the interaction timeslot for the case, where 10% of the nodes in the network are malicious. In the simulation experiments, a malicious node randomly executes one of the three types of attacks: selective forwarding, DoS, and data tampering, at each timeslot. Moreover, the malicious node correctly calculates the trust evidence but modifies the recommended values sent to its neighbors by inverting the true evidence ($evi\_rec = 1 - evi\_tru$), where evi_rec and evi_tru refer to the recommended evidence sent to neighbors and the true evidence, respectively. Based on the results presented in Figure 9, the average prediction error for all scenarios except TUMRL exhibits a decreasing trend and stabilizes over time. Overall, the average prediction error of the proposed FedTM scheme is lower than that of the other comparative schemes. Additionally, between timeslot 0 and timeslot 15, the prediction performance of FedTM is inferior to that of the TUMRL scheme because the initial trust prediction model is trained by interacting with a virtual environment and therefore has insufficient prediction accuracy. However, the rapid decrease in the average prediction error at this stage further demonstrates the ability of the FedTM scheme to quickly converge and achieve better prediction accuracy than the other schemes, validating the reliability of the proposed scheme.

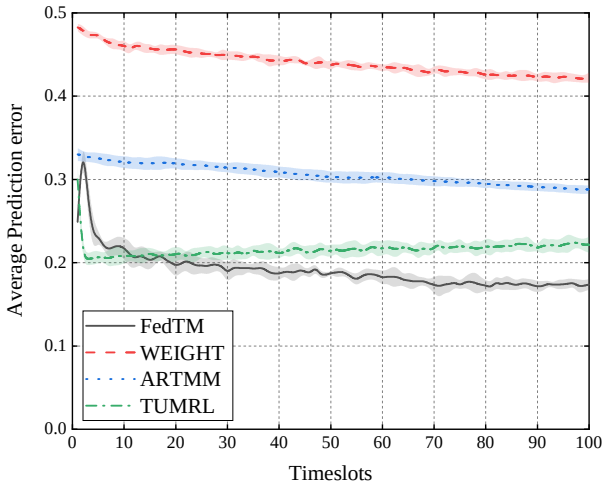


Fig. 9: Average prediction error versus timeslots of the examined trust models (FedTM, WEIGHT, ARTMM, and TUMRL).

Figure 10 illustrates the average prediction error versus the proportion of malicious nodes in the network. The average prediction error is the result of running the network up to the 100^{th} timeslot, as the average prediction error for each scenario has largely reached a steady state at this point. Overall, the proposed FedTM scheme significantly outperforms other comparative algorithms in terms of the average prediction error. Moreover, Fig. 10 depicts an unexpected result, where the average prediction error of both the FedTM and WEIGHT algorithms decreases as the proportion of malicious nodes increases. It can be attributed to the fact that both of them use the same trust evidence quantification method, in which only data trust evidence requires recommendation

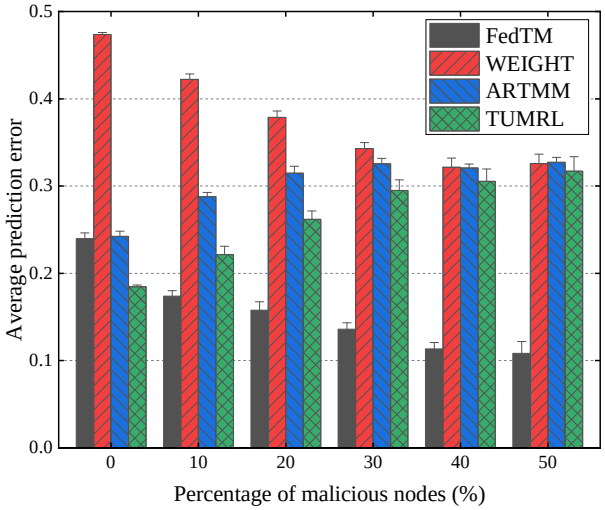


Fig. 10: Average prediction error as a function of changing proportion of malicious nodes.

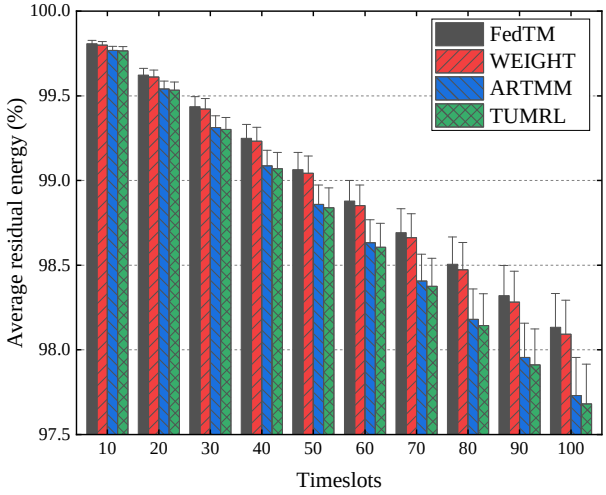


Fig. 11: Average residual energy versus timeslots.

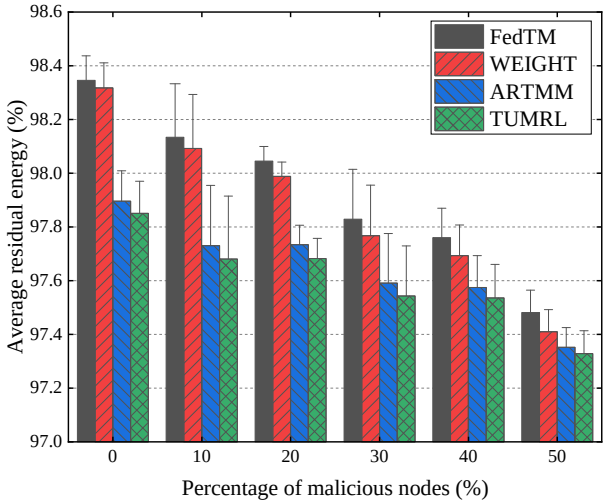


Fig. 12: Average residual energy with changing proportion of malicious nodes.

information from neighbors. However, the proposed data trust calculation method designs an anomaly recommendation filtering mechanism, which effectively limits the influence of malicious recommendations. Furthermore, as the proportion of malicious nodes increases, malicious behavior makes the true trustworthiness of the malicious nodes more predictable compared to normal nodes, thereby decreasing the average prediction error.

2) Comparison of the Average Residual Energy: The accurate functioning of the scheme is supported by the data interaction between nodes, which is the primary source of energy consumption. Therefore, we further compare the system performance by observing the average residual energy (are). The average residual energy represents the average ratio of the residual energy to the initial energy of all nodes and is defined as

$$are = \frac{1}{N}\sum_{i=1}^{N}\frac{E_i}{E_0}, \tag{14}$$

where N denotes the number of nodes involved in the interaction in the network, E_i represents the energy remaining at node i at the end of some timeslot, and E_0 denotes the initial energy of the node.

Figure 11 illustrates the variation of the average residual energy over time in the presence of 10% of malicious nodes in the network. Overall, the average residual energy significantly decreases with increasing timeslots for all schemes; however, the reductions are lighter for FedTM and WEIGHT algorithms. The reason lies in that they adopt the same evidence generation method proposed in Sec. IV-A. The method only requires the support of neighborhood data during the quantification of data evidence, and otherwise relies on the direct interaction between the trustor and trustee, thus effectively reducing the energy consumption of the interaction process compared to traditional evidence generation strategies. Furthermore, the proposed FedTM scheme outperforms WEIGHT in terms of energy consumption because FedTM can predict the true trustworthiness of attackers more accurately. Therefore, the interaction probability of the attackers can be reduced, thereby reducing the additional energy consumption of the attacked node. This contribution is ultimately reflected in the fact that FedTM outperforms WEIGHT in terms of energy consumption.

In Fig. 11, it can be observed that the length of the error bars gradually increases with time. This is because there are many random factors in the simulation environment, such as the probability of interaction between nodes, trust prediction results, and data generated by nodes, which accumulate their influence over time, eventually showing up as an increase in the error of the results. However, this phenomenon is not evident in Fig. 9, the reason lies in the fact that the average prediction error is verified with fixed interaction probabilities in preference to random ones.

In Fig. 12, the performance of the average residual energy is further tested with different proportions of malicious nodes, where the values in the vertical coordinate represent all the results collected at the 100^{th} timeslot. The results presented

in Fig. 12 reveal that the average residual energy of all the schemes gradually decreases as the proportion of malicious nodes rises; however, FedTM outperforms the other schemes from start to finish. This is due to the fact that FedTM can effectively detect malicious nodes in the network, which in turn reduces the probability of interacting with malicious nodes, thus saving energy in the nodes.

VI. Conclusions

This study explored the problem of effective trust modeling in the presence of movable underwater devices, heterogeneous hydroacoustic environments, and variable attack patterns. A novel trust model based on federated DRL was proposed. The entire UASN was divided into sub-networks attached to different local controllers to enhance cross-domain experience interaction, and the scheme was centrally regulated by using a global controller. Based on an improved evidence generation method, each local controller quantifies the interaction data between nodes in its jurisdiction into corresponding trust evidence. The obtained evidence was then fed into the corresponding local trust model, which is based on DRL for trust prediction and model training. The global controller periodically aggregates and updates the parameters of each local model using FL. Moreover, the experimental analysis indicates that the proposed FedTM algorithm provides increased performance compared to previously reported methods, particularly in terms of defective node detection and energy utilization. The superiority of the proposed scheme is more evident when measuring the long-term performance of FedTM, where the networks are composed of heterogeneous devices with relatively high dynamics and varying attack conditions.

References

[1] E. M. Sozer, M. Stojanovic, and J. G. Proakis, "Underwater acoustic networks," *IEEE journal of oceanic engineering*, vol. 25, no. 1, pp. 72–83, 2000.

[2] M. Erol-Kantarci, H. T. Mouftah, and S. Oktug, "A survey of architectures and localization techniques for underwater acoustic sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 3, pp. 487–502, 2011.

[3] G. Han, C. Zhang, L. Shu, and J. J. Rodrigues, "Impacts of deployment strategies on localization performance in underwater acoustic sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 3, pp. 1725–1733, 2014.

[4] G. Han, J. Jiang, N. Sun, and L. Shu, "Secure communication for underwater acoustic sensor networks," *IEEE communications magazine*, vol. 53, no. 8, pp. 54–60, 2015.

[5] G. Han, Y. He, J. Jiang, H. Wang, Y. Peng, and K. Fan, "Fault-tolerant trust model for hybrid attack mode in underwater acoustic sensor networks," *IEEE Network*, vol. 34, no. 5, pp. 330–336, 2020.

[6] J. Wang, Z. Yan, H. Wang, T. Li, and W. Pedrycz, "A survey on trust models in heterogeneous networks," *IEEE Communications Surveys & Tutorials*, 2022.

[7] B. Cai, X. Li, W. Kong, J. Yuan, and S. Yu, "A reliable and lightweight trust inference model for service recommendation in siot," *IEEE Internet of Things Journal*, vol. 9, no. 13, pp. 10 988–11 003, 2021.

[8] Z. Li, F. Xiong, X. Wang, Z. Guan, and H. Chen, "Mining heterogeneous influence and indirect trust for recommendation," *IEEE Access*, vol. 8, pp. 21 282–21 290, 2020.

[9] S. Mandal and A. Maiti, "Heterogeneous trust-based social recommendation via reliable and informative motif-based attention," in *2022 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2022, pp. 1–8.

[10] Q. Cui, Z. Zhu, W. Ni, X. Tao, and P. Zhang, "Edge-intelligence-empowered, unified authentication and trust evaluation for heterogeneous beyond 5g systems," *IEEE Wireless Communications*, vol. 28, no. 2, pp. 78–85, 2021.

[11] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017.

[12] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE signal processing magazine*, vol. 37, no. 3, pp. 50–60, 2020.

[13] J. Jiang, G. Han, F. Wang, L. Shu, and M. Guizani, "An efficient distributed trust model for wireless sensor networks," *IEEE transactions on parallel and distributed systems*, vol. 26, no. 5, pp. 1228–1237, 2014.

[14] X. Wu, J. Huang, J. Ling, and L. Shu, "Bltm: beta and lqi based trust model for wireless sensor networks," *IEEE Access*, vol. 7, pp. 43 679–43 690, 2019.

[15] B. Pang, Z. Teng, H. Sun, C. Du, M. Li, and W. Zhu, "A malicious node detection strategy based on fuzzy trust model and the abc algorithm in wireless sensor network," *IEEE Wireless Communications Letters*, vol. 10, no. 8, pp. 1613–1617, 2021.

[16] S. Ding, Z. Yue, S. Yang, F. Niu, and Y. Zhang, "A novel trust model based overlapping community detection algorithm for social networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 11, pp. 2101–2114, 2019.

[17] Q. Wang, W. Zhao, J. Yang, J. Wu, S. Xue, Q. Xing, and S. Y. Philip, "C-deeptrust: A context-aware deep trust prediction model in online social networks," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[18] C. Ge, L. Zhou, G. P. Hancke, and C. Su, "A provenance-aware distributed trust model for resilient unmanned aerial vehicle networks," *IEEE Internet of Things Journal*, vol. 8, no. 16, pp. 12 481–12 489, 2020.

[19] Z. Yang, R. Wang, D. Wu, B. Yang, and P. Zhang, "Blockchain-enabled trust management model for the internet of vehicles," *IEEE Internet of Things Journal*, 2021.

[20] P. Zhang, M. Zhou, and Y. Kong, "A double-blind anonymous evaluation-based trust model in cloud computing environments," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 3, pp. 1805–1816, 2019.

[21] W. Mo, W. Liu, G. Huang, N. N. Xiong, A. Liu, and S. Zhang, "A cloud-assisted reliable trust computing scheme for data collection in internet of things," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 7, pp. 4969–4980, 2021.

[22] G. Han, J. Jiang, L. Shu, and M. Guizani, "An attack-resistant trust model based on multidimensional trust metrics in underwater acoustic sensor network," *IEEE Transactions on Mobile Computing*, vol. 14, no. 12, pp. 2447–2459, 2015.

[23] A. Bolster and A. Marshall, "Single and multi-metric trust management frameworks for use in underwater autonomous networks," in *2015 IEEE Trustcom/BigDataSE/ISPA*, vol. 1. IEEE, 2015, pp. 685–693.

[24] ——, "Analytical metric weight generation for multi-domain trust in autonomous underwater manets," in *2016 IEEE Third Underwater Communications and Networking Conference (UComms)*. IEEE, 2016, pp. 1–5.

[25] J. Jiang, G. Han, L. Shu, S. Chan, and K. Wang, "A trust model based on cloud theory in underwater acoustic sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 1, pp. 342–350, 2015.

[26] J. Jiang, G. Han, C. Zhu, S. Chan, and J. J. Rodrigues, "A trust cloud model for underwater wireless sensor networks," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 110–116, 2017.

[27] G. Han, J. Du, C. Lin, H. Wu, and M. Guizani, "An energy-balanced trust cloud migration scheme for underwater acoustic sensor networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 1636–1649, 2019.

[28] J. Du, G. Han, C. Lin, and M. Martinez-Garcia, "Itrust: an anomaly-resilient trust model based on isolation forest for underwater acoustic sensor networks," *IEEE Transactions on Mobile Computing*, 2020.

[29] J. Du, G. Han, C. Lin, and M. Martínez-García, "Ltrust: An adaptive trust model based on lstm for underwater acoustic sensor networks," *IEEE Transactions on Wireless Communications*, 2022.

[30] G. Han, Y. He, J. Jiang, N. Wang, M. Guizani, and J. A. Ansere, "A synergetic trust model based on svm in underwater acoustic sensor networks," *IEEE transactions on vehicular technology*, vol. 68, no. 11, pp. 11 239–11 247, 2019.

[31] Y. He, G. Han, J. Jiang, H. Wang, and M. Martinez-Garcia, "A trust update mechanism based on reinforcement learning in underwater

acoustic sensor networks," *IEEE Transactions on Mobile Computing*, vol. 21, no. 3, pp. 811–821, 2020.
[32] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.
[33] M. H. ur Rehman, A. M. Dirir, K. Salah, E. Damiani, and D. Svetinovic, "Trustfed: A framework for fair and trustworthy cross-device federated learning in iiot," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 8485–8494, 2021.
[34] N. Bugshan, I. Khalil, M. S. Rahman, M. Atiquzzaman, X. Yi, and S. Badsha, "Toward trustworthy and privacy-preserving federated deep learning service framework for industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1535–1547, 2022.
[35] O. A. Wahab, G. Rjoub, J. Bentahar, and R. Cohen, "Federated against the cold: A trust-based federated learning approach to counter the cold start problem in recommendation systems," *Information Sciences*, vol. 601, pp. 189–206, 2022.
[36] V. Mothukuri, R. M. Parizi, S. Pouriyeh, A. Dehghantanha, and K.-K. R. Choo, "Fabricfl: Blockchain-in-the-loop federated learning for trusted decentralized systems," *IEEE Systems Journal*, vol. 16, no. 3, pp. 3711–3722, 2021.
[37] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020.
[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[39] X. Wang, C. Wang, X. Li, V. C. Leung, and T. Taleb, "Federated deep reinforcement learning for internet of things with decentralized cooperative edge caching," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9441–9455, 2020.

Yu He received the B.S. and Ph.D. degree in information and communication engineering from Hohai University, Changzhou, China, in 2017 and 2023. He was also a visiting Ph.D. student under the support of the China Scholarship Council (CSC) at the School of Electrical Engineering, Aalto University, Espoo, Finland from 2021 to 2022. He is currently a Post-Doctoral Researcher with the Information Department, Hohai University, Nanjing, China. His research interests include underwater sensor networks, trust management and reinforcement learning.

Guangjie Han (Fellow, IEEE) is currently a Professor with the Department of Internet of Things Engineering, Hohai University, Changzhou, China. He received his Ph.D. degree from Northeastern University, Shenyang, China, in 2004. In February 2008, he finished his work as a Postdoctoral Researcher with the Department of Computer Science, Chonnam National University, Gwangju, Korea. From October 2010 to October 2011, he was a Visiting Research Scholar with Osaka University, Suita, Japan. From January 2017 to February 2017, he was a Visiting Professor with City University of Hong Kong, China. From July 2017 to July 2020, he was a Distinguished Professor with Dalian University of Technology, China. His current research interests include Internet of Things, Industrial Internet, Machine Learning and Artificial Intelligence, Mobile Computing, Security and Privacy. Dr. Han has over 500 peer-reviewed journal and conference papers, in addition to 160 granted and pending patents. Currently, his H-index is 59 and i10-index is 250 in Google Citation (Google Scholar). The total citation count of his papers raises above 13100+ times.

Dr. Han is a Fellow of the UK Institution of Engineering and Technology (FIET). He has served on the Editorial Boards of up to 10 international journals, including the IEEE TCCN, IEEE Systems, IEEE/CCA JAS, IEEE Network, etc. He has guest-edited several special issues in IEEE Journals and Magazines, including the IEEE JSAC, IEEE Communications, IEEE Wireless Communications, IEEE Transactions on Industrial Informatics, Computer Networks, etc. Dr. Han has also served as chair of organizing and technical committees in many international conferences. He has been awarded 2020 IEEE Systems Journal Annual Best Paper Award and the 2017-2019 IEEE ACCESS Outstanding Associate Editor Award. He is a Fellow of IEEE.

Aohan Li [S'17, M'20] received the Ph.D. degree from Keio University, Yokohama, Japan in 2020. From 2020 to 2022, she was an Assistant Professor at the Tokyo University of Science, Tokyo, Japan. Currently, she is an Assistant Professor at the University of Electro-Communications, Tokyo, Japan. Her current research interests include machine learning, resource management, and Internet of Things. She has published over 50 papers in related journals and international conferences. She was the recipient of the 9th International Conference on Communications and Networking in China 2014 (CHINACOM'14) Best Paper Award, and the 3rd International Conference on Artificial Intelligence in Information and Communication (ICAIIC'21) Excellent Paper Award. She is a member of IEICE.

Tarik Taleb (Senior Member, IEEE) received the B.E. degree (with distinction) in information engineering and the M.Sc. and Ph.D. degrees in information sciences from Tohoku University, Sendai, Japan, in 2001, 2003, and 2005, respectively. He is currently a Professor at the Center of Wireless Communications, the University of Oulu, Finland. He is the founder and the Director of the MOSA!C Lab, Espoo, Finland. He was an Assistant Professor with the Graduate School of Information Sciences, Tohoku University, in a laboratory fully funded by KDDI until 2009. He was a Senior Researcher and a 3GPP Standards Expert with NEC Europe Ltd., Heidelberg, Germany. He was then leading the NEC Europe Labs Team, involved with research and development projects on carrier cloud platforms, an important vision of 5G systems. From 2005 to 2006, he was a Research Fellow with the Intelligent Cosmos Research Institute, Sendai. He has also been directly engaged in the development and standardization of the Evolved Packet System as a member of the 3GPP System Architecture Working Group. His current research interests include architectural enhancements to mobile core networks (particularly 3GPP's), network softwarization and slicing, mobile cloud networking, network function virtualization, software defined networking, mobile multimedia streaming, intervehicular communications, and social media networking.

Chenyang Wang [S'18, M'21] received the B.S. and M.S. degrees in computer science and technology from Henan Normal University, Xinxiang, China, in 2013 and 2017, respectively. He is currently pursuing a Ph.D. degree from the School of Computer Science and Technology, College of Intelligence and Computing, Tianjin University, Tianjin, China. He is also a visiting Ph.D. student under the support of China Scholarship Council (CSC) at the School of Electrical Engineering, Aalto University, Espoo, Finland since 15 May 2021. His current research interests include edge computing, big data analytic, reinforcement learning, and deep learning. He received the Best Student Paper Award of the 24th International Conference on Parallel and Distributed Systems by IEEE Computer Society in 2018. He also received the Best Paper Award of IEEE International Conference on Communications in 2021.

Hao Yu received the B.S. and Ph.D. degree in communication engineering from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2015 and 2020. He was also a Joint-Supervised Ph.D. Student with the Politecnico di Milano, Milano, Italy. He is currently a Postdoctoral Researcher with the Center of Wireless Communications, University of Oulu, Finland. His research interests include network automation, SDN/NFV, edge intelligence, time sensitive networks, deterministic networking.