Intelligent Drone-Assisted Fault Diagnosis for B5G-Enabled Space-Air-Ground-Space Networks

Xiaoding Wang, Hui Lin, Hongyan Zhang, Dejun Miao, Qinyang Miao, and Wenxin Liu

***Abstract*—The ubiquitous network services provided by the Beyond 5G enabled space-air-ground-sea networks (B5G-SAGS) depends on the reliability of each intelligent device within. However, the QoS of B5G-SAGS could be compromised if there exists faults on individual network. That suggests the significance of fault-diagnosis in the B5G-SAGS design. Previous works on fault diagnosis were designed without extra information to improve diagnosis accuracy. In this paper, we propose an Intelligent Drone-assisted Fault-diagnosis Algorithm (IDFA) utilizing B5G-enabled Multiple-access Edge Computing/Cloud (B5G-MEC) services to detect faulty buoys. Specifically, IFDA first employs a Cubature Kalman Filter based Radial Bias Function Neural Network (CKF-RBFNN) for each fault-diagnosis center to perform preliminary fault detection based on the data provided by both buoys and drones. The data collection path is planned utilizing the deep reinforcement learning algorithm, Deep Deterministic Policy Gradient (DDPG), on B5G-MEC servers for energy efficiency. Eventually, the collective decision made by all fault-diagnosis centers determines the faulty status of each buoy. The theoretical analysis and validation experiments show that: (i) the IDFA has a better diagnosis accuracy in both single fault detection and multi-fault classification while compared with contemporary algorithms; (ii) the IDFA obtains a high aggregation ratio and a low energy cost.**

***Index Terms*—Beyond 5G, deep reinforcement learning, fault-diagnosis, multiple-access edge computing, space-air-ground-sea networks.**

I. Introduction

THE rapidly growing Internet traffic, the demand for various terminals access, the highly required low-latency, and the reliability of industrial applications have given birth to 5G communication technology. The application of 5G have brought a great leap on communication capability in mobility, delay, traffic density, and energy efficiency. Further, Beyond 5G (B5G) should be able to expend communications on terrain surface to sea, air and even space to truly realize the "ubiquitous connection" of the entire world by coordinating transmission of multiple access and unifying management of multiple system resources [1]. That suggests a cross-airspace, cross-regional and cross-sea network is required for seamless global coverage. Thereby, an integrated network should have the following components, i.e., a space network of satellites on various orbits, an air network of aircrafts, a ground network of cellular networks and vehicle networks, and a sea network of ships and buoys. Besides, mobile satellite terminals, satellite base stations, and ground data processing centers should be considered as well.

There are three components of a satellite communication system [2], i.e., the space section, the ground section, and the space-ground link. Specifically, a variety of satellite control centers, control networks and stations consists of the ground section. In space section, satellites play import roles, the inter-satellite links of which can receive and forward data. Apart from both space section and ground section, there are various smart terminals, i.e., IoT devices, that comprises the user segment. One of many advantages of the satellite communication system is the coverage without being confined by terrains. That indicates the competitiveness of satellite communications within the areas of insufficient ground coverage.

In addition, the High-Altitude Platform Station/HAPS IMT BS (HAPS/HIBS) can provide the air communications, i.e., telecommunication services at high altitude through wireless base-stations installed on aircrafts, and air-ground communications as well utilizing existing communication technologies [3]. That suggest the shortcomings of ground networks are made up due to flexibility in deployment, wide coverage in service, and slight impact caused by ground networks. Furthermore, the line-of-sight transmission signals allow HAPS/HIBS to communicate with mobile phones in low energy loss and high transmission quality. Thereby, HAPS/HIBS is considered as an effective extension of the ground network.

The space-air-ground-sea integrated communication network (SAGS) is capable of providing ubiquitous coverage of low-cost. In SAGS, the ground network not only supports the storage and process of big data but improves data transmission efficiency in the majority of land areas as well. On the other hand, the 3D coverage is provided for surveillance on remote areas, airspace and oceans by non-ground networks to aid ground networks for ubiquitous coverage. In addition, by deeply integrating with space network, air network, ground network, and sea network, we can make full use of HAPS/HIBS, satellites and terrestrial 5/6 G networks to offer both intelligent

Manuscript received August 5, 2020; revised November 26, 2020; accepted December 3, 2020. Date of publication December 9, 2020; date of current version December 9, 2021. This work was supported by the National Natural Science Foundation of China under Grants 61702103 and U1905211, Natural Science Foundation of Fujian Province of China under Grants 2020J01167 and 2020J01169. Recommended for acceptance by Dr. A. Al-Dubai. *(Corresponding author: Hui Lin and Hongyan Zhang.)*

Xiaoding Wang, Hui Lin, Qinyang Miao, and Wenxin Liu are with the College of Mathematics and Informatics, Fujian Normal University, Fuzhou 350117, China (e-mail: wangdin1982@163.com; linhui@fjnu.edu.cn; 18291972715@163.com; sixwenxin@163.com).

Hongyan Zhang is with the Concord University College Fujian Normal University, Fuzhou 350117, China (e-mail: zhyan82712@163.com).

Dejun Miao is with the Yangzhou Polytechic College, Yangzhou 225009, China (e-mail: miaodejunuk@gmail.com).

Digital Object Identifier 10.1109/TNSE.2020.3043624

accesses and time-sensitive services globally [4], [5]. Without affecting by geographical environment and natural disasters, HAPS/HIBS and satellite communications, which can support wide coverage, apply to emergency communications establishment. For example, natural disasters (i.e., earthquakes, floods, etc.) could paralyze the ground network. In this case, we can deploy HAPS/HIBS and satellite communications quickly due to the flexibility and mobility of terminal devices such that massive rescue data will be dispatched and aggregated reliably.

All in all, space-air networks have disadvantages in high-latency, besides the cost of storage of satellites is much higher compared with that of terrestrial ones. Thereby, the Multiple-access Edge Computing/Cloud based technology (MEC) [6], [7] is introduced to provide services to overcome the shortcoming of high capacity cost and high latency of non-terrestrial networks. Specifically, as a computation resource deployed at the edge, MEC combines storage, core network computing, and other capabilities at the edge of the space-air-ground-sea integrated network to reduce transmission delays while providing service. Although B5G-enabled SAGS networks (B5G-SAGS) have the ability to make up the individual deficiencies of either space, air, ground or sea network, the faults within each network could compromise the reliability of data [8] such that the quality of service (QoS) provided by B5G-SAGS is deteriorated. That suggest the importance of the reliable and efficient fault-diagnosis. Due to a significant amount of computation resources required in the fault-diagnosis, B5G-MEC services are required, i.e., the high performance computation is implemented on B5G-MEC servers. Fig. 1 gives the architecture of B5G-SAGS. In Fig. 1, space-air networks consist of satellites, aircrafts (e.g., plane, airship, helicopter and drone) and corresponding communication links, while the terrestrial cellular networks and the internet of vehicles comprise ground networks and the internet of vessels (i.e., the network of ships or buoys) constitutes the sea network respectively. Data collected by B5G-SAGS is analyzed and diagnosed on B5G-MEC servers for fault diagnosis.

Previous works on fault diagnosis [9]–[24] were designed without extra relevant information to improve diagnosis accuracy. Besides, in B5G-SAGS, the purpose of buoy deployment is to cover a certain area of the ocean such that it is less likely for two buoys to monitor the same location while drifting with the sea tide. That suggests the relevant data collection by drones is a reliable supplement for fault-diagnosis. In this paper, we propose an Intelligent Drone-assisted Fault-diagnosis Algorithm (IDFA) to detect faulty buoys utilizing B5G-MEC services. The details of our contribution are given as follows.

1) To improve the reliability of the B5G-SAGS, a number of fault diagnosis servers are deployed on B5G-MEC servers, the collective decision of which determines the fault status of each buoy instead of the poor decision made by only one fault diagnosis process. Each fault diagnosis process is implemented using a Cubature Kalman Filter based Radial Bias Function Neural Network (CKF-RBFNN) for fault diagnosis accuracy improvement.
2) To further improve the fault diagnosis accuracy, drones are assigned to collect relevant information on the proximity of faulty buoys. All data collected by drones are uploaded to the airship, which is responsible for the communications between target buoys and the satellite, to achieve drone-aided fault diagnosis, i.e., the data provided by drones are integrated with the one offered by buoys for fault diagnosis. Meanwhile, the data collection path of drones are planned utilizing Deep Deterministic Policy Gradient (DDPG) algorithm on B5G-MEC servers as well for energy efficiency.
3) The theoretical analysis and validation experiments indicate that: (i) the IDFA has a better diagnosis accuracy in single fault detection and multi-fault classification as well while compared with contemporary algorithms; (ii) the IDFA obtains a high aggregation ratio and a low energy cost.

II. Related Work

To ensure the QoS of B5G-SAGS, the efficient and reliable fault-diagnosis is required. Plenty of excellent works that focus on fault-diagnosis on non-linear systems have been proposed.

The majority of fault-diagnosis algorithms on non-linear systems are filter-based. Strong tracking Kalman filter is studied by Ge *et al.* in [9]. Dual estimation is applied to the particle filter by Daroogheh *et al.* [10] to verify states and time-varying parameters as well. In [11], a novel filter based on the unbiased finite impulse response is proposed, in which the mean square error is minimized under the unbiased constraint. In [12], the strong tracking filter is utilized to assess faulty status and meanwhile the logistic regression is used to detect the root fault. In [13], the residual signal is used to detect bias faults and the residual analysis is applied to fault isolation. In [14], the problem of misleading state estimation caused by particle impoverishment is solved for particle diversity improvement. Zhao *et al.* [15], develop an adaptive square-root cubature Kalman filter (CKF) to solve the problem of decline or divergence with a noise statistic estimator. An interactive residual fault detection mechanism is designed [16] to detect hard faults and soft faults. In [17], Bernoulli random variables are applied to dead-zone effect modeling. The short circuit faults and transient load are detected based on the time-domain uncensored Kalman filter [18]. In [19], the combination of majority voting and cut set are utilized for fault detection. In [20], Sharma *et al.* develop a spatio-temporal correlation based fault diagnosis algorithm. In [21], Pearson test is applied to locate faulty devices. In [22], three Sigma test is employed to detect fault. In [23], the return message is used to discover faulty devices. Bayes classification is employed by Lau *et al.* [24] to design a centralized diagnosis algorithm for diagnosis accuracy improvement. In [25], Lu *et al.* propose a fault diagnosis model for echo state network utilizing granular computing to improve the efficiency and accuracy of electronic equipment. In [27], Sun *et al.* propose a fault diagnosis strategy to detect actuator failures of autonomous underwater vehicles (AUV) utilizing the Gaussian particle filter. In this strategy, mathematical models are

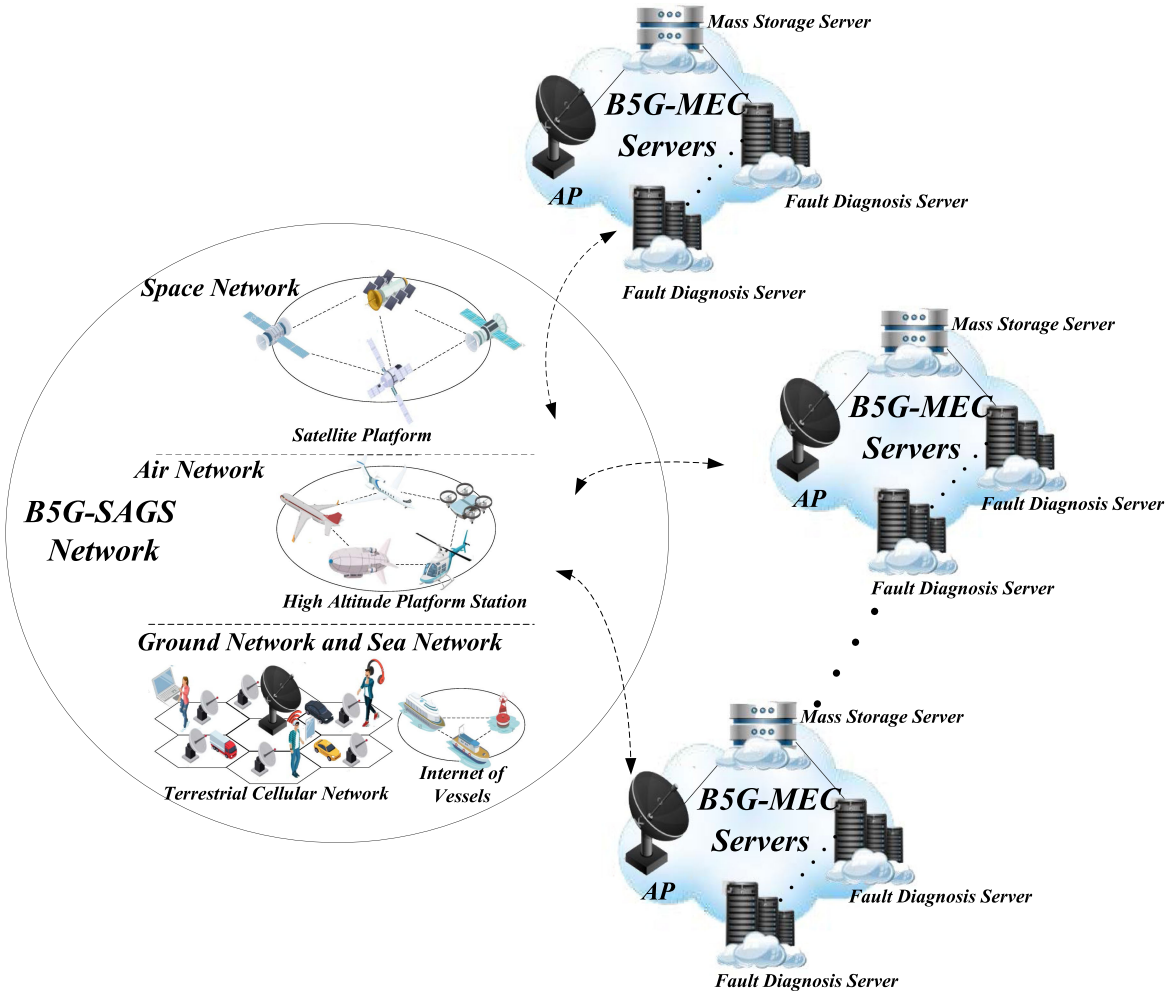

Fig. 1. Architecture of B5G-SAGS.

established for AUV motions first. Then, the control loss is used to measure the malfunction of AUVs. Eventually, the improved Gaussian particle filter is developed to detect AUV failures and motion states. In [26], a machine learning based fault-diagnosis algorithm is proposed by Qi *et al.* for reciprocating compressors. The data is sparse coded first for the dictionary training, based on which faults are classified by SVM.

Although these works contribute to improve the Qos in B5G-SAGS through fault-diagnosis, there remain two problems: (i) how to be energy efficient in extra relevant information collection; (ii) how to improve fault diagnosis accuracy with such information. In this paper, a novel Intelligent Drone-assisted Fault-diagnosis Algorithm (IDFA) is proposed to address these two problems.

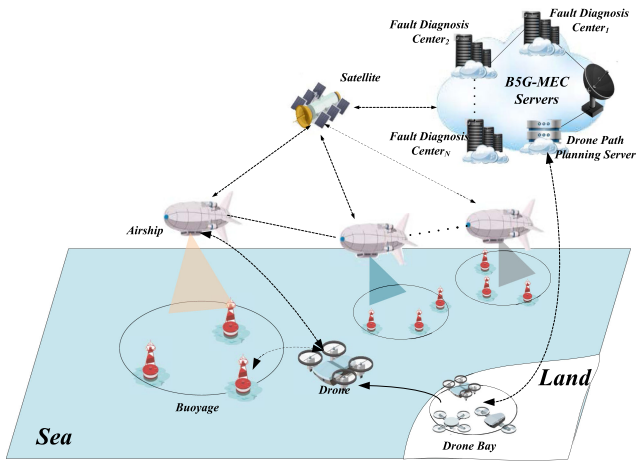

Fig. 2. System model of the IDFA in B5G-SAGS.

III. System Model

To maintain the quality of service (Qos) of B5G-SAGS, we design an Intelligent Drone-assisted Fault-diagnosis Algorithm (IDFA). Specifically, the data collected by buoys are uploaded to B5G-MEC servers via the airship-satellite channel. Then, the fault status of each buoy is verified by the collective decision of multi-fault diagnosis servers deployed on B5G-MEC servers with the extra information provided by cruising drones [28]. Therefore, we consider three entities in this model, i.e., the buoys, the drones, and the fault-diagnosis servers. Fig. 2 gives the system model.

1) *Buoys:* Usually, buoys are deployed to monitor a specific area of ocean for relevant information collection,

i.e, temperature, humidity, wind velocity, visit record of vessel, etc. All these data are uploaded via airship-satellite channel for oceanographic research. In addition, if buoys fall into the communication range of an airship, then they are governed by the airship. Due to the unpredictable changes of weather, i.e., tsunami, lighting and hurricane, or other nature disasters, i.e., undersea volcano eruption, buoys might be damaged or even destroyed. Thereby, it is important to monitor the operating state of each buoy.

2) *Fault-Diagnosis Servers:* Since buoys are suffering from nature disasters, the fault-diagnosis server is responsible to detect the faulty buoys based on the data collected by them. The fault-diagnosis server is deployed on B5G-MEC servers due to the high performance computation required [29]. Because the decision made by one fault-diagnosis server could be unreliable, a number N of fault-diagnosis servers are participating the decision making for diagnosis accuracy improvement.
3) *Drones:* For further improve the diagnosis accuracy, drones are sent to the proximity of the potential faulty buoy to collect relevant data. Due to the limited communication range of the drone, all data collected by the drone will directly sent to fault-diagnosis servers via the corresponding airship-satellite channel. The data collection path should be well designed for energy efficiency.

Note that the environmental surveillance of buoys is a complex nonlinear process. Therefore, constructing a definite model for fault diagnosis is difficult. On the other hand, as a feedforward neural network, the RBF neural network (RBFNN) has only one hidden layer. The nonlinear transformation is employed for the input layer and the hidden layer to generate the connection weights. The similar weights between the hidden layer and the output layer can be generated utilizing a linear transformation. In RBFNN, the problem of linear indiscernibility in low-dimensional space is solved by transforming the low-dimensional data to the high-dimensional ones. Therefore, the RBFNN is able to establish the fault diagnosis model for detecting either single-fault or multi-faults based on the relevant data collected by both buoys and drones.

IV. Implementation Details of the IDFA

The proposed IDFA consists of two important components, i.e., CKF-RBF based fault diagnosis and drone path planning.

A. CKF-RBF Based Fault Diagnosis

1) Fault-Diagnosis on Single B5G-MEC Server: To improve the fault-diagnosis accuracy, the Radial Bias Function Neural Network is applied to detect faulty buoys. The data required in the fault diagnosis will be collected from different devices, i.e., buoys and drones. Specifically, each buoy provides its own sensing data towards the fault diagnosis servers deployed on edges via airship-satellite channels. In addition, the drones are sent to the proximity of the potential faulty buoy to collect relevant data. All these data will be analyzed and labelled first by fault diagnosis servers, based on which the fault diagnosis is implemented simultaneously. Note that the collection of data originate from different devices, the corresponding fault diagnosis is more reliable, considering the case that a faulty buoy could provide unreliable data all the time. In addition, we consider several readings of a buoy as an input data for fault diagnosis. For example, each data $\mathbb{X}$ considered is presented in a vector, i.e., $\mathbb{X} = (tem_i, hum_i, pre_i)$, where tem_i, hum_i and pre_i represents the temperature, the humidity and the pressure of the area around the ith faulty buoy.

And the number of nodes of on each layer is set as follows. Specifically, for the single-fault detection, the structure of the neural network is designed as follows. Since the state of a buoy is either faulty or normal, only two outputs are required for each state. However, compared with the single-fault detection, the multi-fault detection should distinguish what kinds of faults each buoy suffers rather than determining the buoy is faulty or not. In this case, the output layer for the multi-fault detection should be able to generate the faulty state of each buoy in a $(n+1)$ dimension vector, where n denotes the number of faults considered in this paper. Apart from the output layer, the number of nodes on the input layer and the hidden layer for either the single-fault detection or the multi-fault detection are set to n and m, respectively. This is because we can label the data sample for single-fault detection to fit for multi-fault detection, i.e., labelling the data sample for a specific fault, instead of adding more input layers for multi-fault detection. In addition, the number of nodes on the hidden layer is chosen based on the fault diagnosis accuracy, i.e., a larger m results in a more accurate fault diagnosis. Thereby, there are totally n nodes on the input layer, m nodes on the hidden layer, and one node on the output layer for the single-fault detection, compared with n, m, and n nodes on each layer respectively.

To implement the fault diagnosis, we should find the proper parameters, i.e., weight ω_{ij}, center c_m and the width b_m, for a better fault diagnosis accuracy, where m denotes the number of hidden layer nodes. Thereby, we integrate the Cubature Kalman Filter (CKF) and RBFNN to discover the optimal value of above parameters. Firstly, the state vector X_k of CKF-RBFNN is given by

$$\mathcal{X}_k = [\mathcal{W}, \mathcal{C}, \mathcal{B}],$$

where $\mathcal{W} = (\omega_{11}, \ldots, \omega_{mn})$, $\mathcal{C} = (c_1, \ldots, c_m)$, and $\mathcal{B} = (b_1, \ldots, b_m)$. Accordingly, we obtain

$$\begin{cases} X_{k+1} = X_k + W_k, \\ Z_k \ \ = h(X_k, \mathbb{X}_k) + V_k \end{cases} \tag{1}$$

where the input of CKF-RBFNN (e.g., the data collected by both drone and buoy) is denoted by $\mathbb{X}_k$, while the measurement vector is represented by Z_k; V_k denotes the measurement noise with covariance R_k, while W_k denotes the process noise of covariance Q_k; $h(.)$ is the nonlinear mapping function from the input layer to the output layer, e.g., $h(X_k, \mathbb{X}_k) = \mathcal{W}_k \zeta_k$ with

$$\mathcal{W}_k = \begin{bmatrix} \omega_{11} & \cdots & \omega_{m1} \\ \omega_{12} & \cdots & \omega_{m2} \\ \vdots & \vdots & \vdots \\ \omega_{1\,l} & \cdots & \omega_{mn} \end{bmatrix}, \zeta_k = \begin{bmatrix} g(\mathbb{X}_k, c_1, b_1) \\ g(\mathbb{X}_k, c_2, b_2) \\ \vdots \\ g(\mathbb{X}_k, c_m, b_m) \end{bmatrix}$$

where $g(.)$ represents a nonlinear mapping. Specifically, we implement the nonlinear mapping with the Gaussian kernel function by mapping the finite-dimensional data to high-dimensional space, in which data is linearly separable.

In this paper, we apply the square root based updating rule for CKF to improve the stability. In addition, by introducing the strong tacking factors and the adaptation of both process noise and measurement noise, the stability of the training process of the CKF-RBFNN is further enhanced. Let $k = 0$ and initialize $\hat{x_{0|0}}$, $P_{0|0}$, Q_0, and R_0. Then, we give the details of a six-step training process for CKF-RBFNN as follows:

step 1: Employ the Choleskey decomposition on $P_{0|0}$ to update the initial value of the characteristic square root of the error covariance matrix as

$$S_{0|0} = chol(P_{0|0})^T; \tag{2}$$

step 2: $\forall k \in \{1, \ldots, m\}$, use the $S_{k-1|k-1}$ to estimate the volume point and calculate the volume point after propagation by

$$X_{i,k-1|k-1} = S_{k-1|k-1}\xi_i + \hat{x}_{k-1|k-1} \tag{3}$$

$$X^*_{i,k|k-1} = X_{i,k-1|k-1} \tag{4}$$

Then, estimate the state by

$$\hat{x}_{k|k-1} = \frac{1}{m}\sum_{i=1}^{m} X^*_{i,k|k-1}. \tag{5}$$

And estimate the prediction error covariance matrix characteristic square root by:

$$[A\ B] = qr([X^*_{k|k-1}\ S_{Q,k-1}]^T) \tag{6}$$

and

$$S_{k|k-1} = B(1:n,:)^T, \tag{7}$$

where

$$Q_{k-1} = S_{Q,k-1}S^T_{Q,k-1}, \tag{8}$$

$$\begin{aligned} X^*_{k|k-1} = \frac{1}{\sqrt{m}}[&X^*_{1,k|k-1} - \hat{x}_{k|k-1}X^*_{2,k|k-1} \\ &- \hat{x}_{k|k-1} \cdots X^*_{m,k|k-1} - \hat{x}_{k|k-1}]. \end{aligned} \tag{9}$$

Next, calculate the volume point and update the measurement equation to spread the volume point by

$$X_{i,k|k-1} = S_{k|k-1}\xi_i + \hat{x}_{k|k-1}, \tag{10}$$

where $\xi_i = \sqrt{\frac{m}{2}}[1]$, $i = 1, 2, \ldots, m$.

$$Z_{i,k|k-1} = h(X_{i,k|k-1}). \tag{11}$$

And, calculate the measured predicted value by

$$\hat{z}_{k|k-1} = \frac{1}{m}\sum_{i=1}^{m} Z_{i,k|k-1}. \tag{12}$$

Thereby, estimate the characteristic square root of the predicted output covariance matrix as

$$[A\ B] = qr([Z^*_{k|k-1}\ S_{R,k}]^T), \tag{13}$$

where

$$S_{zz,k|k-1} = B(1:l,:)^T, \tag{14}$$

$$R_k = S_{R,k}S^*_{R,k}, \tag{15}$$

$$\begin{aligned} Z^*_{k|k-1} = \frac{1}{\sqrt{m}}[&Z^*_{1,k|k-1} - \hat{z}_{k|k-1}Z^*_{2,k|k-1} \\ &- \hat{z}_{k|k-1} \cdots Z^*_{m,k|k-1} - \hat{z}_{k|k-1}]. \end{aligned} \tag{16}$$

Then, cross-covariance matrix is estimated by

$$P_{xz,k|k-1} = X_{k|k-1}(Z^*_{k|k-1})^T. \tag{17}$$

step 3: Calculate multiple strong tracking factors by

$$\lambda_k = \begin{cases} \lambda_k, & \lambda_k \geq 1 \\ 1, & \lambda_k < 1 \end{cases} \tag{18}$$

where

$$\lambda_k = \frac{tr(N_k)}{tr(M_k)}, \tag{19}$$

$$N_k = V_k - H_kQ_{k-1}H_k^T - \beta R_k, \tag{20}$$

$$M_k = H_kF_kP_kF_k^TH_k^T, \tag{21}$$

$$V_k \begin{cases} \varepsilon_1\varepsilon_1^T, & k = 1 \\ \frac{\rho V_{k-1} + \varepsilon_1\varepsilon_1^T}{1+\rho}, & k > 1 \end{cases} \tag{22}$$

and F_k is the state linearization matrix, H_k is measurement linearization matrix calculated by Jacobian matrix; $tr(.)$ is the trace calculation operation; ρ is the forgetting factor of the residual sequence $\varepsilon_k = z_k - \hat{z}_{k|k-1}$; V_k is the covariance matrix of the output residual sequence; β is the weakening factor chosen based on empirical values to avoid possible over-adjustment.

step 4: Replace 6 by the following equation

$$[A\ B] = qr([\sqrt{\lambda_k}X^*_{k|k-1}\ S_{Q,k-1}]^T) \tag{23}$$

where $qr(.)$ denotes the QR-decomposition, and repeats Step 2 to obtain the characteristic square root of the prediction error covariance matrix $S_{k|k-1}$ after introducing the strong tracking factor; Then, predict characteristic square root of the output error covariance matrix $S_{zz,k|k-1}$ and the cross-covariance matrix $P_{xz,k|k-1}$.

step 5: Obtain the optimal filter gain K_k, optimal prediction state $\hat{x}_{k|k}$ and optimal error covariance matrix characteristic square root $S_{k|k}$ by

$$K_k = \frac{P_{xz,k|k-1}}{S_{zz,k|k-1}S_{zz,k|k-1}^T}, \tag{24}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(z_k - \hat{z}_{k|k-1}), \tag{25}$$

$$[A\ B] = qr([X_{k|k-1} - K_k Z_{k|k-1}^*\ K_k S_{R,k}]^T), \tag{26}$$

$$S_{k|k} = B(1:n,:)^T. \tag{27}$$

step 6: $\forall k \geq 2$, both process noise and measurement noise are adapted by

$$Q_k = (1-d_k)Q_{k-1} + d_k[K_k\varepsilon_k\varepsilon_k^T K_k^T], \tag{28}$$

$$R_k = (1-d_k)R_{k-1} + d_k[\varepsilon_k\varepsilon_k^T], \tag{29}$$

where

$$d_k = \frac{1-b}{1-b^{k+1}}, 0 < b < 1. \tag{30}$$

and b represents the forgetting factor in the recursive formula of the noise statistical estimator, and its value selection should be considered as a trade-off between the algorithm's strong tracking performance and noise insensitivity.

Once the training of CKF-RBFNN is completed, it can be applied to either single-fault diagnosis or multi-fault diagnosis by fault diagnosis servers.

2) Fault-Diagnosis Based on the Collective Decision of Multiple B5G-MEC Servers: Once the training of CKF-RBFNN is completed, the faulty status of each buoy can be detected. In order to improve the diagnosis accuracy, we employ a number N of fault-diagnosis servers for joint decision-making. Specifically, for the single fault detection, the output Y of CKF-RBFNN is presented by

$$Y = \begin{cases} 1 & Faulty \\ 0 & Normal \end{cases}$$

For the multi-fault detection case, the output Y, which is a n dimension vector $Y = (y_1, y_2, \ldots, y_n)$, represents the specific faults detected. For example, if $y_i = 1$, $1 \leq i \leq m$, then the buoy suffers from up to m types of faults. Note that if the buoy is faulty-free, then we obtain $Y = (0, 0, \ldots, 0)$.

We denote the output of the ith B5G-MEC Server by Y^i. Then, the collective decision is made based on all N B5G-MEC servers, i.e., the buoy is faulty only if the average value of N outputs approaches 1 for the single-fault detection. That is,

$$\frac{1}{N}\sum_{i=1}^{N} Y^i \to 1. \tag{31}$$

Otherwise, the buoy is normal, i.e.,

$$\frac{1}{N}\sum_{i=1}^{N} Y^i \to 0. \tag{32}$$

Similarly, for the multi-fault detection, we let $Y^i = (Y_1^i, Y_2^i, \ldots, Y_n^i)$ denote the output of the ith B5G-MEC Server. Then, the jth type of fault of the buoy is diagnosed only if

$$\frac{1}{N}\sum_{i=1}^{N} Y_j^i \to 1. \tag{33}$$

Otherwise, the buoy is jth fault-free, i.e.,

$$\frac{1}{N}\sum_{i=1}^{N} Y_j^i \to 0. \tag{34}$$

Note that the single decision made by the only one B5G-MEC server is unreliable. However, if the faulty state of a buoy is made based on the collective decision by several B5G-MEC servers, then it is more reliable. Furthermore, we allow all N B5G-MEC servers to run CKF-RBFNN based fault-diagnosis algorithm simultaneously, although there might exist differences between models trained by different B5G-MEC servers. Besides, the collective decision is made by taking the average of all decisions of each B5G-MEC servers. That suggests such decision making is more fair and more reasonable. Thereby, the proposed IDFA is applicable to improve the QoS of the B5G-SAGS.

B. Drone Path Planning

How to design a reasonable drone path poses a great challenge. This is because the drone should arrive the proximity of the fault buoys as soon as possible for relevant data collection. However, due to the weather condition and other environment interferences, an irrational path could result in a tremendous energy cost.

To solve this problem, we develop a deep reinforcement learning (DRL) based drone path planning, which is implemented on B5G-MEC servers. Note that DRL is built on the structure of deep learning [30] w.r.t the principle of reinforcement learning to solve decision-making problems; it utilizes the representation ability of the neural network to fit the Q-table or the strategy to solve the problem of excessive state-action space or continuous state-action space, i.e., discovering the control on the movement of drones. In this paper, as a DRL, the Deep Deterministic Policy Gradient (DDPG) is utilized to design the energy efficient drone path. In general, a DDPG decision system consists of a critic network Q, a target

critic network Q', an actor networks π and the corresponding target actor networks π, the parameters of which are denoted by ϑ^{Q}, $\vartheta^{Q'}$, ϑ^{π} and $\vartheta^{\pi'}$, respectively.

As a DRL, the action is given at a state to obtain the reward and then the next state is observed from the environment. And the experience is stored in an experience pool $\mathcal{P}$ in the form of a transition of a quadruple (s_t, a_t, r_t, s_{t+1}). In the drone path planning, we consider the position of the drone as a state s. The action, i.e., hovering or moving toward any direction, is represented by a, while the reward is denoted by r. At each state s, the drone decides to hover for data collection or to move toward the faulty buoy, based on which the reward can be calculated. In addition, the amount of data collected should be considered. Note that within each time interval a drone might collect no data. For example, while a drone is still approaching the communication coverage area of the faulty buoy, there is no data required to be collected. In this case, the drone receives a negative reward r_t in timeslot t; otherwise, the reward r_t is calculated as a tradeoff function based on the state-action pair (s_t, a_t) as

$$r_t = \begin{cases} -\frac{1}{Energy_{i,t}}, & if\ Data_{i,t} = 0 \\ \frac{Data_{i,t}}{Energy_{i,t}}, & otherwise \end{cases} \tag{35}$$

where $Data_{i,t}$ and $Energy_{i,t}$ represents the amount of data collected and the energy consumed by timeslot t.

It is worth to mention that the energy cost consists of two parts, i.e., the energy consumed by both receiving and transmitting data, and the energy required within the data collection path that depends on the velocity and different flight stages including acceleration, deceleration, hovering, and turning. We use the energy model proposed by Ding *et al.* [31] to quantify the UAV energy consumption. If the drone is flying in a straight line at different speeds, i.e., 2 m/s, 4 m/s, 6 m/s and 8 m/s, then the power consumption will be 242 W, 245 W, 246 W, and 268 W, respectively. Obviously, the power consumption increases as the velocity and vice versa while rotating at angles with the increment of 45°. For example, the power consumption $\mathcal{P}_{turn}$ equal to 260 W when angular speed v_{turn} reaches 2.07 rd/sec. Thereby, the energy consumption during turn $Energy_{turn}$ is given by

$$Energy_{turn} = \Lambda_{\theta} \mathcal{P}_{turn} / v_{turn}, \tag{36}$$

where $\mathcal{P}_{turn}$ denotes the power consumption, Λ_{θ} denotes turning angle, v_{turn} represents angular velocity while turning. Then the energy cost $Energy_{v,d}$ is calculated by

$$Energy_{v,d} = \int \mathcal{P}_{acc} dt + \int \mathcal{P}_{v} + \int \mathcal{P}_{dec} dt, \tag{37}$$

where acceleration power consumption is denoted by $\mathcal{P}_{acc}$, deceleration power consumption is denoted by $\mathcal{P}_{dec}$, power consumption while flying at uniform velocity is denoted by $\mathcal{P}_{v}$, and the travel distance is represented by d. On the other hand, the energy cost for communication between drones and buoys consists of that of receiving $Energy_{rec}$ and transmitting $Energy_{tra}$ of data, which is given by

$$Energy_{tra} = \begin{cases} p_s \times (Energy_c + Energy_s \times d^2), if\ d_0 > d \\ p_s \times (Energy_c + Energy_l \times d^4), otherwise \end{cases} \tag{38}$$

where $Energy_{rec} = p_s \times Energy_c$, the energy dissipation on the circuitry of receiver or transmitter is denoted by $Energy_c$, the packet size is denoted by p_s, d_0 is a predetermined threshold, $Energy_s$ and $Energy_l$ denote the energy cost on amplifier for short or long distance transmission of one bit, respectively. Thereby, for the ith drone, the energy cost $Energy_{i,t}$ on time slot t is then calculated by

$$\begin{aligned} Energy_{i,t} = \ & Energy_{tra} + Energy_{rec} \\ & + Energy_{turn} + Energy_{v,d}. \end{aligned} \tag{39}$$

Note that the goal of DDPG is to find the optimal action a_t for each state s_t in order to maximize the reward r_t. Therefore, we choose the one of the maximal Q value, which is

$$a_t = \arg\max_{a_t \in A} Q(s_t, a_t), \tag{40}$$

where $A = \{a_t\}$. Then, experience (s_t, a_t, r_t, s_{t+1}) is stored in the experience poor $\mathcal{P}$.

In the training process, we sample N experience from $\mathcal{P}$ to update the critic network utilizing the following loss function

$$\mathcal{L}(\vartheta^{Q}) = \frac{1}{N} \sum_{i}^{N} [Q(s_i, a_i | \vartheta^{Q}) - \mathcal{Y}_i]^2, \tag{41}$$

where

$$\mathcal{Y}_i = r_i + \gamma (Q(s_{i+1}, \pi(s_{i+1} | \vartheta^{\pi'}) | \vartheta^{Q'})). \tag{42}$$

Note that γ denotes the learning rate, which should be chosen within the range of [0,1] for the stability improvement of the learning process. Accordingly, we update π utilizing policy gradient as

$$\begin{aligned} \nabla_{\vartheta^{\pi}} J = \frac{1}{N} \sum_{i}^{N} [& \nabla_a Q(s, a | \vartheta^{Q}) | s = s_i, a = \pi(s_i | \vartheta^{\pi}) \\ & \nabla_{\vartheta^{\pi}} \pi(s | \vartheta^{\pi}) | s = s_i]. \end{aligned} \tag{43}$$

Target networks are copies of the actor π and critic Q networks of different update rules. Once networks πs and Q are updated, we then update the parameters of target networks $\vartheta^{Q'}$ and $\vartheta^{\pi'}$ with a learning rate κ

$$\vartheta^{Q'} = \kappa \vartheta^{Q} + (1 - \kappa) \vartheta^{Q'}, \tag{44}$$

$$\vartheta^{\pi'} = \kappa \vartheta^{\pi} + (1 - \kappa) \vartheta^{\pi'}, \tag{45}$$

The accuracy of fault diagnosis mainly relies on the information collected by drones. However, the proposed path planning for drones aims to achieve energy-efficiency during the data collection for the single faulty buoy case. That suggests how to detect multi faulty buoys is an open problem. We distinguish two cases to solve this problem.

Algorithm 1. Drone Path Planning

Input: sensing areas of all buoys $sensingarea_b$s, the sensing area of the drone $sensingarea_d$
for all $i = 1, n$ **do**
  if $sensing_area_{b,i}$s are overlapping**then**
    the location $centroid\_sensingarea_b$ is set as the destination of the drone path planning
  else
    if $sensingarea_d$ is overlapping with $sensingarea_b$s**then**
      $centroid\_sensingarea_d$ is set as the destination of the drone path planning
    else
      each faulty buoy will be sent a drone
    end if
  end if
end for
drone are send to collect relevant data along the paths established by Algorithm 2

Algorithm 2. DDPG based Drone Path Design

Input: Initialized s_0, Action Set A, Experience Poor $\mathcal{P}$, Sample Number N, Actor Networks π and π', Critic Networks Q and Q'
Output: Drone paths
for $t = 1, T$ **do**
  select action via Eq. (40)
  execute action a_t, calculate reward r_t via Eq. (35) and observe next state s_{t+1}
  store (s_t, a_t, r_t, s_{t+1}) in experience poor $\mathcal{P}$
  randomly sample N experiences from experience poor $\mathcal{P}$
  update critic network via Eq. (41) and Eq. (42)
  update actor network via Eq. (43)
  update target networks via Eq. (44) and Eq. (45)
end for

Case 1: If the sensing areas $sensingarea_b$s of two or more buoys are overlapping, then only one drone will be assigned to the position around the centroid of the overlapping area $centroid\_sensingarea_b$ to collect relevant data for both faulty buoys. That suggests the destination for a drone is set to one location rather than two separated ones. And the data collected will be labelled individually and sent to the fault diagnosis servers via the airship-satellite channels.

Case 2: For the case that the sensing areas of buoys are non-overlapping, we check if there exists a position that the sensing area of the drone $sensingarea_d$ is overlapping with that $sensingarea_b$s of buoys. If the location (i.e., the centroid of the sensing area of the drone $centroid\_sensingarea_d$) does exist, then the drone is assigned to this location; otherwise, each faulty buoy will be sent a drone for relevant data collection.

We then summarize the drone path planning in Algorithm 1.

V. Performance Evaluation

A. Simulation Setup

The simulation is conducted to validate the performance of the proposed IDFA in Python on 7 computers, each of which is equipped with an Intel Core i7 processor, a 64 G running memory, and a CPU frequency 6.4GHZ 64-bit win7 system. We let each computer represent a fault diagnosis edge server. In addition, we assume each quadrotor is equipped with a 2.56 GHz processor of 64-bit quad-core and a Li-Po battery of 3300 mAH. Table I gives the rest parameters of this simulation.

TABLE I
Simulation Setup

Parameter	*Range*
Area of experiment	1000*1000 m^2
Number of fault diagnosis servers N_f	[1,7]
Speed of drone V	[1,8] m/s
Rotating angleΛ_θ	[45°, 90°, 135°, 180°]
Number of buoys N_b	[50,400]
Number of drones N_d	[2,16]
Energy level of drone at Start Time	100 Watt Hour
Sensitivity of receiver	-100 dBm
Bit Rate	120 kbps
Frequency of transmission	2.45 GHZ

1) Performance Index and Baseline Approaches: The performance of IDFA is validated in test error, multi-fault classification, aggregation ratio and energy cost. We then give the performance indexes as follows.

1) *Test Error:* To verify the faulty status of a buoy, the test error is adopted, i.e, a correct diagnosis should have a test error less than 0.5, while a test error greater than 0.5 indicates the incorrect one.
2) *Multi-fault Classification:* Due to the inhospitable working environment, a buoy could suffers from various faults. That means the fault-diagnosis mechanism should be capable of distinguishing a particular fault from others.
3) *Aggregation Ratio:* Since the extra information provided by drones helps to improve the fault-diagnosis accuracy, the relevant data should not be collected. That suggests a higher aggregation ratio might result in a better fault-diagnosis accuracy.
4) *Energy Cost:* The energy cost of fault-diagnosis results from data collection by drones. The less energy cost indicate the better performance of the fault-diagnosis system. In this experiment, we normalize the energy cost for better performance comparison.

The proposed IDFA first compares with baseline approaches, i.e, RBF, PSO-SVM and UKF-RBF [32], in test error and multi-fault classification, respectively. The reason for choosing these algorithms is as follows. In fact, RBF neural network can establish a mapping between input information and output information, which is developed w.r.t the function and structure of human brain. The approximation ability, classification and recognition ability, and learning speed of RBF neural network are better than that of BP neural network. Therefore, it is applicable for fault diagnosis of buoys in extreme environments. On the other hand, the CKF is based on the point estimation. That suggests there is no need to linearize the nonlinear system. Thereby, the CKF can improve the fitting accuracy of the nonlinear function and avoiding the numerical instability of UKF in the high-dimensional system. In general, the CKF has a better performance in applicability and stability, especially for high-dimensional nonlinear systems, while compared with

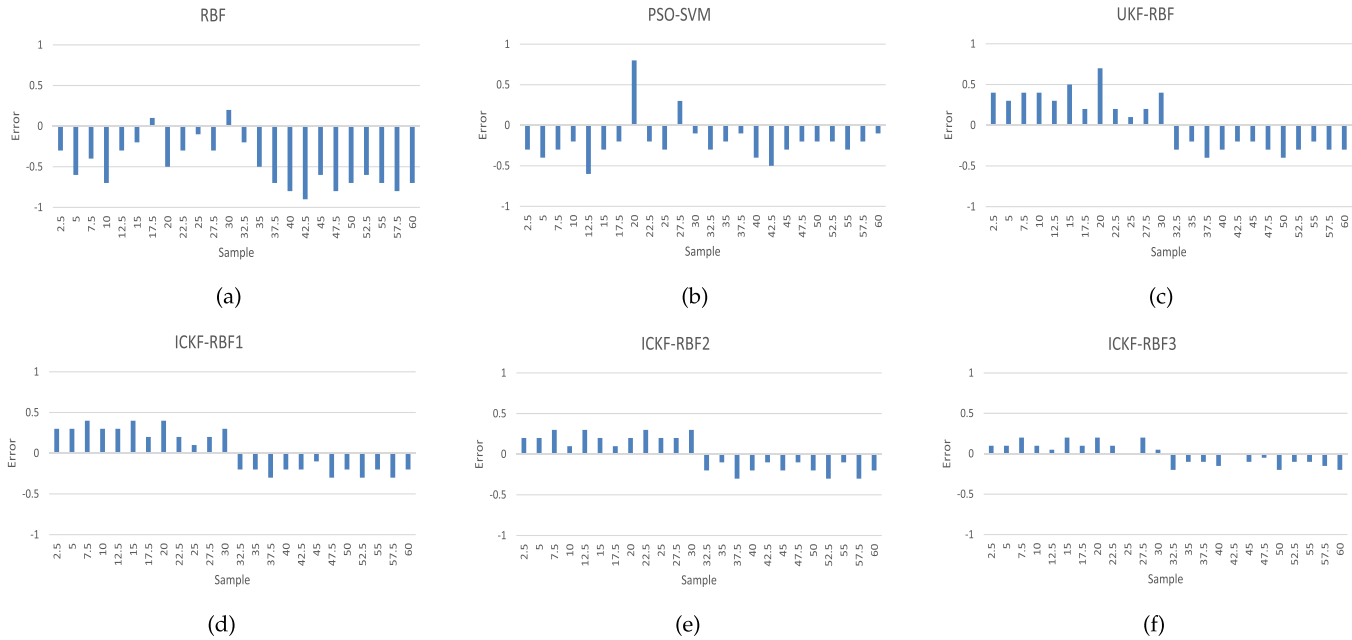

Fig. 3. Test error comparison in statistic between RBF, PSO-SVM, UKF-RBF and CKF-RBF.

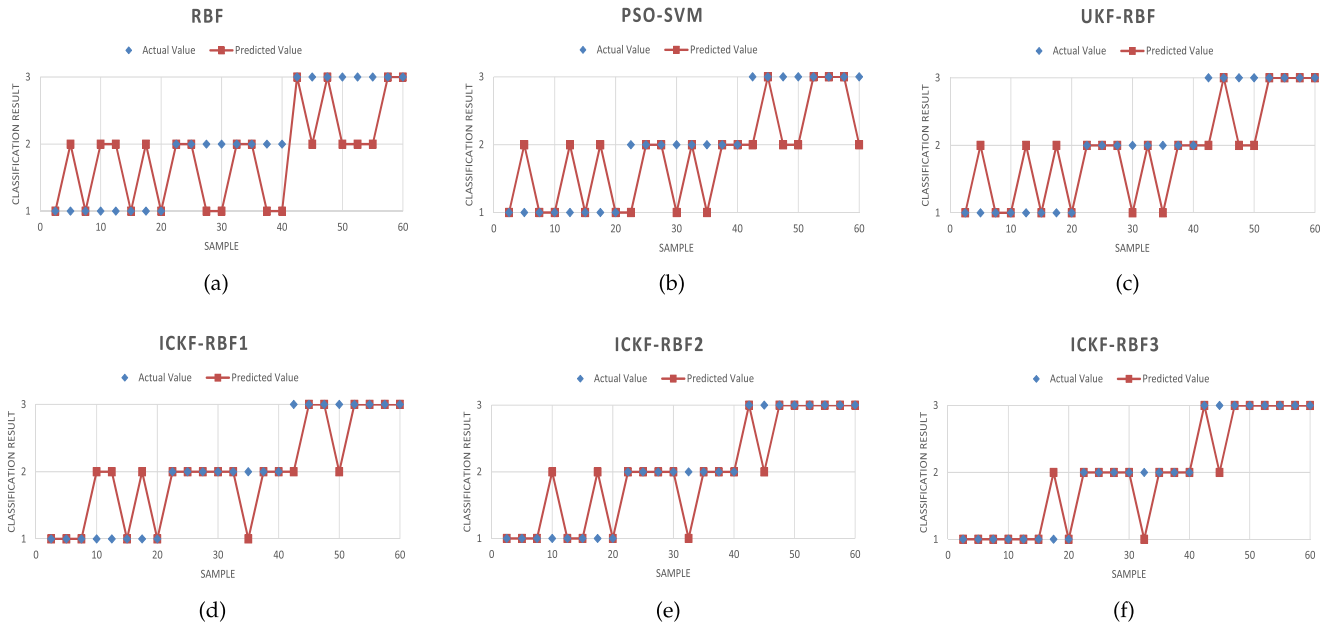

Fig. 4. Multi-fault classification comparison between RBF, PSO-SVM, UKF-RBF and CKF-RBF.

TABLE II
AVERAGE TEST ERROR FOR SINGLE-FAULT DIAGNOSIS

Type \ N_f	4	5	6	7
Single-fault	0.27	0.22	0.13	0.04

TABLE III
MULTI-FAULT CLASSIFICATION ACCURACY

Indexes \ N_f	4	5	6	7
FAR	14%	13%	8%	4%
MDR	10%	9%	7%	3%

UKF. Similar to CKF and UKF, the PSO-SVM can achieve the fault classification, however the accuracy of which is less than that of either UKF or CKF. Then, the drone path planning of the IDFA is compared with baselines, e.g., RTR [33] and ECMC [34], in aggregation ratio and energy cost. The RTR solves the round trip routing problem by locating the shortest path, considering the constrains of both delivery deadline and energy budget. The ECMC achieves 3D continuous movement control to maximize the energy efficiency in communication coverage for drone-cell networks and meanwhile the network connectivity is preserved. Since either RTR or ECMC aims to discover

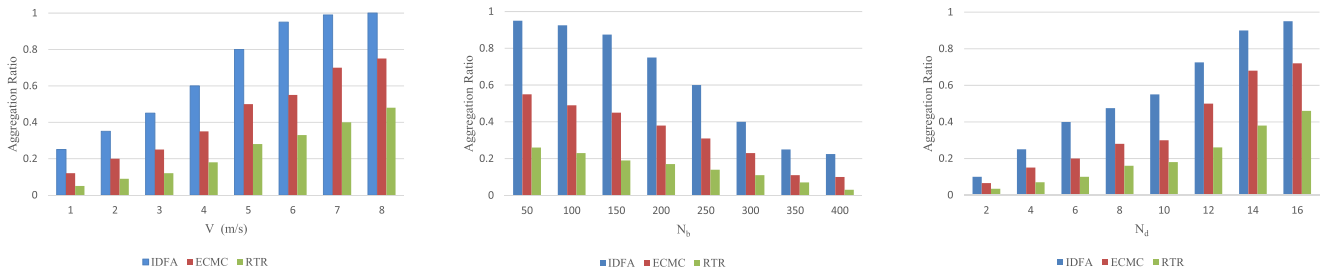

Fig. 5. Aggregation ratio while varying (a) v, (b) N_b, and (c) N_d.

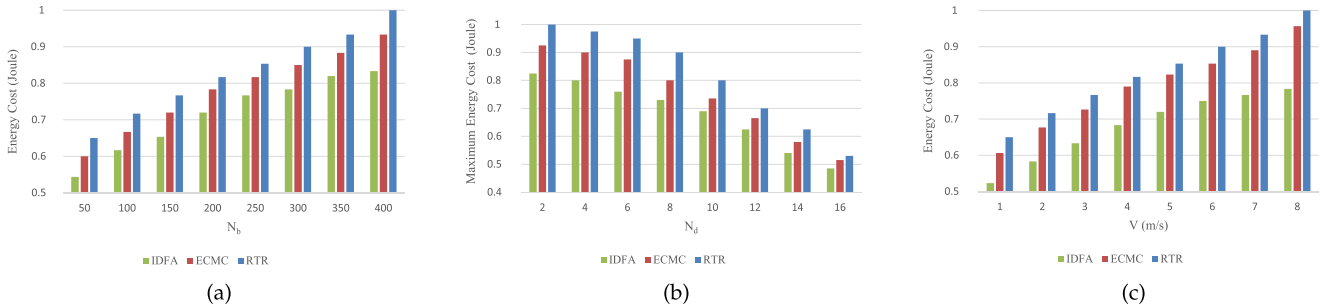

Fig. 6. Energy Cost while varying (a) N_b, (b) N_d, and (c) V.

the optimal energy-efficient drone paths, they are chosen as baselines for aggregation ratio and energy cost comparisons.

B. Experiment Results

1) Diagnosis Accuracy: Fig. 3 gives statistical test error comparison between RBF, PSO-SVM, UKF-RBF and CKF-RBF. We denote the number i of fault-diagnosis servers involved in fault detection as CKF-RBF$_i$. In addition, we assume that a correct diagnosis should be of a test error less than 0.5. Observed from Fig. 3, we know that with the number of fault-diagnosis servers grows the test error drops. Compared with RBF and PSO-SVM, CKF-RBF has a better diagnosis accuracy. For example, all test errors of CKF-RBF are beneath the threshold 0.5. The maximum false alarm rate is achieved by RBF. Fig. 3 suggests the proposed IDFA is more efficient in decreasing the false alarm rate with more fault-diagnosis servers involved.The multi-fault classification comparison is shown in Fig. 4. There are three fault states considered for effectiveness validation in classifying multi-faults among RBF, PSO-SVM, UKF-RBF and CKF-RBF. Observed from Fig. 4, we know that the classification accuracy increases when integrating UKF and RBF. It is obvious that UKF-RBF achieves a better classification accuracy while compared with PSO-SVM. Although all baselines perform well in multi-fault classification, CKF-RBF has the highest classification accuracy. Furthermore, with the number of fault-diagnosis servers grows, the classification accuracy further improves. Fig. 4 verifies the effectiveness of the proposed IDFA in multi-fault classification.

We then give the diagnosis accuracy analyses for both single-fault diagnosis and multi-fault diagnosis in Table IIand Table III with the number of fault diagnosis servers N_f varying from 4 to 7. Observed from Table II, we find that the test error decreases as N_f increases. The least test error is only 0.04, when the number of fault diagnosis servers reaches 7. That means the proposed IDFA can accurately detect faulty buoys. As shown in Table III, it is clear that either false alarm rate (FAR) or miss detection rate (MDR) is less than 15%. In addition, with the growth of N_f, both FAR and MDR drop. When $N_f = 7$, the proposed IDFA achieves the lowest FAR and MDR, which are only 4% and 3% respectively. Table II and Table III verify the advantages of IDFA in fault diagnosis accuracy.

2) Aggregation Ratio: Fig. 5 gives the aggregation ratio comparison with different v, N_b, and N_d respectively. Observed from Fig. 5 (a), we know that with the growth of v the aggregation ratio increases rapidly. Since the faster the drone flies, the more data is collected that results in a higher aggregation ratio. However, IDFA achieves the highest aggregation ratio due to the DRL based drone path planning. As shown in 5 (b), the aggregation ratio decreases as the N_b increases. This is because more buoys deployed results in more data required to be collected such that the aggregation ratio drops. Compared with baselines, IDFA is more efficient in data collection. The aggregation ratio sustainably grows with the N_d as shown in Fig. 5 (c). When there are sufficient drones arrived at the faulty buoys, all data will be collected such that the aggregation ratio should approach 100% for each approach. Thanks to the DRL based drone path planning, IDFA performs better than baselines with the highest aggregation ratio.

3) Energy Cost: Fig. 6 shows the impacts of N_b, N_d and V on both energy cost and maximum energy cost, respectively. In Fig. 6 (a), it is obviously that the growth of N_b results in the increment of energy cost for each approach. And IDFA consumes less energy cost than baselines due to the deep reinforcement learning algorithm DDPG employed. Observed from Fig. 6 (b), we find that the maximum energy cost decreases as N_d increases. The IDFA outperforms baseline approaches in the

least maximum energy cost. As shown in Fig. 6 (c), it is clear that as the speed V increases the energy cost rises rapidly. However, the proposed IDFA consumes the least energy cost while compared with ECMC and RTR. Fig. 6 verifies the advantages of IDFA in energy cost while performing data collections.

VI. Conclusion

The quality of service of B5G-SAGS depends on the faulty status of each device within. That indicates the importance of fault-diagnosis in the B5G-SAGS design. Previous works on fault-diagnosis didn't integrate extra information to improve the diagnosis accuracy. In this paper, we propose an Intelligent Drone-assisted Fault-diagnosis Algorithm (IDFA) utilizing B5G-MEC services to detect faulty buoys. Specifically, IFDA first employs a Cubature Kalman Filter based Radial Bias Function Neural Network for each fault-diagnosis server to perform preliminary fault detection based on the data provided by both buoys and drones. The data collection path is planned utilizing the deep reinforcement learning algorithm, Deep Deterministic Policy Gradient, on B5G-MEC servers for energy efficiency. Eventually, the collective decision made by all fault-diagnosis centers determines the faulty status of each buoy. The theoretical analysis and validation experiments show that: (i) the IDFA has a better diagnosis accuracy in both single fault detection and multi-fault classification while compared with contemporary algorithms; (ii) the IDFA obtains a high aggregation ratio and a low energy cost.

References

[1] B. Zong, C. Fan, X. Wang, X. Duan, B. Wang, and J. Wang, "6G technologies: Key drivers, core requirements, system architectures, and enabling technologies," *IEEE Veh. Technol. Mag.*, vol. 14, no. 3, pp. 18–27, Sep. 2019.

[2] M. Jia, X. Zhang, J. Sun, X. Gu, and Q. Guo, "Intelligent resource management for satellite and terrestrial spectrum shared networking toward B5G," *IEEE Wireless Commun.*, vol. 27, no. 1, pp. 54–61, Feb. 2020.

[3] S. Sekander, H. Tabassum, and E. Hossain, "Multi-tier drone architecture for 5G/B5G cellular networks: Challenges, trends, and prospects," *IEEE Commun. Mag.*, vol. 56, no. 3, pp. 96–103, Mar. 2018.

[4] J. Qiu, D. Grace, G. Ding, M. D. Zakaria, and Q. Wu, "Air-ground heterogeneous networks for 5G and beyond via integrating high and low altitude platforms," *IEEE Wireless Commun.*, vol. 26, no. 6, pp. 140–148, Dec. 2019.

[5] M. Gheisarnejad, M. H. Khooban, and T. Dragičević, "The future 5G network-based secondary load frequency control in shipboard microgrids," *IEEE J. Emerg. Sel. Top. Power Electron.*, vol. 8, no. 1, pp. 836–844, Mar. 2020.

[6] T. X. Tran, A. Hajisami, P. Pandey, and D. Pompili, "Collaborative mobile edge computing in 5G networks: New paradigms, scenarios, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 4, pp. 54–61, Apr. 2017.

[7] L. Zhao, K. Yang, Z. Tan, X. Li, S. Sharma, and Z. Liu, "A novel cost optimization strategy for sdn-enabled UAV-assisted vehicular computation offloading," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3664–3674, Jun. 2021.

[8] J. Hu, H. Lin, X. Guo, and J. Yang, "DTCS: An integrated strategy for enhancing data trustworthiness in mobile crowdsourcing," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4663–4671, Dec. 2018.

[9] Q. Ge, T. Shao, S. Chen, and C. Wen, "Carrier tracking estimation analysis by using the extended strong tracking filtering," *IEEE Trans. Ind. Electron.*, vol. 64, no. 2, pp. 1415–1424, Feb. 2017.

[10] N. Daroogheh, N. Meskin, and K. Khorasani, "A dual particle filter-based fault diagnosis scheme for nonlinear systems," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 4, pp. 1317–1334, Jul. 2018.

[11] S. Zhao, Y. Shmaliy, and F. Liu, "Fast kalman-like optimal unbiased fir filtering with applications," *IEEE Trans. Signal Process.*, vol. 64, no. 9, pp. 2284–2297, May 2016.

[12] J. Ma, C. Lu, and H. Liu, "Fault diagnosis for the heat exchanger of the aircraft environmental control system based on the strong tracking filter," *PLoS One*, vol. 10, no. 3, pp. 294–299, 2015.

[13] X. He, Z. Wang, Y. Liu, and L. Qin, "Fault tolerant control for an internet-based three-tank system: Accommodation to sensor bias faults," *IEEE Trans. Ind. Electron.*, vol. 64, no. 3, pp. 2266–2275, Mar. 2017.

[14] S. Yin, and X. Zhu, "Intelligent particle filter and its application on fault detection of nonlinear system," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3852–3861, Jun. 2015.

[15] L. Zhao, J. Wang, T. Yu, H. Jian, and T. Liu, "Design of adaptive robust square-root cubature Kalman filter with noise statistic estimator," *Appl. Math. Comput.*, vol. 256, pp. 352–367, 2015.

[16] L. Zhang, Y. Cui, Z. Xiong, J. Liu, J. Lai, and P. Lv, "Research on adaptive multi-source information fault-tolerant navigation method based on no-reference system diagnosis," *Sensors*, vol. 19, no. 13, p. 2911, 2019.

[17] J. Huang, and X. He, "Detection of intermittent fault for discrete-time systems with output dead-zone: A variant tobit kalman filtering approach," *J. Control Sci. Eng.*, vol. 2017, 2017. Art no. 7849841.

[18] R. Cisneros-Magaña, A. Medina, and O. Anaya-Lara, "Time-domain voltage sag state estimation based on the unscented Kalman filter for power systems with nonlinear components," *Energies*, vol. 11, no. 6, p. 1411, 2018.

[19] R. R. Swain, T. Dash, and P. M. Khilar, "An effective graph-theoretic approach towards simultaneous detection of fault(s) and cut(s) in wireless sensor networks," *Int. J. Commun. Syst.*, vol. 30, no. 13, 2017, Art. no. e3273.

[20] K. P. Sharma, and T. P. Sharma, "rDFD: Reactive distributed fault detection in wireless sensor networks," *Wireless Netw.*, vol. 23, no. 4, pp. 1145–1160, 2017.

[21] M. Nandi, A. Nayak, B. Roy, and S. Sarkar, "Hypothesis testing and decision theoretic approach for fault detection in wireless sensor networks," *Int. J. Parallel, Emergent Distrib. Syst.*, vol. 30, no. 4, pp. 262–285, 2015.

[22] M. Panda, and P. M. Khilar, "Distributed self fault diagnosis algorithm for large scale wireless sensor networks using modified three sigma edit test," *Ad Hoc Netw.*, vol. 25, pp. 170–184, 2015.

[23] W. Gong, K. Liu, and Y. Liu, "Directional diagnosis for wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 5, pp. 1290–1300, May 2015.

[24] B. C. P. Lau, E. W. M. Ma, and T. W. S. Chow, "Probabilistic fault detector for wireless sensor network," *Expert Syst. Appl.*, vol. 41, no. 8, pp. 3703–3711, 2014.

[25] C. Lu, P. Xu, and L. H. Cong, "Fault diagnosis model based on granular computing and echo state network," *Eng. Appl. Artif. Intell.* vol. 94, 2020, Art. no. 103694.

[26] G. Qi, Z. Zhu, K. Erqinhu, Y. Chen, Y. Chai, and J. Sun, "Fault-diagnosis for reciprocating compressors using big data and machine learning," *Simul. Modelling Pract. Theory*, vol. 80, pp. 104–127, 2018.

[27] Y. S. Sun, X. R. Ran, Y. M. Li, G. C. Zhang, and Y. H. Zhang, "Thruster fault diagnosis method based on Gaussian particle filter for autonomous underwater vehicles," *Int. J. Nav. Architecture Ocean Eng.*, vol. 8, no. 3, pp. 243–251, 2016.

[28] N. Lin, L. Fu, L. Zhao, G. Min, A. Al-Dubai and H. Gacanin, "A novel multimodal collaborative drone-assisted VANET networking model," *IEEE Trans. Wireless Commun.*, vol. 19, no. 7, pp. 4919–4933, Jul. 2020.

[29] J. Mills, J. Hu, and G. Min, "Communication-efficient federated learning for wireless edge intelligence in IoT," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5986–5994, Jul. 2020.

[30] Z. Chen, J. Hu, G.Min, A. Zomaya, and T. El-Ghazawi, "Towards accurate prediction for high-dimensional and highly-variable cloud workloads with deep learning," *IEEE Trans. Parallel Distrib. Syst.*, vol. 31, no. 4, pp. 923–934, Apr. 2020.

[31] L. Ding, D. Zhao, H. Ma, H. Wang, and L. Liu, "Energy-efficient min-max planning of heterogeneous tasks with multiple UAVs," in *Proc. IEEE 24th Int. Conf. Parallel Distrib. Syst.*, 2018, pp. 339–346.

[32] W. Zhou, X. Li, J. Yi, and H. He, "A novel UKF-RBF method based on adaptive noise factor for fault diagnosis in pumping unit," *IEEE Trans. Ind. Inform.*, vol. 15, no. 3, pp. 1415–1424, Mar. 2019.

[33] H. Huang, A. V. Savkin, and C. Huang, "Round trip routing for energy-efficient drone delivery based on a public transportation network," *IEEE Trans. Transp. Electrific.*, vol. 6, no. 3, pp. 1368–1376, Sep. 2020.

[34] P. Yang, X. Cao, X. Xi, W. Du, Z. Xiao, and D. Wu, "Three-dimensional continuous movement control of drone cells for energy-efficient communication coverage," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 6535–6546, Jul. 2019.

Xiaoding Wang received the Ph.D. degree in 2016 from the College of Mathematics and Informatics, Fujian Normal University, Fuzhou, China, where he is currently an Associate Professor. His main research interests include network optimization and fault tolerance.

Hui Lin received the Ph.D. degree in computing system architecture from the College of Computer Science, Xidian University, Xi'an, China, in 2013. He is currently a Professor with the College of Mathematics and Informatics, Fujian Normal University, Fuzhou, China. He is also an M.E. Supervisor with the College of Mathematics and Informatics, Fujian Normal University. He has authored or coauthored more than 50 papers in international journals and conferences. His research interests include mobile cloud computing systems, blockchain, and network security.

Hongyan Zhang received the master's degree in 2008 from the College of Mathematics and Informatics, Fujian Normal University, Fuzhou, China, where she is currently working toward the Ph.D. She is also a Lecturer with the College of Concord University College, Fujian Normal University. Her main research interests include network optimization and privacy protection.

Dejun Miao received the B.Sc. degree in electrical engineering from the Nanjing University of Science and Technology, Nanjing, China, in 2001, the M.Sc. degree in microelectronic system design from the University of Southampton, Southampton, U.K., in 2005, and the Ph.D. degree in computing science from the University of Derby, Derby, U.K., in 2018. He is currently an Associate Professor with the School of Automation and Automobile, Yangzhou Polytechnic College, Yangzhou, China, and a Visiting Scholar with the School of Informatics, University of Leicester, Leicester, U.K. His research interests include Internet of Things, fog computing, service computing, and P2P computing.

Qinyang Miao received the bachelor's degree in information security from Xi'an University of Posts and telecommunications, Xi'an, China, in 2019. He is currently working toward the the master's degree in cyberspace security from the School of Mathematics and Information, Fujian Normal University, Fuzhou, China. His research interests and directions include deep learning, intensive learning, and network security.

Wenxin Liu received the bachelor's degree in information security from the Xi'an University of Posts and Telecommunications, Xi'an, China, in 2019. He is currently working toward the master's degree in cyberspace security from the School of Mathematics and Information, Fujian Normal University, Fuzhou, Chin. His research interests include edge computing, deep learning, and data security.