IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 14, 2019, accepted December 19, 2019, date of publication December 25, 2019, date of current version January 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2962130

Intelligent Resource Collaboration in Mobile Target Tracking Oriented Mission-Critical Sensor Networks

LONGYU ZHOU, SUPENG LENG, QIANG LIU, SUN MAO, AND YINHUA LIAO
School of Information and Communication Engineering, University of Electronic and Technology of China, Chengdu 611731, China

Corresponding author: Supeng Leng (spleng@uestc.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61731006, in part by the Fundamental Research Funds for the Central Universities, China, under Grant ZYGX2016Z011, and in part by the Science and Technology Department of Sichuan Province under Grant 18ZDYF0329.

ABSTRACT Mobile target tracking-oriented sensor networks are a special kind of Mission-critical Sensor Networks (MCSN), in which the various missions with the diverse priorities exist. However, it is challenging to achieve real time tracking while keeping the MCSN a long life time with limited energy provision in a complicated environment. In this paper, we develop a collaborative perception and intelligent scheduling scheme, which jointly optimizes the system responding latency and tracking accuracy with the constraint of the available energy. A new hierarchical architecture is proposed to realize the coupled function of perception and computation. In particular, the multi-node collaborative perception scheme is applied to obtain the excellent sensing capacity, and the Unmanned Aerial Vehicles (UAVs) play as the edge nodes to provide the computing service for those resource-constrained sensor nodes. To reach the sustained target tracking, we propose an intelligent tracking policy by exploiting the deep deterministic policy gradient (DDPG) method. Simulation results demonstrate that the proposed intelligent collaboration scheme can improve the tracking accuracy by 45.5% compared with the random selection scheme. The system cost is also reduced approximately by 17.3% while guaranteeing the tracking accuracy.

INDEX TERMS Multi-target tracking, energy consumption, collaborative perception, deep reinforcement learning.

I. INTRODUCTION

With the ability of connecting diverse devices with each other, internet of things (IoT) has been concerned as a key technology for the next generation mobile networks. Mission-Critical Sensor and Sensor Networks (MC-SSN) are a special kind of IoT, which focus on mission-critical applications of interest oriented wireless sensor networks (WSNs) [1]. Mission-critical WSN performs the low priority energy consumption and regards the critical mission as the top priority. Mission-critical WSN applications are regarded as those applications demanding data delivery bounds in the critical time, wherein mobile target tracking plays an important role. Sensor nodes can monitor surrounding areas to guarantee environmental safety and they can achieve critical tasks through monitoring moving targets. Monitoring data is collected and forwarded to the sink node or remote cloud, and sensor nodes carry out corresponding actions according to the feedback from cloud or the sink node. This kind of networks are widely applied in many promising fields, such as space expedition, marine monitoring, and terror attacks [2], [3].

MC-SSN has some unique features compared with ordinary WSN. The following challenges are regarded as key issues to work out.

(1) *Targets with diverse levels of priority invade to the monitoring region.* When multiple targets invade to the monitoring area, it is difficult to identify precisely the threatening target. The target with the top priority may be missed in some important monitoring areas.

(2) *There exists the irrational bandwidth utilization in mission-critical oriented wireless sensor network.* The invading targets with low priority may consume the limited bandwidth, thus the communications for monitoring tasks cannot be ensured.

The associate editor coordinating the review of this manuscript and approving it for publication was Qilian Liang.

(3) *Limited coverage area.* Due to the mobility of sensor nodes, the target with high priority may be missed or misjudged. High tracking accuracy may be not satisfied.

(4) *High latency response and feedback.* High-intensity computing consumption in the central server leads to the high system latency, which is not meet the real time control. Valid feedback may be not guaranteed.

The existing mobile target tracking strategies in sensor networks are divided into four categories [4]–[8], i.e., optimum in energy consumption, optimum in tracking accuracy, optimum in system latency, and optimum in system throughput. Nevertheless, these target tracking studies cannot directly apply to MC-SSN. We observe that sensor nodes with limited energy is quite difficult to track targets in long-term process, and accurate recognition and efficient transmission maybe not always guaranteed. Thus, efficient tracking and low missing probability maybe not obtained from traditional MC-SSN. Jondhale. *et al.* proposes an implementation of generalized regression neural network as an alternative to this traditional RSSI-based approach [9]. Mahboubi. *et al.* proposes an algorithm as graph divided into sufficient small cells, and uses a shortest path algorithm [10].

However, the existing work focuses on the improvement of energy consumption or tracking accuracy, which are not always feasible in the energy-limited networks. For instance, target nodes with quite critical missions require sufficient resource to obtain valid control and feedback. Low energy consumption maybe not guaranteed if the moving sensor node tracks target unremittingly. These issues drives us to design a joint optimal strategy, which can guarantee the high tracking accuracy and low responding latency with low energy consumption.

It is obvious that excellent tracking performance is based on the characteristics of quick response and efficient collaborative perception. In this paper, we investigate a collaborative perception scheme and intelligent tracking strategy to recommend the MMT in MC-SSN, taking account of invading target nodes with different priority. Considering the complicated and dynamic tracking environment, the tracking problem is modeled as a Markov Decision Process (MDP) composed of request queue, system delay and diverse critical missions. However, it is difficult to obtain the optimal scheduling strategy with a accepted latency when the system state space becomes large. The design of mobile target tracking-oriented MC-SSN need the joint optimization of collaborative perception, efficient computing, and real time feedback. In this case, we design a Deep Determined Policy gradient (DDPG)-based MMT scheme with collaborative perception. This scheme focuses on the determined policy, wherein the agent interacts with the complex environment and the optimal policy is obtained. Below are the main contributions of this paper.

- We propose a distributed intelligent framework for the MC-SSN. The collaborative perception scheme is applied to enhance the transmission efficiency for the critical missions. In order to achieve the real time feedback, information sharing is realized in the computation layer.
- The hierarchical architecture is proposed to achieve the coupled function of perception, computation, and service. System state and action are analyzed through the Markov Decision Process (MDP). The Kalman filter and the Euclidean space are applied to jointly establish a prediction model with random moving trajectories.
- An intelligent optimum scheduling scheme is proposed based on the deep deterministic policy gradient algorithm, wherein the deterministic policy is given through Actor-Critic architecture. Energy consumption, responding delay, and tracking accuracy are jointly considered in this paper. Numerical results reveal the better performance compared with another scheme, in terms of tracking accuracy and system cost.

The rest of this paper is organized as follows. An extensive surveys related this work is given by section II. The system model is given by section III. Section IV presents the problem formulation. The DDPG is provided in section V. Simulation results are provided in VI Finally section VII concludes this paper.

II. RELATED WORKS

Many researchers proposed mobile multi-target tracking schemes which attempt to improve the performance in terms of tracking accuracy and energy consumption. Considering the issue of the target tracking accuracy in indoor wireless networks. Luo *et al.* [11] proposed a cooperative localization and tracking scheme to improve the tracking accuracy respectively, which are divided into off-line phase and on-line phase. The former establishes the cooperative localization and the latter uses a region overlapping mechanism to narrow location area. The accuracy depends on the weight allocation and grid positioning error.

In terms of target tracking in the local area, firstly proposed a tracking scheme named "t-tracking" in [12] to achieve two goals, high Quality of Tracking (QoT) and high energy efficiency. The optimal function is formulated by authors based on the above two objectives. But there exists a fly in the ointment, if sensor nodes are static and sink nodes move, sensor nodes will face big problems of frequently switching and high energy consumption.

On the data processing of target tracking side, in [13], authors proposed a multi-rate distributed fusion estimation scheme to improve the energy efficiency. A minimum-error function between prediction and measurement, objects to covariance of the innovation vector based on conventional Kalman filter.

Considering the deployment and convergence problem, Mahboubi *et al.* [14] applies the concept of Graph and the shortest path algorithm to solve the above issues. The objective of Minimizing system energy consumption is formulated, subject to related constraints. But bringing in Graph to service the entire region of mobile sensor networks can result in some inadequacy, such as specific deployment, which limits the flexibility of moving nodes to some extent.

The authors of [15] and [16] proposed games-based scheme and a distributed strategy with consensus on estimates and missing data in terms of multi-target tracking, respectively. The objective of maximizing information about the target state and of estimating real-time position is designed to service the total mobile sensor network better. Alshamaa *et al.* [17] uses the belief function to embody the tracking accuracy and get good results in indoor positioning.

However, these works are based on the consideration in energy consumption or tracking accuracy, which are not always feasible in the energy-limited networks. Some special inferences, such as the environmental changing or the offloading of massive data, are not discussed specifically, especially in maritime. Besides, the real time feedback is not enhanced. In this paper, trade-off between energy consumption and tracking accuracy is guaranteed for real time tracking in maritime applications. Intelligent collaboration scheme and real time tracking strategy are realized in the MMT network. We mainly focus on the optimal tracking strategies, which is aided with the deep deterministic policy gradient algorithm.

III. SYSTEM MODEL

In this section, we illustrate the hierarchical framework for mobile multi-target tracking networks, and develop the communication model to represent the information delivery from sensor nodes to UAVs.

Fig. 1 illustrates our proposed cooperative framework for mobile multi-target tracking (MMT) networks, which can be widely applied in different application scenarios such as space expedition, maritime monitoring, and agricultural management, etc. In this framework, the sensor nodes in the perception layer can detect invasive targets and form multiple clusters, which connect and associate with different UAVs located in the computation layer. The UAVs collect deliverable data, execute computation and schedule the sensor nodes for tracking tasks. The remote cloud located in the service layer is regarded as the data center to provide the whole information in the MMT network.

Specifically, in the perception layer, the sensor nodes monitor the moving targets, which may be attacking nodes or the stranded persons. The collaborative perception is embodied in this layer. Once arbitrary sensor node detects a target, other nodes are informed using the broadcasting scheme and execute the common and critical mission. In the computation layer, the UAVs with the intelligent processing scheme to cope with the massive data requirement. The storage devices are encouraged to cache information. After computing, the control information is transmitted through downlink to schedule the proper sensor nodes for the critical tracking mission. However, the UAVs maybe cannot obtain fresh data immediately. In this case, they can require the remote cloud located in the server layer to exchange the data acquisition for the fetching cost.

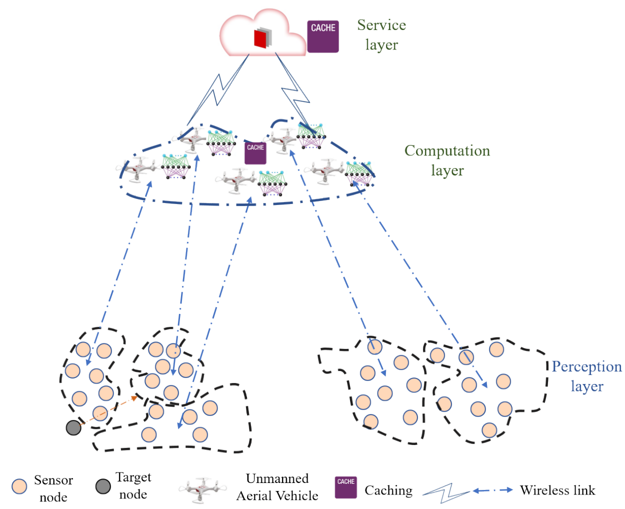

FIGURE 1. The paradigm of hierarchical model for MMT.

TABLE 1. Summary of key notations.

Notation	Description
$\mathcal{M}$	Index set of target nodes
$\mathcal{N}$	Index set of sensor nodes
$\mathcal{H}$	Index set of different priorities for target nodes
$\mathcal{C}$	Index set of Unmanned Aerial Vehicle (UAVs)
$g_{c,n}$	Average channel gain between UAV c and sensor node n
$\varpi_r^{c,n}$	Average receiving signal power between UAV c and sensor node n
$\gamma_{c,n}$	Signal Interference Noise Ratio (SINR) power
λ_n	Fetching cost for the sensor node n
P_{fetch}	The exception of the fetching cost
$\tau_{n,t}$	Transmission delay of the sensor node n
ψ_n	Scheduling delay of the sensor node n

Let $\mathcal{M} = \{0, 1, 2, \ldots, M\}$ represent the index set of target nodes, where 0 denotes that monitoring regions are not attacked by target node. In addition, let $\mathcal{N} = \{1, 2, \ldots, N\}$ denote the index set of sensor nodes to detect the illegal invasion. We further define $\mathcal{H} = \{1, 2, \ldots, H\}$ as the index set of diverse target nodes with different priorities, i.e., m_h represents the priority of target node m. Let $\mathcal{C} = \{1, 2, \ldots, C\}$ denote the index set of UAVs, which is regarded as Mobile Edge Computing (MEC) devices. This device is geographically close to the sensor nodes, which can reduce the transmission distance and transmission delay. It is worth noting that the set of UAVs is a subset of sensor nodes, i.e., $\mathcal{C} \subset \mathcal{N}$, furthermore, the size of C is equal to the number of self-organized networks. Let $t \in \{1, 2, \ldots, T\}$ denote the index of tracking times. The key notations are summarized in Table 1.

In the target tracking-oriented wireless sensor networks, transmission interference should be analyzed to enhance systematic timeliness. In order to reduce the transmission interference, we apply the orthogonal frequency-division multiple access (OFDMA) to achieve multi-node transmission. In the information delivery process, channel gain $g_{c,n}$ between the

UAV c and the sensor node n is decided by line-of-sight (LoS) $g_{c,n}^{LoS}$ and non-line-of-sight (NLos) $g_{c,n}^{NLos}$ from [18], and [19] gives specific expressions

$$\begin{cases} g_{c,n}^{LoS} = \left(\frac{4\pi f}{\upsilon}\right)^2 \mu_{LoS} l_{c,n}^{-\alpha_{LoS}} |h_{Rice}|^2, \\ g_{c,n}^{NLos} = \left(\frac{4\pi f}{\upsilon}\right)^2 \mu_{NLos} l_{c,n}^{-\alpha_{NLos}} \left|h_{Rayleigh}\right|^2, \end{cases} \tag{1}$$

where f denotes the carrier frequency, υ represents the speed of UAV, $l_{c,n}$ is the distance between UAV c and sensor node n, α_{LoS} and α_{NLos} indicate the LoS and NLos path loss exponents, respectively. *Rice* and *Rayleigh* denote the small-scale fading coefficient and are distributed Rice(λ, δ) and Rayleigh(0, δ) respectively, where λ denotes the mean value and δ is the standard deviation.

The weights of LOS and NLOS are given to obtain the average channel gain between the UAV c and the sensor node n, which is given by

$$g_{c,n} = \zeta_{LOS}^{c,n} \times g_{c,n}^{LOS} + \zeta_{NLOS}^{c,n} \times g_{c,n}^{NLOS}, \tag{2}$$

where $\zeta_{LOS}^{c,n}$ and $\zeta_{LOS}^{c,n}$ are the weights which are decided by the flight angle and specific environment. The weight of LoS will be

$$\zeta_{LOS}^{c,n} = \frac{a-b}{1+(\frac{\theta_{c,n}-c}{d})^e}, \tag{3}$$

where symbol a, b, c, d, e are constant values related to the environment, and $\theta_{c,n} = arcsin(\frac{h_n}{l_{c,n}})$ is the flight angle, and $\zeta_{LOS} + \zeta_{NLOS} = 1$.

In the MMT network, the different cluster with homogeneous characteristics have the same transmission power, in each time t, the average receiving signal power is represented as

$$\varpi_r^{c,n} = \frac{1}{N} \sum\nolimits_{n=0}^{N} P_t g_{c,n}(n), \tag{4}$$

where N is the number of transmitting with UAV c and P_t is the transmission power of each sensor node at the time t. Thus, the Signal Interference Noise Ratio (SINR) at the UAV c from the sensor node n at the time t is

$$\gamma_{c,n} = \frac{\varpi_r^{c,n}}{\sum_{j\in N\backslash n} \varpi_r^{c,j} + \delta^2}, \tag{5}$$

where δ^2 is the power of Gaussian white noise. According to the Shannon theorem, the transmission rate from the sensor n to the UAV c is

$$r_{c,n} = \frac{B\log(1+\gamma_{c,n})}{N_{c,t}}, \tag{6}$$

where $N_{c,t}$ is the number of requiring to access the UAV c in the time t.

IV. PROBLEM FORMULATION

In this section, the collaborative perception and irrational scheduling are discussed to obtain the optimal policy. Communication cost and delay cost are analyzed to establish state and action spaces. The objective function is formulated to request the optimal policy.

A. ANALYSIS OF THE COMMUNICATION MODEL

The perception strategy is proposed to realize the efficient monitoring for moving targets. Arrival of target nodes and entrance direction are random, we assume the arrival time is obeyed the Poisson distribution. The probability of the sensor node n monitoring the target m in the time t is $\eta_{n,m,t}$, when the event happens, transmission consumption for monitoring is produced, which is denoted as $p_{r,t}$. The requirement is real time for the sensor node n without data computing if the corresponding UAV has related information, but the overhead for fetching transmission should be considered which is embodied as follows referred to [20].

$$p_{tr,t} = \varepsilon_{elec} \times 2 + \varepsilon_{amp} \times d^2, \tag{7}$$

where ε_{elec} and ε_{amp} denote circuit and gain consumption of amplifier consumption of a bit respectively, due to simplify of data transmitting, downlink of 1 bit can realize scheduling strategy, i.e., 0 is prohibition, otherwise, permission and d is the distance between sensor and UAV node.

UAV with cache selects proper sensor nodes to track wisely, the probability of the scheduled sensor node n to track target m at time t is $\varphi_{n,m,t}$. Thus, the moving consumption of the sensor node n as a candidate at the time t is denoted as $p_{s,t}$. In the dynamic cluster, it is worthy noting that during the tracking, the sensor node n moves into another cluster, in which the cache is absent to the sensor node n, the corresponding data is obtained from another UAV or remote cloud. The fetching cost is associated. Let b_n denote the cost between UAV and remote cloud, the fetching cost is written as follows.

$$\lambda_n = 1(n \notin C^n) b_n, \quad C^n \subset \mathcal{C}, \tag{8}$$

where $1(*)$ is a indicator function, if condition $*$ is established, cost will product, if not, cost is zero.

As for the broadcasting consumption among UAVs, we assume that the probability of the UAV c without the sensor node n information. The probability of requiring remote cloud is φ_n, the probability of UAV with relative information receiving at time t is χ_t. The broadcasting consumption includes fetching cost which is the same as the equation (7). The fetching cost is summarized as

$$P_{fetch} = \chi\left(1-\varphi_n\right)\left(2\varepsilon_{elec} + \varepsilon_{amp} \times d^2\right) + \varphi_n b_n, \tag{9}$$

To simplify, the time t is omitted.

B. ANALYSIS OF THE DELAY MODEL

In the beginning of time t, let $\mathcal{N}_{m,t}^b$ denote the set of the sensor nodes requesting for the corresponding UAV, and $\mathcal{N}_{m,t}^b \subset \{\mathcal{N} \cup 0\}$. 0 denotes that no request to be scheduled. Let $\mathcal{N}_{m,t}^d$ denote the set of sensor nodes requesting for the UAV to track the target m during the time t, the alternating renewal process of time series is dynamic which is written as

$$\mathcal{N}_{m,t+1}^b = \mathcal{N}_{m,t}^d \cup \mathcal{N}_{m,t}^b, \tag{10}$$

$$\left|\mathcal{N}_{m,t+1}^{d}\right| = \begin{cases} |\mathcal{N}| \ or \ 0, & if \ r_t = m \\ \left|\mathcal{N} \backslash \mathcal{N}_{m,t}^{d}\right| \ or \ 0, & if \ r_t \neq m \end{cases} \quad (11)$$

where $r_t = m$ denotes sensor nodes request for UAV to track the target node m, otherwise, no request. Policy is obtained through smart agent.

From the above formula, the queue state is only relative to that at the last time, and the size of any requesting is unit length, so the exception for length of the requesting queue $R_{m,t+1}$ at the time $t+1$ is also represented as

$$R_{m,t+1} = \sum_{n=1}^{\mathcal{N}} \eta_{n,m,t}, \quad (12)$$

Due to the independence among sensor nodes, the scheduling set is explored as a subset of the requesting queue. In case $R_{n,m,t} = 0$, the scheduling set is empty, if not, the scheduling set is a non-empty subset of the request queue. The state matrix $\mathcal{R}_t$ with dimensions of $M \times N$ is structured and $\mathcal{R}_t \in \boldsymbol{R}$, $\boldsymbol{R}$ is the request queue state space. It is worthy noting that scheduling strategy is based on conditional probability, the precondition of realization for scheduling is that the request queue is non-empty. Due to the independence among sensor nodes, the scheduling strategy is uncorrelated to request probability and only related to the non-empty request queue.

In the beginning of each time t, the sensor node n receiving radio wave of the target m transmits own node information to UAV and is traversed according to the corresponding node ID in cache. The transmitting delay is given by

$$\tau_{n,t} = \frac{\kappa_{n,t}}{r_{c,n}}, \quad (13)$$

where κ_n denotes that transmission size of the sensor node n. if be scheduled, receiving consumption is conferred from the equation (7), so the whole scheduling delay consumption at the time t is rewrote as

$$\psi_n = \int_t^{t+1} \left(\tau_{n,t} + \frac{1}{r_{c,n,t}} + \frac{1}{v_{n,t}} \right) dt, \quad (14)$$

where $v_{n,t}$ denotes the moving speed of sensor nodes and assume each sensor node tracks with uniform speed.

To simplify and integral analysis, Let $L_{m,t} \in L_m \in [0, \overline{L_m}]$ denote delay constraint range for the target node m at the time t, where $\overline{L_m}$ is the maximum delay constraint. $L_m \in \boldsymbol{L}$ and $\boldsymbol{L}$ denotes the delay state space. The dynamic delay is embodied as

$$L_{m,t+1} = \begin{cases} 0, & if \ \eta_{m,t} = 0 \\ \min\left[L\left(\varphi_{m,t}\right), \overline{L}_{m,t+1}\right], & otherwise \end{cases} \quad (15)$$

where $L_m\left(\varphi_{m,t}\right)$ is a unity function wherein the delay mainly includes transmitting delay between sensor nodes and UAVs, tracking delay, and fetching delay with corresponding probability.

C. ANALYSIS OF THE STATE AND ACTION

In this section, we mainly use state space and action space to realize combination with reinforcement learning and convolutional Neural Network to obtain determined optimal policy [21]–[24].

In the MC-SSNs, each target with different priorities is found by sensor nodes, let $\theta_{i,t}, \forall i \in \mathcal{M}$ denote the priority of the target node m. Priority for each target nodes is denoted as $\theta_t = \left[\theta_{1,t}, \theta_{2,t}, \ldots \theta_{m,t}\right]$, and $\theta_t \in \boldsymbol{\theta}$ where $\boldsymbol{\theta}$ is the priority state space. Considering the sensor node n tracks target node m, Kalman Filter algorithm is introduced to obtain predicted trajectory, which is given by

$$\mathbf{x}_{t+1} = \mathbf{F}\mathbf{x}_t + \omega_t, \quad (16)$$

where $\mathbf{x}_t$ is a vector shown as $[x_t, y_t, \widehat{x_t}, \widehat{y_t}]^T$, $\mathbf{F}$ is transfer matrix based on the time t, and ω_t is noise matrix. Tracking accuracy at time t is represented as

$$e_t = \sqrt{\left(x_{m,t} - x_{n,t}\right)^2 + \left(y_{m,t} - y_{n,t}\right)^2}, \quad (17)$$

Accuracy is embodied by the Euclidean space. Let $\boldsymbol{e}$ denote the accuracy space. System state can be wrote as $\boldsymbol{S} = \boldsymbol{R} \times \boldsymbol{L} \times \boldsymbol{\theta} \times \boldsymbol{e}$ by integrating delay state space, request queue state space, priority state space, and accuracy state space. In order to simplify complexity, we consider single cluster due to same policy is realized in the MMT network. Let $\mathcal{N}_1 \subset \mathcal{N}$, $\mathcal{M}_1 \subset \mathcal{M}$ denote the set of the single network, so the state space $\boldsymbol{S}$ is reduced to $\boldsymbol{S}_1 = \boldsymbol{R}_1 \times \boldsymbol{L}_1 \times \boldsymbol{\theta}_1$, in the time t, the state space is given by

$$\begin{aligned} \boldsymbol{S}_1 = \{ s_t = & \left(\boldsymbol{R}_{1,t} \times \boldsymbol{L}_{1,t} \times \boldsymbol{\theta}_{1,t} \times \boldsymbol{e}_{1,t}\right) \\ & \times |p\left(L_{1,n,t}, \theta_{1,m,t}, e_{1,n,m,t} | R_{1,n,t}\right)\}, \end{aligned} \quad (18)$$

where delay constraint is only appeared when request queue is non-empty and $n \in \mathcal{N}_1$, $m \in \mathcal{M}_1$, the reduced state $\boldsymbol{S}_1 \in \boldsymbol{S}$.

The exploring strategy is analyzed in this section by means of using deep deterministic policy gradient algorithm with interaction between smart agent and the MC-SSN. The detail is discussed in the next subsection.

Given a random state $s \in \boldsymbol{S}_1$, the action with penalty mechanism is arose resulted from the interacting state. Policy π is a mapping connecting state and action, which is given by

$$\pi(\mathcal{S}) \rightarrow \mathcal{A}, \quad (19)$$

In the real-time MMT network, the single network state space $\boldsymbol{S}_1$ with four dimensions is difficult to realize the mapping through exploring the MMT network, we use the convolutional neural network (CNN) to overcome curse of dimensions. To cope with the convergence of self-learning, we adopt the Actor-Critic to guarantee the convergence of the algorithm. However, when to reach the convergence is uncertain. The deep deterministic policy gradient merging deep Q-network is employed to solve the difficulty of convergence [24]–[27].

For a practical mobile target tracking system, energy consumption is unknown to the sensor node m at each time t

because of the dynamic environment. For a given target node m, the systematic energy consumption is represented by

$$P(\pi) = \sum_{t=1}^{T} P_t(s_t, a_t), \tag{20}$$

where

$$P_t(s_t, a_t) = \eta_{n,t} P_{tr,t} + P_{n,fetch,t} + \varphi_{n,t} P_{s,t}, \tag{21}$$

The delay penalty for target nodes with diverse delay constraints is denoted by

$$D(\pi) = \sum_{t=1}^{T} D_t(s_t, a_t), \tag{22}$$

where

$$D_t(s_t, a_t) = \sum_{n=1}^{N} \sum_{m=1}^{M} f_m(L_{m,t} - \psi_{n,m,t}), \tag{23}$$

denotes systematic delay penalty, $f_m(*)$ denotes delay of total target nodes with diverse delay constraints at the time t and f_m is a general function. To ensure the real time tracking, the shorter the scheduling time, the better in theory. Moreover, the target nodes with diverse priority is monitored and tracked which is random for the dynamic scenario, so our object is that we reduce the delay as much as possible even through the delay is within the $\bar{L}_{m,t}$. The specific expression is given by

$$D_t(s_t, a_t) = \begin{cases} f_m(\vartheta_{m,t}), & if\ \vartheta_{m,t} \le 0 \\ g_m(\vartheta_{m,t}), & if\ \vartheta_{m,t} > 0 \end{cases} \tag{24}$$

where $\vartheta_{m,t}$ denotes the error value between practice delay and delay constraint at the time t. The punishment intensity of function g_m is much weighted than f_m which means that delay is generated with different punishment. Once delay $\vartheta_{m,t}$ exceeds 0, the penalty will be aggravated gradually.

For a dynamic MMT network, it is difficult to use polynomial time to obtain the optimal policy, so the deep deterministic policy gradient algorithm is employed in MMT. Our object is to joint minimize the systematic energy consumption including the fetching cost, the delay penalty, and the tracking accuracy. the unity function of the joint optimal object is given by

$$U(\pi) = \beta_1 P(\pi) + \beta_2 e(\pi) + \beta_3 D(\pi), \tag{25}$$

where β_1, β_2, and β_3 are the weighted parameters which satisfy $\beta_1 + \beta_2 + \beta_3 = 1$, respectively. In each time t,

$$U(s_t, a_t) = \beta_1 P(s_t, a_t) + \beta_2 e(\pi) + \beta_3 D(s_t, a_t), \tag{26}$$

is the unity function to minimize the systematic cost.

As the above state, the optimal strategy π should be obtained to minimize the systematic cost. Scheduling problem is denoted as

$$P1 : u^* = \min U(s_t, a_t), \tag{27}$$

where u^* is the optimal joint solution resulting from optimal policy π^*. For a infinite horizon systematic cost MDP, the optimal strategy is learned through interaction between the agent of DDPG and the MMT with the infinite state space and the infinite action space which is multidimensional.

V. DDPG-BASED SCHEDULING STRATEGY

Due to the high-complex dynamic mobile target tracking system, we need deep network construction, experience pool is adopted in DDPG. policy gradient method selects random action by learning policies in the continue action space and the deterministic method is employed to assist policy gradient to output unique action value stably [27]–[29]. To cope with the difficulty of exploring complex environment, deterministic policy is used to solve through the off-policy sampling. As for the DDPG, Actor-Critical architecture is adopted to import the neural network based policy and value. each state-action pair (s_t, a_t) is mapped to reward or penalty, take the reward for instance, the specific expression is given by

$$R_t = \sum_{i=t} \gamma^{i-t} r(s_i, a_i), \tag{28}$$

where γ is the discount factor, and the reward is accumulated from the previous state to i_{th} state of future.

As the above state, we transfer the P1 to P2 using DDPG algorithm. Deterministic policy is considered to optimize systematic scheduling strategy. The action function is wrote as

$$a_t = \mu(s_t|\theta^\mu), \tag{29}$$

where μ is the optimal action policy, and θ^μ is the parameter of policy network. We adopt the reward to perform the selected policy, which is expressed as

$$J(\mu_\theta) = E_{s,\rho^\mu}[r(s, \mu_\theta(s))], \tag{30}$$

We give the iteration to obtain optimal policy and the deterministic policy is given by

$$\nabla_\theta J(\mu_\theta) = E_{s,\rho^\beta;a,\pi_\theta}[\nabla_\theta \log \pi_\theta(a|s) Q^\pi(s, a)], \tag{31}$$

where $Q(*)$ denotes the value function, β is the behavior policy.

Due to the random character of policy gradient, the computation is difficult to obtain optimal policy based on previous action, the deterministic policy is adopted and the policy gradient is rewrote as

$$\nabla_\theta J(\mu_\theta) = E_{s,\rho^\beta}[\nabla\theta \mu_\theta(s) Q^\mu(s, a)|_{a=\mu_\theta}], \tag{32}$$

DDPG is based on the equation (30) and deep Q-network, and the parameter is given through policy network with deterministic action using θ^μ, actor is embodied by policy network μ and critical is obtained by fitting state-action function under the value network. The reward function is rewrote as

$$J(\theta^\mu) = E_{\theta^\mu}[r_1 + \gamma r_2 + \gamma^2 r_3 + \ldots], \tag{33}$$

The reward is selected under the deterministic policy μ. In DDPG, the Q-network is adopted to fit the value function. The value function is given by the P2.

$$P2 : Q^\mu(s_t, a_t) = E[r(s_t, a_t) + \gamma Q^\mu(s_{t+1}, \mu(s_{t+1}))], \tag{34}$$

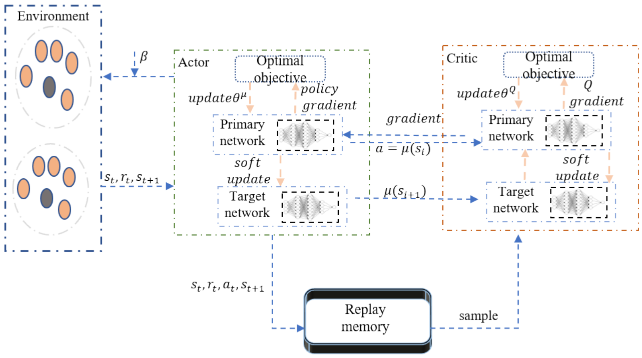

FIGURE 2. The construction of deep deterministic policy gradient algorithm.

In order to evaluate the capacity of policy μ, the specific expression is given by

$$J_{\beta}(\mu) = E_{s,\rho^{\beta}}[Q^{\mu}(s, \mu(s))], \tag{35}$$

In Fig. 2, we describe the specific interaction process. In order to minimize cost, state space as input and the agent learns important information. θ^{μ} is updated and delivered to online policy network. After obtaining the corresponding state-action pair and reward, relative parameters are stored to experience reply memory. θ^{μ} is updated in online Q-network and target network samples randomly from the reply memory to reduce correlation of values. behavior policy β as noisy is obtain using Uhlenbeck-Ornstein stochastic process to explore the optimal policy. It is worthy noting that two networks are asynchronous. Convolution neural network is applied to train, in which the state space is regarded as the input. In order to guarantee non-correlation, the state and action data sequence of interaction is stored to replay memory, in which the min batch data is sampled. This algorithm stabilizes learning process and guarantee the convergence. The algorithm flow is showed as follows.

The time complexity is analyzed specifically which is divided into two parts [30]–[32].

- Considering the whole system from algorithm 1, we can obtain the obvious time complexity with $\mathcal{O}(n^2)$. It is noteworthy that the feedback process of neural network should be consider because of the computation of inverse of the matrix.
- For DQN algorithm with random characteristic, action a is sampled from set of actions $\mathcal{A}$, therefore there exists a time complexity in $\mathcal{O}(1)$ for each iteration. Due to calculation of constant value for forward-propagation, the main consideration is focused on back-propagation. According to [11], the time complexity of proposed algorithm is $O(k(\phi)T\mathcal{M}_1 |\mathcal{C}| |\mathcal{A}_1|)$, where $\mathcal{A}_1$ and $\mathcal{M}_1$ are the reduced action space and set of target nodes, the $|\mathcal{C}|$ is the number of sink nodes in a region, and the $k(\phi)$ is the a function of the depth and number of the hidden layers ϕ.

Algorithm 1 DDPG based scheduling algorithm

Input: state vector s_t, step number N, discount factor γ, learning rate α, the number of step $\mathcal{M}$;

Initial network parameters θ^{Q}, θ^{μ} and replay memory buffer $\mathcal{R}$;
for each interaction with environment *episode* $\in \mathcal{M}$ **do**
  for each time $t \in T$ **do**
    Select a_t through behavior policy, which is represented by equation (27);
    Get the new reward r_t and the next state s_{t+1};
    Sample $\mathcal{N}$ data from replay memory $\mathcal{R}$ and is represented through (s_i, a_i, r_i, s_{i+1});
    Calculate the gradient of primary network.
    Updating policy Q value θ^{Q} and compute policy gradient which is represented through equation (30);
    Update network parameter θ^{μ};
    Compute target network μ' and Q'
    $Q \leftarrow Q'$
  end for
end for

Output: Value Q;

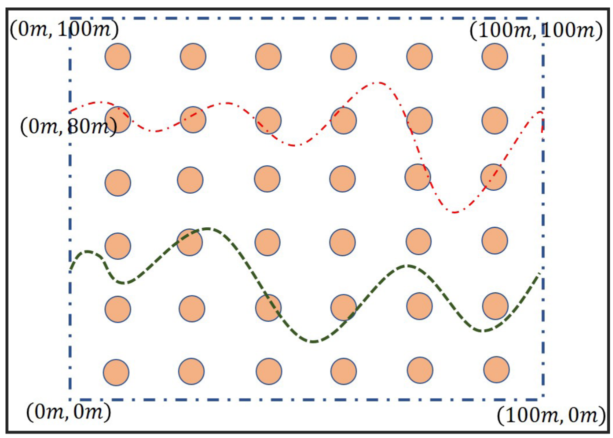

FIGURE 3. The paradigm of multi-target tracking in MC-SSN.

VI. SIMULATION RESULTS

In this section, Simulation experiments are conducted to illustrate the performance of the proposed optimal policy using the DDPG algorithm. We use Python 3 to build a simulation environment for MMT task in MC-SSN networks. TensorFlow architecture is built to verify the capacity of multi-target tracking and scheduling strategies. The learning process consumes approximately 12 minutes when the iteration number reaches 1000.

A. SIMULATION SETUP

In Fig. 3, we consider the mobile target tracking scenario, in which the area of each region is 100m $\times$ 100m. The number of target nodes $\mathcal{M} = 2$, the number of sensor nodes $\mathcal{N} = 40$, and the velocities of sensor nodes and

TABLE 2. Simulation parameters of MC-SSN environment.

Parameters description	value
Area of each region	100m × 100m
Number of target nodes	2
Velocity of target nodes	1m/s
Number of sensor nodes	36
Velocity of sensor nodes	2m/s
Maximum tolerate delay of target node 1	3s
Maximum tolerate delay of target node 2	4s
Primary energy of each sensor node	40J
Primary coordinate of target node 1	(0m,40m)
Primary coordinate of target node 2	(0m,80m)
Energy consumption for static nodes	0.1J/ unit time
Energy consumption for moving nodes	0.6J/ unit time
Range of learning rate	0.6-0.9
Discount factor	0.9
Size of min-bath	32
Size of replay memory	500

targets are 2m/s and 1m/s, respectively. There exists two target nodes with different critical missions, in which the maximum tolerate delay of the two target nodes are $3s$ and $4s$, respectively. We set the energy of each sensor node $p_i = 40J, \forall i \in \mathcal{N}$. The coordinates of running into each region for two targets are set as $(0m, 40m)$ and $(0m, 80m)$ and the moving trajectory are stochastic distributions. Not only that, the energy consumption of static nodes in each unit time length t also is considered as 0.1J but 0.6J for each moving sensor. To clearly state, the simulation parameters of MMT for executing critical missions are summarized in Table 2. We compare our proposed scheme with the random selection scheme and non-myopic planning (OL) based on the potential game [15].

B. RESULTS ANALYSIS

We first study tracking accuracy under the different schemes. The Mean Squared Error (MSE) is proposed to express the accuracy, which is given by $MSE(t) \triangleq \frac{1}{N}\sum_{i=1}^{N}(x_i(t) - x_t(t))^2 + (y_i(t) - y_t(t))^2$. The random selection scheme is introduced to compare with the proposed scheme. In the random selection scheme, sensor nodes are scheduled by the corresponding UAV without intelligence. No penalty or reward rules are achieved in this scheme and sensor nodes are selected through random probability.

Fig. 4 shows the comparison of the cumulative system tracking error of different scheduling schemes based DDPG learning algorithm. For a scheduling system with $\mathcal{M} = 2$, Both the two scheduling schemes can approach their stable cumulative tracking errors as the number of iterations increases. We can draw the following observations from Fig. 4. Firstly, random scheduling scheme has the lower tracking accuracy compared with the proposed scheme. The reason is that the sink node needs to adjust state of each sensor node, in which self-energy and tracking capacity determine the optimal scheduling instead of selecting randomly. Secondly, intelligent scheduling scheme can enhance tracking accuracy in he complicated environment. Thirdly, our proposed scheme can improve tracking accuracy 45.5% approximately, compared with random scheduling scheme.

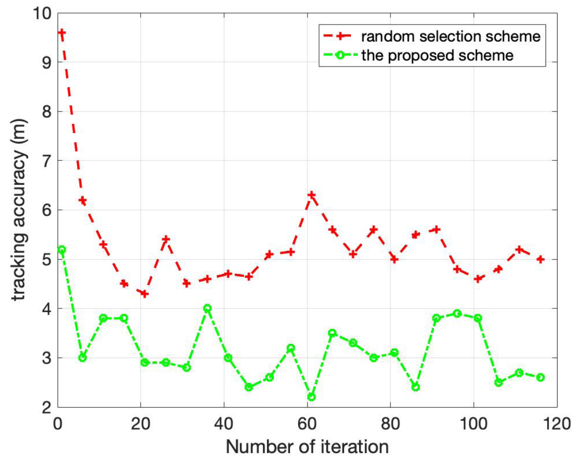

FIGURE 4. Cumulative average tracking error by different scheduling schemes.

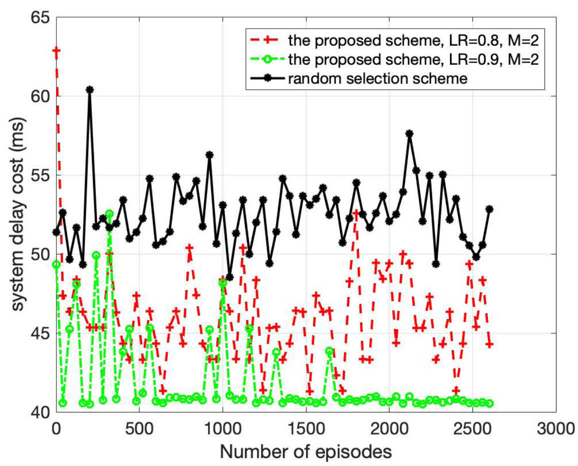

FIGURE 5. Cumulative average system tracking latency by different scheduling schemes and different learning rate (LR).

Fig. 5 provides the comparison of cumulative system tracking latency of moving sensor nodes. In order to reduce system delay and enhance responding speed, intelligent scheduling scheme seeks a quick response mode to obtain the lowest penalty if there exists a critical target, which is tracked urgently by sensor nodes. The proposed scheme can reduce system delay 12.5% approximately compared with random scheduling scheme. Furthermore, our proposed scheme results in the stable system delay due to the joint optimization the collaborative tasks updating and feedback.

In Fig. 6, we compare the system cost with random scheduling scheme. Under the condition of guaranteeing tracking accuracy, both the two scheduling schemes can reach their stable cumulative system cost as the number of episodes increases. Based on the DDPG algorithm, the system cost is convergent with slight confusion within little ranges. The reason is that the smart agent interacts with environment

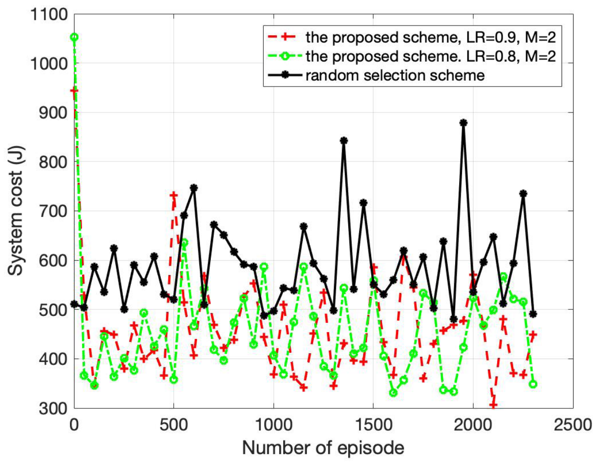

FIGURE 6. Cumulative average system cost by different scheduling schemes and different learning rate (LR).

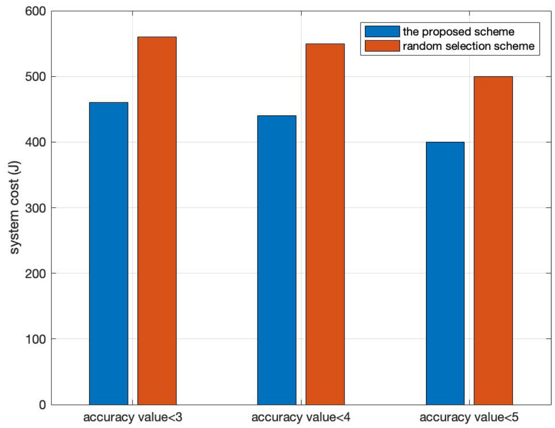

FIGURE 7. Average system cost achieved by different tracking accuracy.

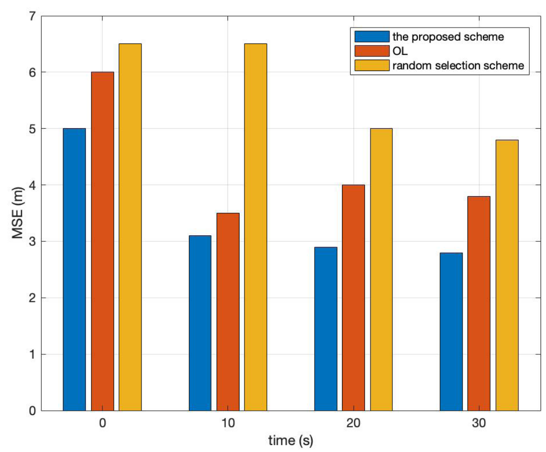

FIGURE 8. The comparison with OL, random selection scheme in Mean Squared Error (MSE).

through penalty mechanism. Different punish factors result in different system convergent capacity. The proposed scheme reduce 27.3% system cost compared with random scheduling scheme, which is given under guarantees system tracking accuracy. Thus, this result also demonstrates the efficiency of joint optimization of collaborative task offloading and valid feedback.

In Fig.7, we compare the cumulative system cost achieved by variable tracking accuracy using different scheduling schemes. Firstly, when the number of target nodes is 2, the different system cost is obviously significant while guaranteeing tracking accuracy. The collaborative scheduling scheme reduces the system cost 16.3%, 17.3%. 20.8%, compared with the random scheduling scheme, respectively. This result shows that the proposed scheme can make full use the unexploited resource of MC-SSN. Moreover, the results demonstrate the long-term tracking can be guaranteed with the proposed scheme, especially in the critical missions. Fig. 8 shows the mean square error by using different schemes. We can obverse that the error is decreased with the increase of the time for all the proposed scheme, OL [15], and the random selection scheme. Simulation results show that the proposed scheme outperforms OL about 16.7% on the tracking error.

VII. CONCLUSION

In this paper, we first propose a collaborative perception mechanism and confirm its performance for real time multiple targets tracking applications in MC-SSN networks, With this perceptive method, a distributed collaborative paradigm is proposed to offload the critical tasks to the UAV nodes, which is regarded as the MEC platforms with a certain intelligence. Furthermore, an intelligent scheduling strategy is proposed to cooperatively schedule the tracking tasks and guarantee low-latency and low energy consumption. Numerical results indicate that our proposed scheme can reduce scheduling delay and systematic consumption while guaranteeing tracking accuracy. Based on the analyzed indicators, our proposed scheme improves about 20% and 16.7% contrasted with random scheduling scheme and OL [15] scheme in system capacity.

REFERENCES

[1] P. Suriyachai, U. Roedig, and A. Scott, "A survey of MAC protocols for mission–critical applications in wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 2, pp. 240–264, 2nd Quart., 2012.

[2] L. Gao, G. Battistelli, L. Chisci, and P. Wei, "Distributed joint sensor registration and multitarget tracking via sensor network," *IEEE Trans. Aerosp. Electron. Syst.*, to be published.

[3] L. Yang, L. Xiuju, J. Huixia, F. Yuanyuan, and Z. Chi, "Multi-sensing based target tracking by using decision-making strategy with spatial and temporal properties," *J. Wireless Commun. Netw.*, vol. 2019, no. 1, Dec. 2019, Art. no. 117.

[4] L. Wanjun, S. Hu, and J. Wentao, "Correlation filter tracking algorithm for adaptive feature selection," *Acta Optica Sinica*, vol. 39, no. 6, 2019, Art. no. 0615004.

[5] C. Wu, C. Zhao, and H. Gong, "Energy–efficient target tracking algorithm for WSNs," *3D Res.*, vol. 10, no. 1, Mar. 2019, Art. no. 1.

[6] X. Lu, Y. Zhang, J. Liu, F. Yuan, and L. Cheng, "Mobile target tracking algorithm for wireless camera sensor networks with adjustable monitoring direction of nodes," *Int. J. Commun. Syst.*, vol. 32, no. 10, p. e3944, Jul. 2019.

[7] W. Lu, Z. Zhou, L. Zhang, and G. Zheng, "Multi-target tracking by non-linear motion patterns based on hierarchical network flows," *Multimedia Syst.*, vol. 25, no. 4, pp. 383–394, Aug. 2019.

[8] K. Yoon, Y.-M. Song, and M. Jeon, "Multiple hypothesis tracking algorithm for multi-target multi-camera tracking with disjoint views," *IET Image Process.*, vol. 12, no. 7, pp. 1175–1184, Jul. 2018.
[9] S. R. Jondhale and R. S. Deshpande, "Kalman filtering framework-based real time target tracking in wireless sensor networks using generalized regression neural networks," *IEEE Sensors J.*, vol. 19, no. 1, pp. 224–233, Jan. 2019.
[10] H. Mahboubi, W. Masoudimansour, A. G. Aghdam, and K. Sayrafian-Pour, "An energy–efficient target–tracking strategy for mobile sensor networks," *IEEE Trans. Cybern.*, vol. 47, no. 2, pp. 511–523, Feb. 2017.
[11] J. Luo, Z. Zhang, C. Liu, and H. Luo, "Reliable and cooperative target tracking based on WSN and WiFi in indoor wireless networks," *IEEE Access*, vol. 6, pp. 24846–24855, 2018.
[12] M. Z. A. Bhuiyan, G. Wang, and A. V. Vasilakos, "Local area prediction–based mobile target tracking in wireless sensor networks," *IEEE Trans. Comput.*, vol. 64, no. 7, pp. 1968–1982, Jul. 2015.
[13] X. Yang, W.-A. Zhang, L. Yu, and K. Xing, "Multi–rate distributed fusion estimation for sensor network–based target tracking," *IEEE Sensors J.*, vol. 16, no. 5, pp. 1233–1242, Mar. 2016.
[14] H. Mahboubi, W. Masoudimansour, K. Sayrafian-Pour, and A. G. Aghdam, "An energy-efficient target-tracking strategy for mobile sensor networks," *IEEE Trans. Cybern.*, vol. 47, no. 2, pp. 511–523, Feb. 2017.
[15] S.-J. Lee, S.-S. Park, and H.-L. Choi, "Potential game–based non–myopic sensor network planning for multi–target tracking," *IEEE Access*, vol. 6, pp. 79245–79257, 2018.
[16] M. Vazquez-Olguin, Y. S. Shmaliy, O. Ibarra-Manzano, J. Munoz-Minjares, and C. Lastre-Dominguez, "Object tracking over distributed WSNs with consensus on estimates and missing data," *IEEE Access*, vol. 7, pp. 39448–39458, 2019.
[17] D. Alshamaa, F. Mourad-Chehade, and P. Honeine, "Tracking of mobile sensors using belief functions in indoor wireless networks," *IEEE Sensors J.*, vol. 18, no. 1, pp. 310–319, Jan. 2018.
[18] A. Pokkunuru, Q. Zhang, and P. Wang, "Capacity analysis of aerial small cells," in *Proc. IEEE Int. Conf. Commun.*, May 2017, pp. 1–7.
[19] Y. Cao, L. Zhang, and Y.-C. Liang, "Deep reinforcement learning for user access control in UAV networks," in *Proc. IEEE Int. Conf. Commun. Syst. (ICCS)*, Dec. 2018, pp. 297–302.
[20] X. Yu and J. Liang, "Genetic fuzzy tree based node moving strategy of target tracking in multimodal wireless sensor network," *IEEE Access*, vol. 6, pp. 25764–25772, 2018.
[21] J. Liang, Y. Hu, H. Liu, and C. Mao, "Fuzzy clustering in radar sensor networks for target detection," *Ad Hoc Netw.*, vol. 58, pp. 150–159, Apr. 2017.
[22] K. Xiong, S. Leng, J. Hu, X. Chen, and K. Yang, "Smart network slicing for vehicular fog–RANs," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3075–3085, Apr. 2019.
[23] G. Qiao, S. Leng, K. Zhang, and K. Yang, "Joint deployment and mobility management of energy harvesting small cells in heterogeneous networks," *IEEE Access*, vol. 5, pp. 183–196, 2017.
[24] M. Shaolin and W. Yue, "Design and implementation of an ultra-wideband high-accuracy ranging system," *J. Tianjin Normal Univ., Natural Sci. Ed.*, vol. 37, no. 6, pp. 55–57, 2017.
[25] Z. Qinglian and Z. Xin, "Adaptive genetic algorithm with three-parent crossover for numerical optimization," *J. Tianjin Normal Univ., Natural Sci. Ed.*, vol. 37, no. 5, pp. 55–59, 2017.
[26] S. Yang, F. Teich, and M. Baum, "Network flow labeling for extended target tracking PHD filters," *IEEE Trans. Ind. Informat.*, vol. 15, no. 7, pp. 4164–4171, Jul. 2019.
[27] B. Fan, S. Leng, and K. Yang, "A dynamic bandwidth allocation algorithm in mobile networks with big data of users and networks," *IEEE Netw.*, vol. 30, no. 1, pp. 6–10, Jan. 2016.
[28] J. Liang, X. Yu, and H. Li, "Collaborative energy–efficient moving in Internet of Things: Genetic fuzzy tree versus neural networks," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6070–6078, Aug. 2019.
[29] G. Qiao, S. Leng, K. Zhang, and Y. He, "Collaborative task offloading in vehicular edge multi–access networks," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 48–54, Aug. 2018.
[30] J. Wang, Y. Yang, L. Liu, and N. Wu, "High-speed target tracking system based on multi-interconnection heterogeneous processor and multi-descriptor algorithm," *Sci. China Inf. Sci.*, vol. 62, no. 6, Jun. 2019, Art. no. 69401.
[31] C. Xiu and Z. Chai, "Target tracking based on the cognitive associative network," *IET Image Process.*, vol. 13, no. 3, pp. 498–505, Feb. 2019.
[32] Y. Ding, X. Yu, J. Zhang, and X. Xu, "Application of linear predictive coding and data fusion process for target tracking by Doppler through–wall radar," *IEEE Trans. Microw. Theory Techn.*, vol. 67, no. 3, pp. 1244–1254, Mar. 2019.
[33] S. Mao, S. Leng, S. Maharjan, and Y. Zhang, "Energy efficiency and delay tradeoff for wireless powered mobile-edge computing systems with multi-access schemes," *IEEE Trans. Wireless Commun.*, doi: 10.1109/TWC.2019.2959300.