Asynchronous FDRL-based Low-Latency Computation Offloading for Integrated Terrestrial and Non-Terrestrial Power IoT

Sifeng Li, Sunxuan Zhang, Zhao Wang, Zhenyu Zhou, Xiaoyan Wang, Shahid Mumtaz, Mohsen Guizani, and Valerio Frascolla

***Abstract*—Integrated terrestrial and non-terrestrial power internet of things (IPIoT) has emerged as a paradigm shift to three-dimensional vertical communication networks for power systems in the 6G era. Computation offloading plays key roles in enabling real-time data processing and analysis for electric services. However, computation offloading in IPIoT still faces challenges of coupling between task offloading and computation resource allocation, resource heterogeneity and dynamics, and degraded model training caused by electromagnetic interference (EMI). In this article, we propose an asynchronous federated deep reinforcement learning (AFDRL)-based computation offloading framework for IPIoT, where models are uploaded asynchronously for federated averaging to relieve network congestion and improve global model training. Then, we propose Asynchronous fedeRated deep reinforcemenT learnIng-baSed low-laTency computation offloading algorithm (ARTIST) to realize low-latency computation offloading through joint optimization of task offloading and computation resource allocation. Particularly, ARTIST adopts EMI-aware federated set determination to remove aberrant local models from federated averaging and improve training accuracy. Next, a case study is developed to validate the excellent performance of ARTIST in reducing task offloading and total queuing delays.**

***Index Terms*—6G, integrated terrestrial and non-terrestrial power internet of things, computation offloading, asynchronous federated deep reinforcement learning, electromagnetic interference awareness**

I. INTRODUCTION

Power internet of things (PIoT) provides comprehensive interconnection among human, machine, and things through all the aspects of power generation, transmission, distribution, and consumption. PIoT requires low latency, flexible coverage, and high security to support delay-sensitive power services and connection of external devices into the grid. However, in some remote areas, the coverage of terrestrial networks such as 5G is insufficient to meet the stringent requirements of PIoT. Moreover, terrestrial networks cannot support emergency communication in harsh environments and disasters [1], [2].

With the rapid development of 6G, non-terrestrial networks based on satellites, high altitude platforms (HAPs), and unmanned aerial vehicles (UAVs) have been applied in ocean monitoring, emergency command, and deep space exploration. Integrated terrestrial and non-terrestrial PIoT (IPIoT) provides a new paradigm shift to three-dimensional vertical power communication network, which possesses wide coverage, ubiquitous connection, flexible deployment, and high robustness [3]. In IPIoT, computation offloading is a key technology to realize real-time processing and analysis of massive data by combining the powerful computation capacity of terrestrial networks with the coverage and deployment advantages of non-terrestrial networks. Resource-constrained PIoT devices can offload computation-intensive tasks to edge and cloud servers via terrestrial base station (BS) or non-terrestrial UAV, HAP and satellite. To enable effective computation offloading, the multi-dimensional heterogeneous resources of terrestrial and non-terrestrial networks should be intelligently allocated to satisfy the differentiated requirements of PIoT services through the joint optimization of task offloading and computation resource allocation [4]. However, the joint optimization problem is difficult to be solved by traditional mathematical optimization tools due to dynamic connection, network heterogeneity, large optimization space, and incomplete information. In the 6G era, artificial intelligence (AI) provides an effective solution with powerful learning ability. It is intuitive to explore AI-based intelligent methods to solve the computation offloading problem of IPIoT [5].

Federated deep reinforcement learning (FDRL) combines the capabilities of deep reinforcement learning (DRL) in complex mapping relationship fitting and model-free decision making as well as the advantages of federation learning (FL) in device-side environment observation utilization and secure model training [6]. FDRL has been utilized to address the computation offloading problem [7].

In this paper, we improve FDRL by adding asynchronous functionality to develop a new asynchronous FDRL (AFDRL) framework for computation offloading optimization in IPIoT. The proposed framework adapts to multi-dimensional resource heterogeneity and dynamics through asynchronous model uploading. The total queuing delay is minimized through the

S. Li, S. Zhang, Z. Wang, and Z. Zhou are with Hebei Key Laboratory of Power Internet of Things Technology, School of Electrical and Electronic Engineering, North China Electric Power University, Beijing, 102206, China. (E-mail: sifeng_li@ncepu.edu.cn, sunxuan_zhang@ncepu.edu.cn, zhao_w@ncepu.edu.cn, zhenyu_zhou@ncepu.edu.cn)

X. Wang is with Graduate School of Science and Engineering, Ibaraki University, Ibaraki 310-8512, Japan. (E-mail: xiaoyan.wang.shawn@vc.ibaraki.ac.jp)

S. Mumtaz is with the Department of Applied Informatics, Silesian University of Technology, Akademicka, 44-100 Gliwice, Poland, and also with the Department of Engineering, Nottingham Trent University, NG1 4FQ Nottingham, U.K. (E-mail: dr.shahid.mumtaz@ieee.org)

M. Guizani is with the Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, UAE. (E-mail: mguizani@ieee.org)

V. Frascolla is with Intel Deutschland GmbH, 85579 Neubiberg, Germany. (E-mail: valerio.frascolla@intel.com)

joint optimization of task offloading and computation resource allocation. The main contributions are summarized as follows.

- ***AFDRL-based computation offloading framework***: The proposed framework integrates heterogeneous resources of terrestrial and non-terrestrial networks in IPIoT through the joint optimization of device-side task offloading and server-side computation resource allocation. PIoT devices asynchronously upload adequately trained local models for federated averaging to relieve network congestion and improve convergence of global model training.
- ***Low-latency joint optimization of task offloading and computation resource allocation***: We propose the Asynchronous fedeRated deep reinforcemenT learnIng-baSed low-laTency computation offloading algorithm (ARTIST) to solve the joint optimization problem of task offloading and computation resource allocation. Lyapunov optimization is leveraged to decompose the joint problem into two stages. In the first-stage task offloading subproblem, ARTIST utilizes asynchronous federated deep actor critic (AFDAC) to learn the BS selection and computing paradigm selection decisions. Then, ARTIST approximates the second-stage server-side computation resource allocation problem into a convex one and solve it. Low-latency computation offloading is achieved by reducing queuing delay of task offloading and computation.
- ***EMI-aware federated set determination***: ARTIST adopts electromagnetic interference (EMI)-aware federated set determination to avoid the adverse impacts of sudden and strong EMI on learning optimality and convergence. Specifically, a cost deviation function is defined to evaluate local models. If the cost deviation exceeds a certain threshold, the detected aberrant model is removed from the federated set to achieve EMI awareness and improve training accuracy. The proposed mechanism can be applied in other industrial environments due to its scalability and flexibility.

First, the application scenarios and research challenges of computation offloading in IPIoT are introduced in Section II. Then, the AFDRL-based computation offloading framework and ARTIST are presented in Section III. Afterwards, a case study is developed in Section IV. Finally, the conclution of the article and future research directions are provided in Section V.

II. IPIOT Application Scenarios and Research Challenges

A. Application Scenarios of Computation Offloading in IPIoT

Desert photovoltaic (PV) monitoring: Real-time monitoring of desert solar power stations is challenged by the limited terrestrial communication infrastructure. The positioning, timing, and short message communication technologies of satellites as well as the high-speed communication coverages of UAVs and HAPs in IPIoT can meet the requirements of positioning, synchronization, and long-distance data transmission in desert PV monitoring [8]. Data generated by PV-side controllers, grid-connected inverters, anti-island devices, environment sensors, and other state acquisition devices in solar stations are aggregated and preprocessed locally by terrestrial networks. Then, computation-intensive tasks such as PV scheduling control and real-time output forecasting can be offloaded to the cloud server with powerful computation capabilities through non-terrestrial networks. Finally, through global data processing and analysis, advanced functions such as large-scale grid connection/off-grid of solar stations and autonomous PV output scheduling are executed [9].

Transmission line inspection: Transmission lines are criss-crossed and often distributed in hilly wilderness. Therefore, it is extremely difficult to achieve the full coverage of transmission line inspection. In IPIoT, UAVs are leveraged to collect real-time data such as infrared image of transmission lines insulators. Then, the image data are offloaded to edge servers or cloud server for real-time processing and analysis through terrestrial BSs, satellites, or other UAV relays. In addition, PIoT devices deployed on transmission lines offload monitoring data to servers through multi-mode terrestrial networks including optical fiber composite overhead ground wire (OPGW), wireless local area network (WLAN), and micropower wireless networks. Finally, the collected data are comprehensively analyzed to support real-time calculation of transmission line capacity expansion, icing detection, and electrical equipment fault positioning.

Electric emergency communication: Electric emergency communication supports rapid response to crisis, emergencies, and disasters. Based on the independent and disaster-resistant non-terrestrial networks, the robustness of electric emergency communication is enhanced greatly. In typical scenarios such as damaged device detection and positioning, UAVs act as relay nodes to construct communication links between terrestrial control station and PIoT devices for computation offloading. In addition, in real-time transmission of audio, video, and data information, UAVs and video subsystems are integrated into the traditional vehicle emergency satellite communication system to quickly build an independent private network, which supports the establishment of lingkage and consultation between the forward command and the remote command center to achieve efficient emergency repair.

B. Research Challenges of Computation Offloading in IPIoT

Coupling between task offloading and computation resource allocation: Task offloading reduces computation delay but results in increased edge-side and cloud-side queue backlogs and queuing delay, which imposes new challenge for computation resource allocation optimization. On the other hand, dynamically allocated computation resources cause the fluctuation of queue backlog information and task offloading reward, which in turn affects BS selection and computing paradigm selection in task offloading optimization.

Adverse impacts on synchronous FDRL due to heterogeneity and dynamics of IPIoT: The resource heterogeneity and dynamics in IPIoT cause differentiated training rates of local models. Synchronous FDRL-based federated averaging results in large training latency and poor convergence, caused by waiting for the model with the slowest training rate.

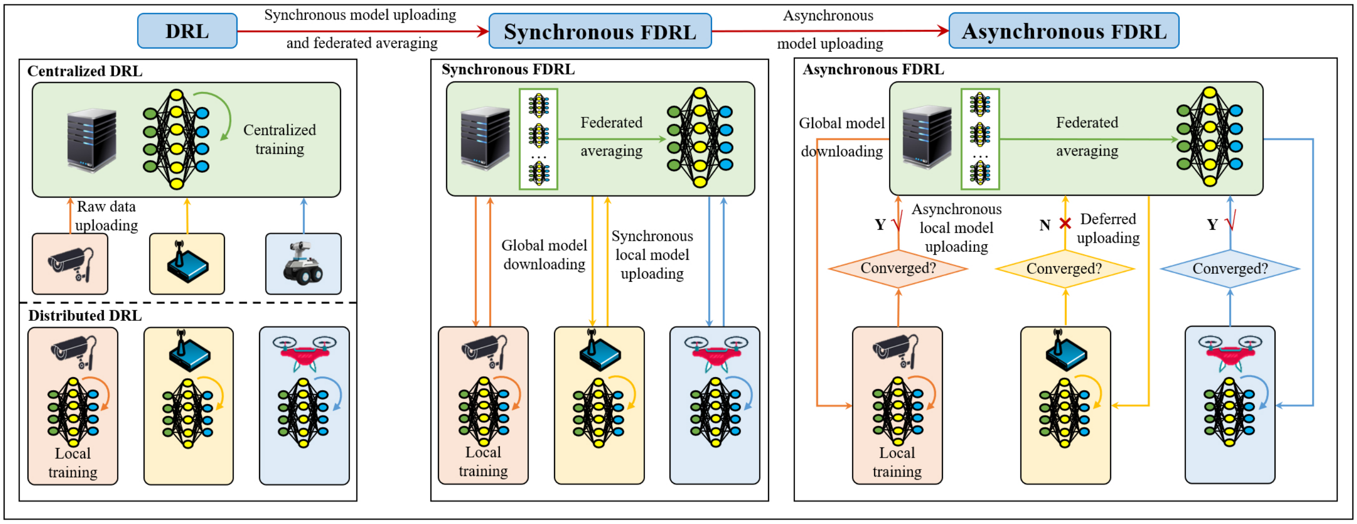


Fig. 1. Comparison of DRL, synchronous FDRL, and AFDRL.

Degraded training performance due to EMI: The sudden and strong EMI caused by partial discharge of high-voltage equipment such as transformers and switchboards deteriorates data transmission and makes local model training aberrant. Specifically, EMI causes errors in training data, resulting in aberrant fitting weights and inaccurate local models. Federated averaging based on aberrant local models leads to divergent global model and degraded training performance.

III. AFDRL-based Computation Offloading Framework

In this section, the fundamentals of AFDRL are firstly introduced. Then, the AFDRL-based computation offloading framework for IPIoT is elaborated. Afterwards, we explain the specific implementation procedure of computation offloading. Finally, the proposed ARTIST algorithm is illustrated.

A. Fundamentals of AFDRL

The comparisons among DRL, FDRL, and AFDRL are shown in Fig. 1, which are introduced as follows.

DRL: DRL combines the advantages of deep learning in approximating complex mapping relationships and reinforcement leaning in model-free decision making [10], [11]. DRL can be implemented in either centralized or distributed fashion. However, the centralized implementation faces the challenge of large communication overheads and security threats due to raw data exchange, whereas distributed one suffers from poor learning performance due to underutilization of adjacent environment observations [12].

FDRL: FDRL integrates FL and DRL to reduce communication overheads, relieve security threats, and exploit environment observations. However, FDRL requires synchronous local models uploading, which leads to large waiting delay for federated averaging under differentiated local model training rates. Moreover, synchronous FDRL inevitably reduces information freshness due to network congestion caused by simutaneous massive data uploading.

AFDRL: AFDRL is introduced to overcome the shortcomings of synchronous FDRL [13]. AFDRL allows devices to defer the uploading of inadequately trained local models, and only converged models are uploaded and utilized to train the global model, which reduces the waiting delay and total queuing delay. Three common AFDRL algorithms, i.e., asynchronous federated deep Q network (AFDQN), asynchronous federated deep actor critic (AFDAC), and asynchronous federated deep deterministic policy gradient (AFDDPG), are introduced.

- ***AFDQN***: AFDQN integrates the potentials of deep neural network (DNN) in learning and Q-learning in intelligent decision making [14]. Each device maintains a local evaluation network to learn the mapping relationship between state-action pair and estimated Q value, and a local target network to assist in evaluation network training. AFDQN employs experience replay to relieve the negative impact of correlation and non-stationary distribution of historical data on network training. The trained weights of local networks are uploaded asynchronously to a central controller. The central controller maintains a global evaluation network and a global target network to perform federated averaging and update local networks of corresponding devices through global network weight delivering.
- ***AFDAC***: In AFDAC, each device maintains a local actor network to draw actions and optimize policy based on policy-based DRL, and a local critic network to criticize and guide policy updating based on value-based DRL. The trained weights of local networks are uploaded asynchronously. The central controller maintains a global actor network and a global critic network, which perform similar functions as those of AFDQN.
- ***AFDDPG***: AFDDPG integrates experience replay and target networks of AFDQN with AFDAC. Specifically, each device maintains four local DNNs, i.e., an actor network, a target actor network, a critic network, and

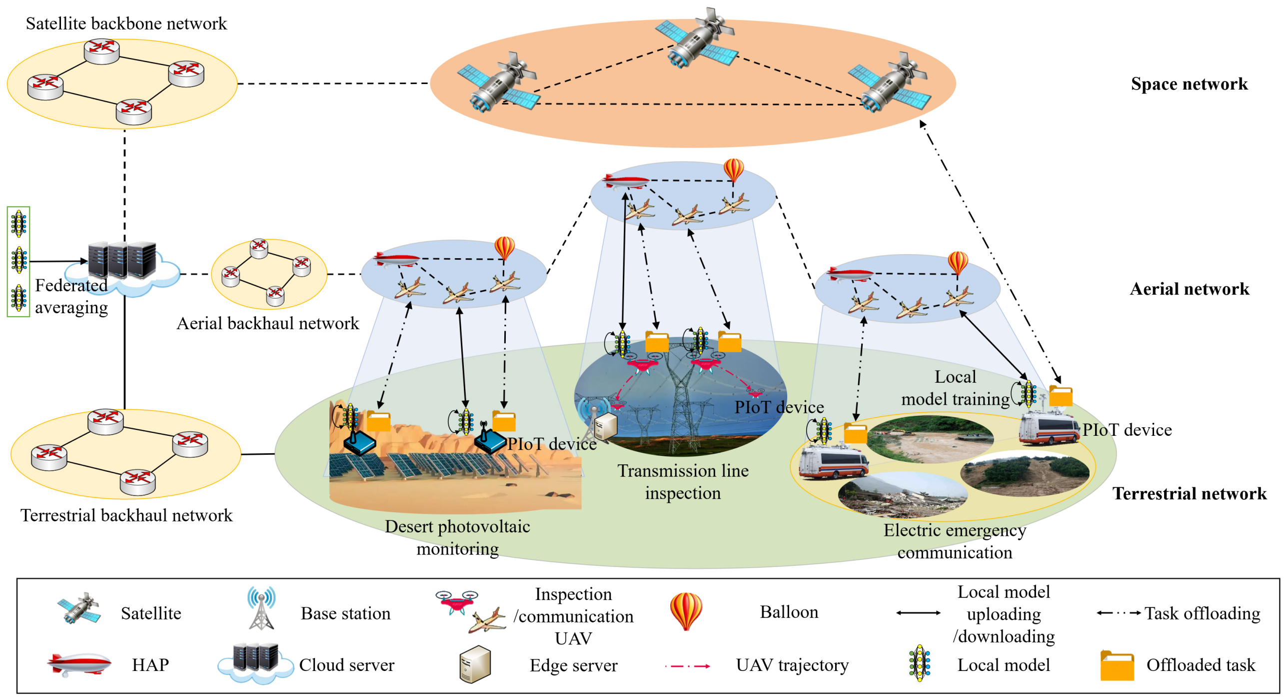


Fig. 2. AFDRL-based computation offloading framework for IPIoT.

a target critic network. The target actor network and target critic network assist in training of the actor network and the critic network, respectively. Similarly, the central controller maintains four global DNNs corresponding to the four local networks. After the training of the four local networks, devices upload the weights asynchronously to the central controller, which acts similarly as AFDQN and AFDAC.

B. AFDRL-based Computation Offloading Framework for IPIoT

The proposed AFDRL-based computation offloading framework for IPIoT is shown in Fig. 2. It consists of terrestrial and non-terrestrial networks.

Terrestrial network includes terrestrial BSs, edge servers, cloud servers, and PIoT devices. PIoT devices are deployed on electrical equipment such as PVs, wind turbines, and charging piles to collect different kinds of data, e.g., infrared image, voltage, current, temperature, and humidity information. Devices make task offloading decisions and offload generated computation tasks to edge servers or cloud server via terrestrial BSs or non-terrestrial networks. Devices train local models of task offloading optimization based on local information, and upload them to the cloud server asynchronously. BSs are co-located with edge servers to provide communication coverage and task computation. BSs also connect with the cloud server via terrestrial backhaul network. The cloud server performs federated averaging, and global model training. The computation resources of edge servers and cloud server are dynamically allocated according to data backlog and service requirements. The terrestrial network provides consistent coverage and powerful computation resources in areas with dense populations, but has the drawbacks of high deployment cost, poor flexibility, weak disaster resistance, and limited coverage in sparsely populated areas.

Non-terrestrial network is composed of aerial network and space network. The aerial network includes UAVs, HAPs, and balloons with miniature edge servers. They provide flexible computing services in isolated areas, but have the drawbacks of intermittent connectivity due to limited battery and uncoordinated trajectory. The space network includes geostationary (GEO), middle Earth orbit (MEO), and low Earth orbit (LEO) satellites. Devices upload and download models from the cloud server via satellites when terrestial and aerial networks are unavailable. The space network has larger latency due to the extremely long transmission distance between satellites and devices.

In order to implement computation offloading in IPIoT, the proposed architecture considers the fusion of terrestrial and non-terrestrial networks from two perspectives, i.e., task offloading and resource management. For task offloading, PIoT devices offload tasks to edge servers and cloud servers via terrestrial BSs and non-terrestrial UAVs and satellites by leveraging converged network protocol and compatible interface. For resource management, based on different geographical distribution characteristics of PIoT devices and differentiated service requirements, the multi-dimensional heterogeneous resources of communication, computing, and energy in IPIoT are aggregated into a unified resource pool and allocated collaboratively.

C. Implementation Procedure of Computation Offloading

The implementation procedure of computation offloading in IPIoT is shown in Fig. 3. We mainly elaborate task offloading

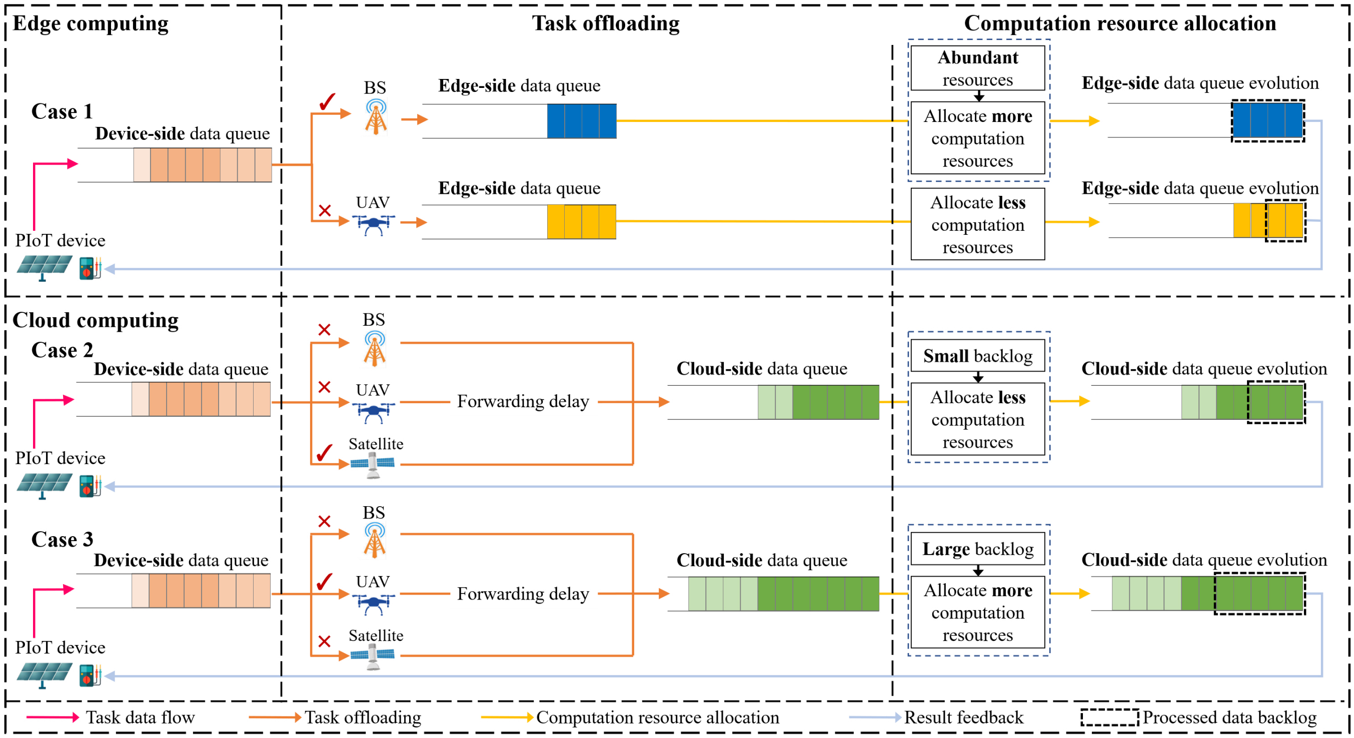


Fig. 3. Implementation procedure of computation offloading in IPIoT.

and computation resource allocation.

Task Offloading: PIoT devices collect data, generate various tasks, and store them in local buffers modeled as device-side data queues. Task offloading consists of BS selection and computing paradigm selection, where each device selects either terrestrial BS, UAV, or LEO satellite for data transmission, and selects edge computing or cloud computing for data processing. Three cases of task offloading are shown in Fig. 3. In Case 1, the device selects terrestrial BS and edge computing paradigm. Data are offloaded to the corresponding edge-side data queue maintained by the edge server. In Case 2 and Case 3, the device selects the cloud computing paradigm. Data are offloaded to the corresponding cloud-side data queues, mantained by cloud server via satellite and UAV, respectively.

Computation Resource Allocation: Edge and cloud servers allocate the available computation resources, i.e., CPU-cycle frequency, to process the offloaded data stored in the edge-side and cloud-side data queues. Computation resource allocation is dynamically adjusted according to internal and external factors such as available resources, data backlogs, and queuing delay requirements. Two cases are shown in Fig. 3. In Case 1, compared with UAV, terrestrial BS with abundant computation resources is selected to allocate more resources for task computation. In Case 3, the larger data backlog enforces the cloud server to allocate more computation resources to reduce queuing delay.

D. ARTIST

We propose the novel ARTIST algorithm to address the problem of low-latency computation offloading in IPIoT through joint optimization of task offloading and computation resource allocation. A slot model is adopted where network status, such as the location of UAVs and channel state information, remains constant within one slot. In each slot, PIoT devices make task offloading decisions including BS selection and computing paradigm selection, and edge servers and cloud server make computation resource allocation decisions. The task offloading and computation resource allocation is optimized based on the assumption of achieving time synchronization among all terrestrial and non-terrestrial entities. The specific time synchronization scheme will be investigated in future work.

The task offloading queuing delay is defined as the ratio of the device-side data backlog to the average data arrival rate. The edge-side and cloud-side queuing delays are defined similarly. Since a device's tasks may be offloaded to more than one edge server and even to, the cloud server, the task computation queuing delay is determined by the largest queuing delay among edge-side and cloud-side data queues. The low-latency computation offloading problem is formulated as follows.

Objective: The objective is to minimize the total queuing delay, which is defined as the sum of the task offloading queuing delay and task computation queuing delay, by jointly optimizing task offloading and computation resource allocation.

Task offloading constraint: Only one BS and one computing paradigm can be selected by each PIoT device in each slot.

Computation resource allocation constraint: The total

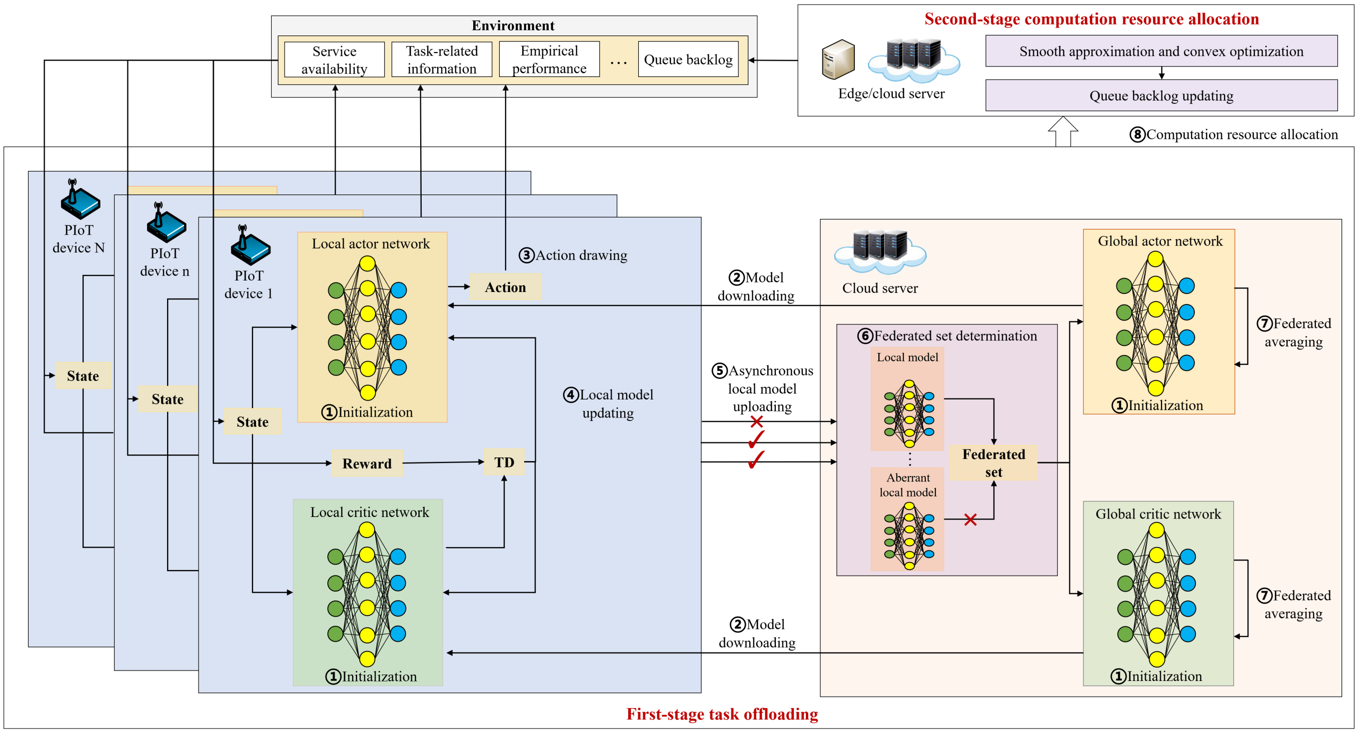


Fig. 4. Framework of ARTIST.

allocated computation resources of each server cannot exceed its upper bounds.

Queue stability constraint: The device-side and server-side data queues should remain mean rate stable.

Lyapunov optimization is leveraged to decouple the short-term optimization and the long-term queue stability constraint. Specifically, the joint optimization problem is converted into the device-side task offloading subproblem and server-side computation resource allocation subproblem through the minimization of the upper bound of the drift-plus-penalty. The drift-plus-penalty is calculated as the weighted sum of the total queuing delay and the one-step conditional Lyapunov drift, which is defined as the difference between the Lyapunov functions in adjacent slots.

The task offloading subproblem and the computation resource allocation subproblem are solved in two stages. The first-stage subproblem is modeled as a Markov decision process (MDP).

State: The state space consists of the data backlogs, task-related information such as empirical data arrival amount, and empirical network performance such as throughput, i.e., the amount of offloaded task data.

Action: The action space is the set of BS selection and computing paradigm selection strategies.

Reward: The reward is the negative of the upper bound of the drift-plus-penalty.

The devices asynchronously upload the converged local model for federated averaging, which avoids network congestion, ensures adequate training of local models and reduces waiting delay for global averaging. Moreover, cost deviation is adopted to detect aberrant local models caused by sudden and strong EMI. If the cost deviation exceeds a certain upper threshold, the corresponding local model is detected as an aberrant model, and is eliminated from federated averaging to increase training accuracy, improve convergence, and achieve EMI awareness.

The framework of ARTIST is shown in Fig. 4, and the implementation process is shown in Fig. 5. Details are explained as follows.

Initialization: Initialize the global critic network, global actor network, local critic network, and local actor network with random weights. The reward is initialized as zero.

First-stage task offloading optimization:

- *Model downloading:* At the beginning of each slot, devices download the global model from the cloud server and set it as local models.
- *Action drawing:* Each device draws the task offloading action according to the policy provided by the local actor network and the current state. Then, the device executes the action, updates the device-side data queue, and calculates the reward.
- *Local model updating:* Each device calculates the temporal-difference (TD) error based on the reward, discount factor, and state-action value. Then, based on the gradient descent method, the parameters of local actor model is updated based on the TD error, actor network learning rate, and policy score function, and the parameters of local critic model is updated based on TD error and critic network learning rate.
- *Asynchronous local model uploading:* A device requires a certain training period to achieve local model convergence. Each device determines whether the local model

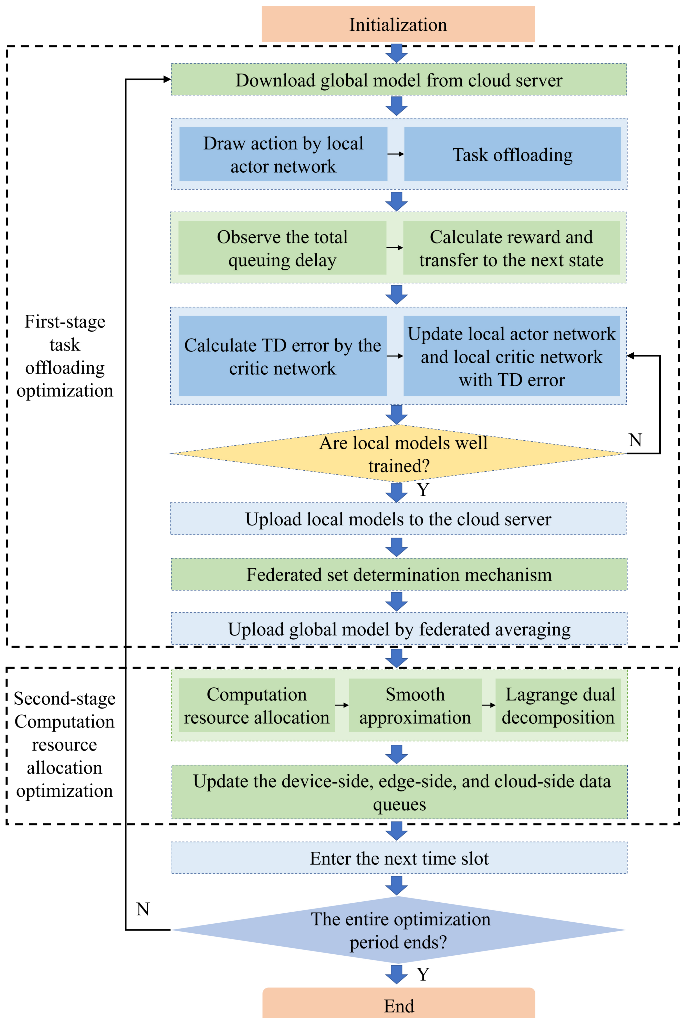
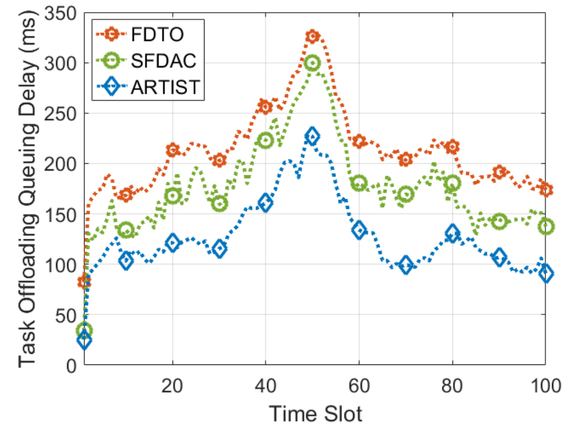


Fig. 5. Implementation process of ARTIST.

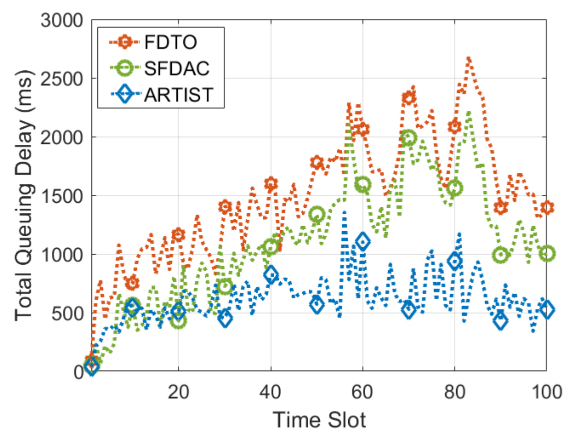
has been adequately trained and asynchronously uploads the converged local model to the cloud server.

- *Federated set determination:* Determine the federated set by removing aberrant local models whose cost deviations exceed upper thresholds to improve convergence and accuracy of global models.
- *Federated averaging:* The cloud server executes federated averaging based on the determined federated set and converged local models to update parameters of the global actor network and global critic network.

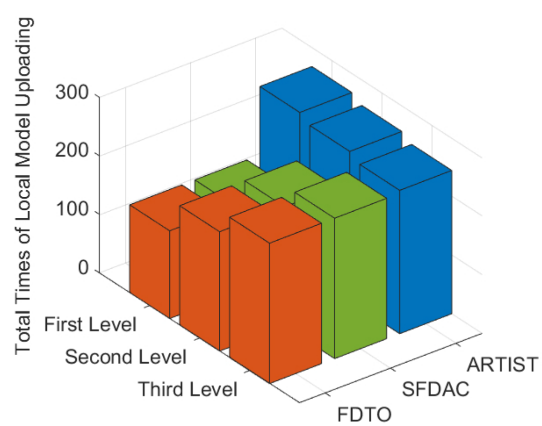
Second-stage computation resource allocation optimization: Based on the first-stage task offloading decision, each device offloads its task data to the corresponding edge server or cloud server. Servers observe the offloaded data backlogs and optimize computation resource allocation. The objective of the computation resource allocation subproblem is defined as the weighted sum of data backlogs and the minimum instantaneous queuing delay on the server side. Due to the existence of a minimization term, the computation resource allocation subproblem is non-convex. Therefore, we leverage a smooth function to approximate the minimization term and transform



(a) The task offloading queuing delay performance.



(b) The total queuing delay performance.



(c) The total times of local model uploading.

Fig. 6. Performance improvement of ARTIST.

the original subproblem into a convex one. The transformed subproblem is solved by Lagrange dual decomposition. Each server processes the task data based on allocated computation resources. The resulted task computation queuing delay and the server-side data backlogs are utilized to calculate the reward and update the state, which in turn affect next-slot local model training and task offloading decisions.

Enter the next slot and repeat the two-stage optimization until the total optimization time ends.

IV. Case Study

The performance improvement of ARTIST is verified in a case study. We consider a 400 m $\times$ 400 m area that contains 4 terrestrial BSs, 2 UAVs, 1 LEO satellite, and 12 PIoT devices. Devices are divided into three levels according to the descending order of local model training rates, i.e., first level, second level, and third level. The first-level devices have the fastest training rate. The numbers of first-level, second-level, and third-level devices are 3, 4, and 5, respectively. The coverage radii of terrestrial BSs and UAVs are 300 m and 100 m, respectively. The UAVs fly along a circle with a radius of 100 m, and the flight altitude is 90 m. The simulation parameter setting is based on [1], [4]. There are 100 slots with equal length 100 ms. The available computation resources of edge servers and cloud server are randomly distributed within [16, 36] GHz and [36, 56] GHz. EMI is modeled based on the symmetric α-stable (SαS) distribution, where the related parameters such as characteristic factor, skew parameter, scale parameter, and location parameter are set based on the electromagnetic environment. To verify EMI awareness, the power of EMI is increased suddenly during [30, 50] slots. Two comparison algorithms are adopted. The first one is the synchronous FDAC-based task offloading algorithm (FDTO) [15], which adopts a fixed computation resource allocation ratio. The second one is synchronous FDAC-based computation offloading algorithm (SFDAC) with EMI-aware federated set determination developed in this paper. Training-testing splitting is not considered in the three algorithms, which will be investigated in future work.

Fig. 6 (a) and (b) show the task offloading queuing delay and total queuing delay, respectively. Compared with FDTO and SFDAC, ARTIST reduces the task offloading queuing delay by 47.90% and 35.17%, and the total queuing delay by 54.91% and 46.25%, respectively. Due to the sudden and strong EMI, the device-side queuing delay of three algorithms increases during [30, 50] slots. Nevertheless, ARTIST has the lowest peak increment by discarding aberrant local models to mitigate the adverse impacts of EMI. Moreover, ARTIST outperforms SFDAC with EMI awareness due to asynchronous local model uploading, as shown in Fig. 6 (c).

Fig. 6 (c) shows the total times of local model uploading. Compared with FDTO and SFDAC, ARTIST improves the total times of first-level local model uploading by 78.30% and 77.82%, and the total times of second-level local model uploading by 25.87% and 25.49%, respectively. ARTIST allows devices to upload models asychronously. Federated averaging is performed only based on uploaded models without waiting for inadequately trained models.

V. Conclusion and Future Research

In this article, we addressed the low-latency computation offloading problem in IPIoT. An AFDRL-based computation offloading framework was proposed to improve the utilization of heterogeneous resources and promote optimal computation offloading decision making. Moreover, we proposed ARTIST to minimize the total queuing delay through the two-stage joint optimization of task offloading and computation resource allocation. Compared with FDTO and SFDAC, ARTIST reduces the task offloading queuing delay by 47.90% and 35.17%, and the total queuing delay by 54.91% and 46.25%. Finally, we identify some open research challenges and potential solutions.

Adverse impact of age of information (AoI) on model training and information freshness: AoI is an effective indicator to measure the freshness of key information. It represents the delay experienced by information from generation to being utilized for model training. A large AoI indicates a poor timeliness of information, resulting in high TD error and poor learning performance. A small AoI indicates a fresher information, which makes the model training more accurate. An effective solution is to jointly optimize AoI with other quality of service (QoS) metrics through multi-dimensional resource allocation and task offloading coordination.

Trust and security threats of computation offloading in IPIoT: The task data are exposed in an untrusted and opaque environment due to the complex affiliations of IPIoT devices. Some eavesdroppers may hack into servers for data tampering and theft, which seriously endangers the security of power systems. Blockchain provides a solution to ensure the trust and security concern in computation offloading optimization based on the distributed ledger, digital signature, and consensus mechanism. However, when the computation capacity of a consensus node is powerful enough to control the blockchain system, trust and security are difficult to be guaranteed. A potential research direction is to combine blockchain with advanced security trust technologies such as trusted computing, white list, multi-layered encryption, and access authentication mechanisms based on horizontal isolation and vertical authentication to ensure mutual trust and improve data integrity.

Acknowledgments

This work was supported by National Key R&D Program of China under Grant Number 2020YFB0905900, the National Natural Science Foundation of China under Grant Number 61971189, the open research fund of National Mobile Communications Research Laboratory, Southeast University (No. 2021D12), State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources under Grant Number LAPS2021-25, and 6G-XR project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101096838.
(Corresponding author: Zhenyu Zhou)

References

[1] H. Liao, *et al.*, "Cloud-Edge-End Collaboration in Air-Ground Integrated Power IoT: A Semi-Distributed Learning Approach," *IEEE Trans Ind. Informat.*, vol. 18, no. 11, Apr. 2022, pp. 8047-8057.

[2] B. Shang, *et al.*, “Computing over Space-Air-Ground Integrated Networks: Challenges and Opportunities,” *IEEE Network.*, vol. 35, no. 4, Aug. 2021, pp. 302-309.
[3] L. Bariah, *et al.*, “A Prospective Look: Key Enabling Technologies, Applications and Open Research Topics in 6G Networks,” *IEEE Access*, vol. 8, no. 99, Aug. 2020, pp. 174792-174820.
[4] H. Liao, *et al.*, “Blockchain and Semi-Distributed Learning-Based Secure and Low-Latency Computation Offloading in Space-Air-Ground-Integrated Power IoT,” *IEEE J. Sel. Top. Signal Process.*, vol. 16, no. 3, Apr. 2022, pp. 381-394.
[5] B. Mao, *et al.*, “Optimizing Computation Offloading in Satellite-UAV-Served 6G IoT: A Deep Learning Approach,” *IEEE Network*, vol. 35, no. 4, Aug. 2021, pp. 102-108.
[6] L. Bariah, *et al.*, “Toward Federated-Learning-Enabled Visible Light Communication in 6G Systems,” *IEEE Wireless Commun.*, vol. 29, no. 1, Feb. 2022, pp. 48-56.
[7] F. Liu, *et al.*, “Device Association for RAN Slicing based on Hybrid Federated Deep Reinforcement Learning,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, Dec. 2020, pp. 15731-15745.
[8] F. Pereira, *et al.*, “IoT Network and Sensor Signal Conditioning for Meteorological Data and Photovoltaic Module Temperature Monitoring,” *IEEE Lat. Am. Trans.*, vol. 17, no. 6, Jun. 2019, pp. 937-944.
[9] F. Yang, *et al.*, “ A Review on State-of-the-Art Power Line Inspection Techniques,” *IEEE Trans. Instrum. Meas.*, vol. 69, no. 12, Dec. 2020, pp. 9350-9365.
[10] F. Rahman, *et al.*, “Deep Reinforcement Learning based Computation Offloading and Resource Allocation for Low-Latency Fog Radio Access Networks,” *Intell. Converged Networks*, vol. 1, no. 3, Dec. 2020, pp. 243-257.
[11] M. Min, *et al.*, “Learning-Based Computation Offloading for IoT Devices With Energy Harvesting,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, Feb. 2019, pp. 1930-1941.
[12] F. Lu, *et al.*, “Edge QoE: Computation Offloading With Deep Reinforcement Learning for Internet of Things,” *IEEE Internet Things J.*, vol. 7, no. 10, Oct. 2020, pp. 9255-9265.
[13] B. Shi, *et al.*, “DARES: An Asynchronous Distributed Recommender System using Deep Reinforcement Learning,” *IEEE Access*, vol. 9, no. 99, Jun. 2021, pp. 83340-83354.
[14] C. Pan, *et al.*, “Asynchronous Federated Deep Reinforcement Learning-based URLLC-Aware Computation Offloading in Space-Assisted Vehicular Networks,” *IEEE Trans. Intell. Transp. Syst.* , vol. PP, no. 99, Feb. 2022, pp. 1-13.
[15] F. Zhu, *et al.*, “Federated Multiagent Actor-Critic Learning for Age Sensitive Mobile-Edge Computing,” *IEEE Internet Things J.*, vol. 9, no. 2, Jan. 2022, pp. 1053-1067.

BIOGRAPHIES

Sifeng Li (sifeng_li@ncepu.edu.cn) is currently working toward the M.S. degree at the State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources, School of Electrical and Electronic Engineering, North China Electric Power University. His research interests include power internet of things, resource allocation, and space-air-ground integrated network.

Sunxuan Zhang (sunxuan_zhang@ncepu.edu.cn) is currently working toward the Ph.D. degree at the State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources, School of Electrical and Electronic Engineering, North China Electric Power University. His research interests include power internet of things, cloud-edge-end collaboration, and network security.

Zhao Wang (zhao_w@ncepu.edu.cn) is currently working toward the Ph.D. degree at the State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources, School of Electrical and Electronic Engineering, North China Electric Power University. Her research interests include power internet of things, cloud-edge-end collaboration, and resource allocation.

Zhenyu Zhou (zhenyu_zhou@ncepu.edu.cn) received the Ph.D. degree from Waseda University, Japan in 2011. Since April 2019, he is a full professor at State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources, School of Electrical and Electronic Engineering, North China Electric Power University, China. His research interests mainly focus on power internet of things, smart grid communication, and cloud-edge-end collaboration. He is a senior member of IEEE, Chinese Institute of Electronics (CIE), and China Institute of Communications (CIC). He is an Associate Editor for IEEE Internet of Things Journal and IEEE Access. (***He is the corresponding author.***)

Xiaoyan Wang (Senior Member, IEEE) received the B.E. degree from Beihang University, China, and the M.E. and Ph.D. degrees from the University of Tsukuba, Japan. From 2013 to 2016, he worked as an Assistant Professor (by special appointment) with the National Institute of Informatics (NII), Japan. He is currently working as an Associate Professor with the Graduate School of Science and Engineering, Ibaraki University, Japan. His research interests include intelligent networking, wireless communications, cloud computing, big data systems, and security and privacy.

Shahid Mumtaz (dr.shahid.mumtaz@ieee.org) received the M.Sc. degree from the Blekinge Institute of Technology, Sweden, and the Ph.D. degree from the University of Aveiro, Portugal. He is currently with the Department of Applied Informatics, Silesian University of Technology, Akademicka, Gliwice, Poland, and the Department of Engineering, Nottingham Trent University. His research interests include MIMO techniques, multihop relaying communication, cooperative techniques, cognitive radios, game theory, and energy-efficient frameworks for 5G.

Mohsen Guizani (mguizani@ieee.org) received the M.S. and the Ph.D. degrees in Electrical and Computer engineering from Syracuse University, Syracuse, NY, USA. He is currently a professor and the associate provost at Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, UAE. His research interests include applied machine learning, smart city, wireless communications/networking, cloud computing, security and its application to healthcare systems. He was elevated to the IEEE Fellow in 2009. He is currently serving on the Editorial Boards of many IEEE Transactions and Magazines.

Valerio Frascolla (Member, IEEE) received the M.Sc. and the Ph.D. degrees in electronic engineering. He is currently the Director of research and innovation with Intel. He is the author of over 70 peer-reviewed publications. His main research interests include being 5G system design, with focus on machine learning algorithms, spectrum management, mmWaves, and edge technologies. He serves as a reviewer for several magazines and journals, as consortium lead for business, standardization, innovation, and exploitation matters in international research projects. He has a track record as an organizer of special sessions, workshops, and panels at flagship IEEE conferences.