CISTER
Research Centre in
Real-Time & Embedded
Computing Systems

Journal Paper

Employing Intelligent Aerial Data Aggregators for Internet of Things: Challenges and Solutions

Kai Li*

Wei Ni

Alam Noor*

Mohsen Guizani

*CISTER Research Centre

CISTER-TR-220401

2022

Employing Intelligent Aerial Data Aggregators for Internet of Things: Challenges and Solutions

Kai Li*, Wei Ni, Alam Noor*, Mohsen Guizani

*CISTER Research Centre

Polytechnic Institute of Porto (ISEP P.Porto)

Rua Dr. António Bernardino de Almeida, 431

4200-072 Porto

Portugal

Tel.: +351.22.8340509, Fax: +351.22.8321159

E-mail: kai@isep.ipp.pt, Wei.Ni@data61.csiro.au, alamn@isep.ipp.pt

https://www.cister-labs.pt

Abstract

Internet-of-Things (IoT) devices equipped with temperature and humidity sensors, and cameras are increasingly deployed to monitor remote and human-unfriendly areas, e.g., farmlands, forests, rural highways or electricity infrastructures. Aerial data aggregators, e.g., autonomous drones, provide a promising solution for collecting sensory data of the IoT devices in human-unfriendly environments, enhancing network scalability and connectivity. The flexibility of a drone and favourable line-of-sight connection between the drone and IoT devices can be exploited to improve data reception at the drone. This article first discusses challenges of the drone-assisted data aggregation in IoT networks, such as incomplete network knowledge at the drone, limited buffers of the IoT devices, and lossy wireless channels. Next, we investigate the feasibility of onboard deep reinforcement learning-based solutions to allow a drone to learn its cruise control and data collection schedule online. For deep reinforcement learning in a continuous operation domain, deep deterministic policy gradient (DDPG) is suitable to deliver effective joint cruise control and communication decision, using its outdated knowledge of the IoT devices and network states. A case study shows that the DDPG-based framework can take advantage of the continuous actions to substantially outperform existing non-learning-based alternatives.

Employing Intelligent Aerial Data Aggregators for Internet of Things: Challenges and Solutions

Kai Li, *Senior Member, IEEE,* Wei Ni, *Senior Member, IEEE,* Alam Noor, *Student Member, IEEE,* and Mohsen Guizani, *Fellow, IEEE*

***Abstract*—Internet-of-Things (IoT) devices equipped with temperature and humidity sensors, and cameras are increasingly deployed to monitor remote and human-unfriendly areas, e.g., farmlands, forests, rural highways or electricity infrastructures. Aerial data aggregators, e.g., autonomous drones, provide a promising solution for collecting sensory data of the IoT devices in human-unfriendly environments, enhancing network scalability and connectivity. The flexibility of a drone and favourable line-of-sight connection between the drone and IoT devices can be exploited to improve data reception at the drone. This article first discusses challenges of the drone-assisted data aggregation in IoT networks, such as incomplete network knowledge at the drone, limited buffers of the IoT devices, and lossy wireless channels. Next, we investigate the feasibility of onboard deep reinforcement learning-based solutions to allow a drone to learn its cruise control and data collection schedule online. For deep reinforcement learning in a continuous operation domain, deep deterministic policy gradient (DDPG) is suitable to deliver effective joint cruise control and communication decision, using its outdated knowledge of the IoT devices and network states. A case study shows that the DDPG-based framework can take advantage of the continuous actions to substantially outperform existing non-learning-based alternatives.**

***Index Terms*—Autonomous Drone, Internet of Things, Data aggregation, Cruise control, Deep reinforcement learning**

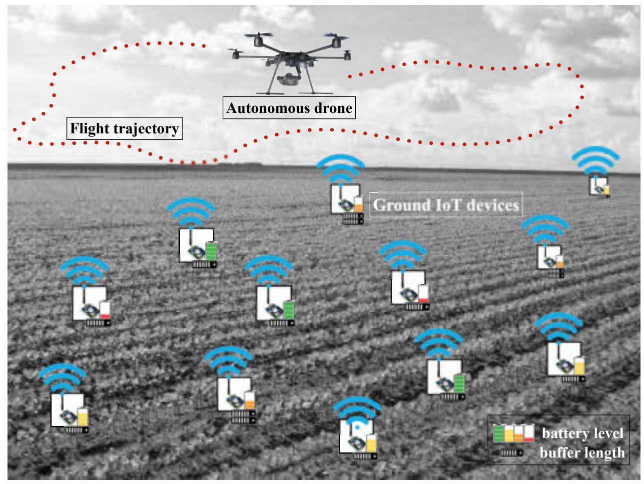


Fig. 1: Energy-harvesting-powered IoT devices can be deployed in a remote farm to monitor crop growth and control the growing environment. The autonomous drone can move around the IoT devices for data aggregation. The IoT devices can be scheduled to send sensing results towards the drone.

I. Drone-assisted Internet of Things

Energy-harvesting-powered Internet-of-Things (IoT) devices are increasingly deployed on farmlands for precision agriculture [1], remote highways for road surveillance [2], electricity infrastructures for structural health monitoring, or forests for environmental sensing. Autonomous drones can serve as aerial data aggregators to collect sensing data from geo-distributed IoT devices, hence extending the coverage of IoT networks to remote and human-unfriendly environments [3]. Figure 1 depicts an example of drone-assisted data aggregation, where energy-harvesting-powered IoT devices are deployed in a remote farm to monitor crop growth for pest detection and yield prediction. The IoT device can buffer sensory data in the queue. For the data aggregation, the drone can patrol and physically approach a ground IoT device to achieve a line-of-sight (LoS) air-ground connection, thereby enabling a high data rate under all terrains and saving the transmission energy of the IoT devices. The drone can perform beamforming to enhance the signal-to-noise ratio (SNR) and reduce the bit error rate at the drone. To connect a broad range of IoT devices, the drone's flight trajectory that is composed of waypoints can be designed to provide a full coverage of the IoT devices. The drone may adjust its moving directions and patrol velocities in real-time, while flying along the trajectory.

Low-power wide-area network (LPWAN) is a long-range wireless transmission protocol, where the communication range can be up to 10 km at the expense of low data rates (ranging from 0.3 to 50 kbits/s) and consequently high communication latency [4]. Take LoRaWAN for example. The data rate is up to 50 kbits/s. It would take 1600 seconds (or 26.67 minutes) for LoRaWAN to transmit 10 Mbytes of data. For an IoT device which is around 10 km away from the receiver, the data rate can be as low as 300 bits/s. It would take 74 hours to complete the transmission. On the contrary, the drone is able to move close to an IoT device by taking advantage of its excellent agility and maneuverability to shorten the communication distance. The drone-IoT LoS communication can benefit from the excellent channel gain for high-speed data transmission. Take onboard Wi-Fi with

K. Li and A. Noor are with Real-Time and Embedded Computing Systems Research Centre (CISTER), Porto, Portugal (E-mail:{kai,alamn}@isep.ipp.pt).

W. Ni is with the Digital Productivity and Services Flagship, Commonwealth Scientific and Industrial Research Organization (CSIRO), Sydney, Australia (E-mail: wei.ni@data61.csiro.au).

M. Guizani is with Computer Science and Engineering Department, Qatar University, Qatar (E-mail: mguizani@ieee.org).

a bandwidth of 80 MHz for example. The data rate can be up to 433 Mbits/s, and it only takes 0.18 second to transfer 10 Mbytes of data from an IoT device to the drone. Given a typical drone's average speed of 72 km/h, the drone can collect data from the IoT devices which are 10 km away from the base station within about 500 seconds. In this sense, the use of a drone can substantially improve the network throughput and coverage. The use of the drone also allows for fast transfer of large data files, and the substantially reduced energy consumption and data backlogs of the IoT devices.

When the drone receives some urgent, high-priority data from the IoT devices, the drone can immediately forward the data to a remote base station. Otherwise, the data can be buffered at the drone, and offloaded to the base station when the drone returns or passes the base station. In most cases, the urgent data requires fast response. The remaining, less critical data with large sizes can benefit from the short-range communication between the drone and the base station, when the drone passes the base station and enjoys an excellent channel condition.

In spite of consistent sampling intervals of the IoT sensors, the data generation of the IoT devices can be highly time-varying, resulting in dynamic arrivals of data at the buffer of an IoT device. The reason is that under many circumstances, the IoT device generates data only when sensing the changes in values, to reduce the signaling overhead of an IoT network and the power consumption of the transceiver [5]. In some applications, the data generation of an IoT device can be event-driven, for example, highway cameras in unpopulated regions take photos only when there are car accidents or emergency situations. Therefore, it is reasonable to consider that the sensory data of the IoT devices randomly arrives in their transmit buffers. On the other hand, data packets can be periodically generated at the IoT devices, leading to a predictable packet arrival. Nevertheless, the battery energy levels and channel gains can still be random and change over time due to the unknown environmental variation.

II. Joint Cruise Control and Communication Scheduling

A. Motivations and challenges

The typical objective of joint cruise control and communication scheduling is to maximize the amount of aggregated data at the drone. To achieve this, it is important to reduce both the data loss during data transmissions on the air interface, and the data loss resulting from the buffer overflow of the IoT devices. Despite the communication range, memory chips, storage and data compression capacity of IoT devices have been continuously improving, the data buffer of the IoT device can still overflow for two reasons.

- The first reason is that a data queue grows rapidly if its incoming data rate is greater than its outgoing rate, hence leading to a buffer overflow. This can be the case whenever there are a large number of devices in an IoT network or the trajectory of the drone is inadequately designed, in which case, the drone cannot collect data from the IoT devices in time. The buffers of the IoT devices would eventually overflow.
- The second reason is that emerging IoT platforms, comprising a considerable number of compact integrated IoT devices (e.g., miniatured cameras used to detect pests and wild animals [6], optoacoustic devices used to identify insect species [7], etc.), generate enormous data volumes. For instance, the size of a high-definition picture or audio/video clip is typically over a few megabytes. On the other hand, commercial-off-the-shelf IoT devices have finite data storage, constrained by cost and energy budgets.

Since an IoT device cannot perform data transmission when its battery level is flat, newly arrived data can potentially result in the buffer overflow of the IoT device. The IoT devices are equipped with rechargeable batteries to harvest renewable energy from ambient sources, such as solar and wind, and power their sensing and transmission operations. With a finite capacity, the batteries can overflow. The environmental conditions have a strong impact on the energy that can be harvested. In this sense, the battery levels of the IoT devices are an indispensable part of the environment variables, and can substantially affect the operation of the drone in the IoT network.

It is critical to improve the data aggregation by properly designing the cruise control of the drone and communication schedules, preventing buffer overflows of the IoT devices. Travelling salesman problem (TSP) can be used for the drone to find a cost-effective way of visiting all the IoT devices and returning to the starting point. In TSP, waypoints of the drone are predetermined according to the locations of IoT devices. Thus, the cost defined in TSP is deterministic to travel between any two IoT devices. In contrast, the problem of joint cruise control and communication schedules is distinctively different from the TSP. The reason is that any location in the target area could be a potential waypoint of the trajectory. The next waypoint, namely, the real-time patrol speed of the drone is determined by the cruise control scheme in the continuous domain. All the potential waypoints of the drone can be connected (or linked) into a daisy chain. To achieve this, the drone has to repeatedly fly over the target field to learn online the time-varying dynamics of packet arrivals, energy harvesting, and channel states of the IoT devices.

B. Bluetooth low energy enabled data aggregation

Several wireless transceivers can be used for the communication between the drone and the IoT device, e.g., Wi-Fi, Zigbee, or 4G. However, the drone-assisted IoT network with Wi-Fi or Zigbee can suffer from strong interference effects with other wireless mobile devices that concurrently function in 2.4 GHz. Mostly, the data traffic in a drone-assisted IoT network is bursty, while large overhead of Wi-Fi communications can result in low spectrum utilization. Direct sequence spread spectrum (DSSS) is used in

Zigbee for low-interference channel selection to alleviate the radio interference. However, the channel selection in Zigbee is static and does not adapt to the changing radio environments and interference situations. Zigbee can hardly be used in drone-assisted IoT networks, because of the high mobility of the drone and channel dynamics. Although the radio coverage of the drone can be extended by using 4G Long-Term Evolution (LTE), widespread telecommunication facilities are often unavailable in rural and unpopulated areas that have a limited or no network coverage.

Bluetooth Low Energy (BLE), *a.k.a.* Bluetooth 5.0, is a new Bluetooth Core Specification for long-distance communications, and can be considered for data aggregation in drone-assisted IoT networks [8]. BLE is based on frequency hopping spread spectrum (FHSS), which conducts carrier sensing to prevent jamming signals. The received signal can be restored at the receiver while the transmitter can determine the receiving sequence order. The drone-assisted IoT network based on BLE can transmit data in a high speed, given the data rate of 2 Mbps at most. Moreover, the radio coverage of BLE is up to 1 km. Thus, the drone can extend the connection duration with the IoT devices, thanks to the long communication distance of the BLE.

C. Contributions

As discussed in [9], deep reinforcement learning provides a solution to resource allocation or data collection in drone-assisted IoT networks. Deep reinforcement learning can be trained to minimize the propulsion energy consumption of a drone or maximize its spectrum efficiency. The authors of [10] apply deep reinforcement learning to the trajectory control and network resource management of a drone to enhance cellular performance. Deep reinforcement learning can enable the drone to improve the quality-of-service (QoS) of sensing and communication with no a-priori knowledge of the environment. Deep reinforcement learning can also be applied to the attitude control of the drone for tracking targets in unknown environments [11].

Distinctively different from the existing studies, this article addresses a new problem of joint cruise control and data collection in remote, human-unfriendly and vast areas, where one or multiple drones need to patrol and visit widespread IoT devices to collect data. The trajectory planning of the drones is critical to prevent the buffer overflows and the transmission failures of the IoT devices resulting from the untimely visits and the lossy airborne channels of the drones, respectively. The key contributions and novelties of this article are summarized, as follows.

- We address a new challenge of drone-assisted data aggregation in IoT networks, where a drone patrols and visits widespread IoT devices to collect data. The cruise control and data collection schedule need to be jointly designed to prevent the buffer overflows and the transmission failures of the IoT devices resulting from untimely visits and the lossy airborne channels of the drone, respectively.
- A new DDPG model is developed to achieve effective data aggregation by learning and refining the actions of cruise control and communication schedule on-the-fly in the absence of the complete knowledge about the network states. The action space of the new DDPG model consists of the continuous cruise control, and the discrete selection of IoT devices for data aggregation. The selection of IoT devices can be first relaxed to be continuous for model training and discretized for action taking.
- We demonstrate the new DDPG model on Google TensorFlow, one of the most widely adopted and accepted machine learning platforms. Numerical results show that the DDPG model can reduce the overall packet loss by at least 52%, as compared to existing non-learning heuristics. The performance evaluation on TensorFlow is conducive to further commercial development for real-world drone-assisted IoT systems.

TABLE I: Typical drone-assisted IoT networks and their deep reinforcement learning (DRL) solutions.

	Drone-assisted IoT networks	**DRL methodologies**
[9]	Spectrum access, data rate selection and transmit power control to minimize the propulsion energy of the UAV or maximize the spectrum efficiency.	Deep Q-Network (DQN) and double DQN (DDQN) with experience replay on channel and base station selection.
[10]	Trajectory planning and radio resource management to deliver the QoS to IoT devices in a cellular network.	DQN-based trajectory control and resource management, where the state is the location of the UAV and the action space contains the power management and channel allocation.
[11]	Attitude control of the drone to adjust the precision and accuracy of tracking targets.	Offline reinforcement learning (RL) is used to train accurate attitude controllers.
This paper	Joint cruise control and data collection scheduling of multiple drones to prevent the buffer overflows and the transmission failures of the IoT devices.	The new DDPG model is developed to achieve online cruise control and IoT device selection. Its state is made up of instantaneous coordinates and velocities of multiple drones, and time-varying channels and battery levels of many IoT devices.

Table I compares the typical drone-assisted IoT networks and deep reinforcement learning methodologies applied.

III. Intelligent Drone-Assisted IoT Networks Based on Deep Reinforcement Learning

A. POMDP for Cruise and Communication Control

At each waypoint along the flight cruise, the drone may not have the complete and up-to-date knowledge of all the IoT devices, e.g., data queue backlogs, channel conditions, and battery energy levels. Instead, the drone can only observe the environment over time, getting some clues of the actual underlying states. The online cruise control of the drone and the scheduling of the IoT transmission can be interpreted as a discrete-time Partial Observable Markov

Decision Process (POMDP), which aims to maximize the aggregated data and prevent the buffer overflows of the IoT devices and the packet delivery failure of the scheduled IoT devices. A state of the POMDP consists of a waypoint of the drone, and the battery levels and data backlogs of the IoT devices. The actions of the POMDP are the next waypoint of the drone (including the instantaneous heading and speed of the drone in Figure 2(a)), and the next IoT device selected to upload data to the drone. A reward is the aggregated data of all IoT devices.

Dynamic programming algorithms, such as value iteration or policy iteration, are typically used to solve POMDPs offline, provided that the a-priori knowledge of the state transition probabilities of the system is available. The optimal action-value function of the value/policy iteration is estimated and updated based on the Bellman optimality equation. The action-value function can be reinforced once the value/policy iteration converges.

It is important to consider a practical scenario, where the drone has no knowledge of the state transitions. This consideration is reasonable since the complete state information of the battery levels and data backlogs of the IoT devices is hardly instantaneously observable at the drone (due to typically less powerful radio of the IoT device). In this case, deep reinforcement learning can help the drone learn the optimal action of the POMDP on the fly at each network state together with the state transitions, while analyzing the action-value function online. As a result, the data aggregation of the network can be enhanced by adjusting the drone's heading, speed and selection of IoT devices, adapting to the bursty data and energy arrivals at the IoT devices and the time-varying channel conditions of the nodes.

B. DDPG-based continuous cruise control and communication schedules

Value decomposition networks (VDN) [12] and monotonic value function factorisation (QMIX) [13] have been used for deep reinforcement learning-based drone control. VDN learns a joint action-value function which is the sum of the action-value functions of all agents. Different from VDN, QMIX replaces the "sum" operation with a mixed network (the parameters of which are generated from the complete network state to ensure monotonicity). As a result, the action-value function can be generalized to a larger family of monotonic functions. However, VDN and QMIX are developed to learn the action-value function according to discrete state and action spaces. For example, the motion control of the drone can be discretized to five actions, i.e., {Up, Down, Left, Right, Hover}. In contrast, we consider the cruise control of the drone in the continuous domain in this article. The DDPG model that leverages actor-critic neural networks is trained onboard at the drone to learn the mixed actions of continuous flight trajectory and speed, and the discrete communication scheduling.

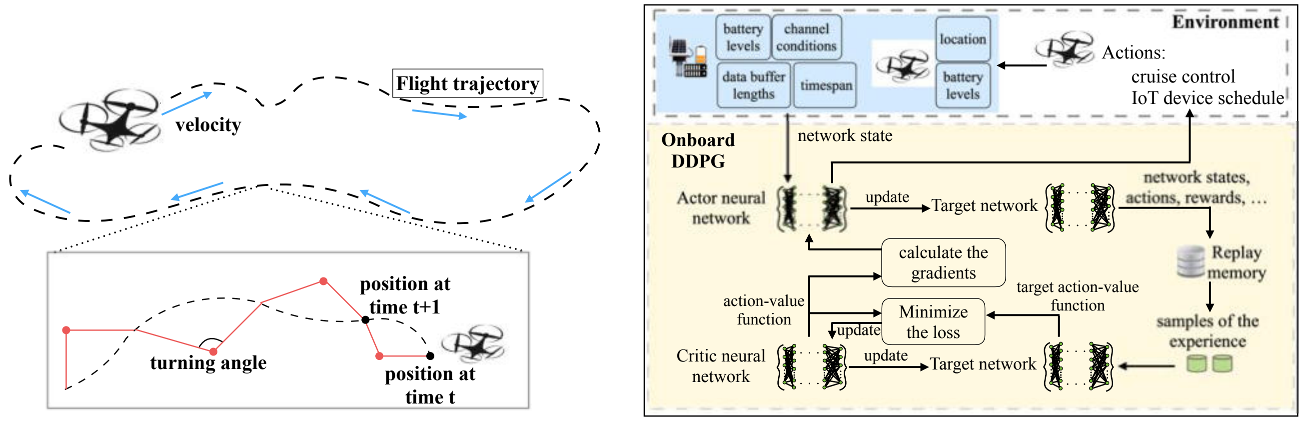
The POMDP problem is often tackled by a reinforcement learning approach, e.g., Q-learning. However, Q-learning is known to succumb to the problem of curse-of-dimensionality [14], making it unsuitable for cruise control and data aggregation because of too many states and actions.

An onboard DDPG-based cruise control and communication scheduling is developed, which can overcome the dimensionality issue of Q-learning. The system architecture is illustrated in Figure 2(b), where the state transitions of the POMDP are unknown to the drone. Specifically, the DDPG model leverages actor-critic neural networks to evaluate the instantaneous headings and velocities of the drone, as well as the data collection schedule of the IoT devices at every instant. A policy gradient scheme can be developed with the DDPG model to implement a stochastic behavior policy for exploring and estimating a deterministic target policy. The deterministic policy gradient allowes the policy to be updated by projecting the network state to the action. A replay memory can be used at the drone to store the training experience at each learning epoch, where mini-batches are taken as random samples of the learning experiences in the replay memory. Using the mini-batches, the DDPG model can be trained along with the network states. As a result, the aggregated data of all IoT devices (i.e., reward) can boost over the large, continuous state and action spaces.

At any moment, the drone can only observe the battery level and data queue backlog of the selected IoT device at a time. Nevertheless, the drone can exploit the training experience in the replay memory, and evaluate the buffer overflow and packet reception errors of different selections. Each experience log in the replay memory is stamped with its recording time, referred to as time-to-alive (TTA). TTA indicates the time lapse since the last time an IoT device was selected to upload its data to the drone. Based on the observed state knowledge of the selected IoT device and the TTAs of the unselected ones in the replay memory, the complete network state can be approximated at the drone. Accordingly, the network data loss can be evaluated and another historical learning experience can be recorded. The new DDPG model can conduct the experience replay to explore and exploit online the time-varying dynamics of the packet arrivals, energy harvesting, and channel conditions of the IoT devices.

In particular, the action space of the new DDPG model consists of the continuous cruise control, and the discrete selection of IoT devices for data aggregation. Since DDPG typically trains continuous actions, the selection of IoT devices can be first relaxed to be continuous for the purpose of model training and then discretized to facilitate action taking. At the beginning of the training phase, the relaxed discrete IoT device selection may give birth to classification errors, resulting in a high network cost. These classification errors can, though, be substantially reduced when the DDPG model is sufficiently trained.

Offboard deep reinforcement learning can be carried out offline, where cruise control and communication scheduling decisions are made for each waypoint along the trajectory before the drone takes off. Offboard deep reinforcement learning for offline cruise control and communication



(a) cruise model of the drone.

(b) actions of the drone can be trained online by the DDPG.

Fig. 2: The drone can adjust the turning angle to control its heading. Actions of the drone can be trained by the deep reinforcement learning-based cruise control and communication scheduling scheme.

schedules would require the drone to collect in prior the network states at each waypoint of the trajectory, i.e., the battery levels, data backlogs, and channel states of all the IoT devices. However, the a-priori knowledge of the network states is hardly acquirable. Therefore, it is of practical value to run the onboard deep reinforcement learning online for real-time cruise control and communication scheduling, where the decisions are adapted to the real-time network state dynamics at each of the waypoints. The DDPG-based cruise control and communication scheduling can perform online for training the instantaneous heading and velocity of the drone, as well as the data collection schedule of the IoT devices at every instant.

The DDPG-based cruise control and communication scheduling is promising, and can be further extended through the deployment and exploitation of drone swarms. It is possible that the drones do not communicate or collaborate with each other since the drones can be deployed in a wide target field, where the drones are out of communication range of each other. The drones have to individually control their instantaneous headings, patrol speeds, and data collection schedules, based on their independent observations of the network state. Each drone is expected to learn the actions of the other drones implicitly from changes in the battery levels and data backlogs of the IoT devices. To this end, the DDPG can be applied to allow each of the drones to act as an agent and learn this hidden Markov process. In this case, the drones can always maintain active communications with each other to coordinate their actions. The action of an agent is trained according to the network states, as well as the environmental changes resulting from the actions of the rest of the agents.

Note that the new DDPG model can be potentially trained offline in a simulator before being deployed in real environment. Once deployed, the DDPG model will continue the learning to refine the model parameters, if needed. By this means, the DDPG model can account for generic collision or obstacle avoidance via the offline training and can adapt to the specific real-world application scenario via online refinement. Additionally, the drones are typically equipped with vision-based techniques or utilize event cameras to avoid collisions and adjust the flight attitudes.

IV. Implementation and Performance

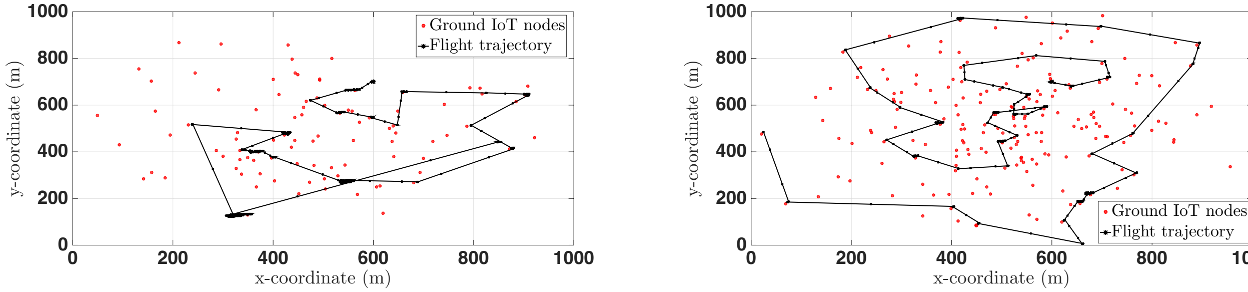
A. *Implementation of the DDPG-based strategy*

The new DDPG model is based on deep reinforcement learning techniques and can improve the data aggregation by learning and refining the actions of cruise control and communication schedule on-the-fly. The DDPG model can be implemented on Google TensorFlow, one of the most widely adopted and accepted machine learning platforms. The performance evaluation on TensorFlow is conducive to further commercial development for real-world drone-assisted IoT systems.

Using offline datasets for the performance evaluation is challenging in the context of the DDPG-based strategy, or more generally, deep reinforcement learning. The reason is that deep reinforcement learning interacts with the environment and makes decisions which can lead to changes in the environment. In the case of the DDPG-based strategy, the decision of the drone on the trajectory and IoT device selection affects the queue and battery statuses of both the selected and unselected devices. To this end, a static real-world dataset which does not interact with the drone or respond to the drone's decisions, would not be adequate to evaluate the DDPG-based strategy.

B. *Numerical analysis*

Figure 3 demonstrates the flight trajectory of the drone according to different deployments of the IoT devices. In Figure 3(a), 100 IoT devices are randomly deployed in the area of interest with the size of 1000 m $\times$ 1000 m. As observed, the new DDPG-based strategy progressively adjusts the trajectory of the drone, where the actions of the heading and the instantaneous patrol speed of the drone are carried out in the continuous action space. In Figure 3(b), 200 IoT devices are deployed in the area. The drone has



(a) 100 IoT devices in a 1000 m × 1000 m field.

(b) 200 IoT devices in a 1000 m × 1000 m field.

Fig. 3: The flight trajectory of the drone according to deployments of the IoT devices.

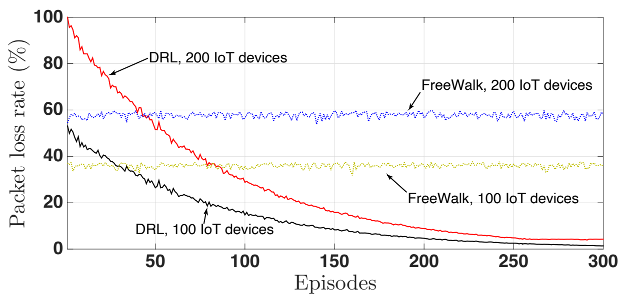


Fig. 4: Performance of the new DDPG-based strategy with regards to the training episodes.

to maneuver over more IoT devices in a wider range to enhance the data aggregation and reduce the packet loss pertaining to buffer overflows and channel fading.

Figure 4 shows the packet loss rate under the new DDPG-based strategy with regards to the training episodes. In particular, an "episode" contains a series of consecutive training epochs, where the onboard DDPG is trained to produce the optimal actions. As shown in Figure 2, the DDPG-based framework executes actions, forecasts the next states, and updates the actor policy in every episode. Figure 4 also shows the impact of the number of IoT devices on the network cost of the new DDPG-based framework, and the convergence of the framework. With the increasing number of episodes (or in other words, the training time), the network cost drops significantly and then stabilizes.

In Figure 4, the DDPG-based strategy is compared with FreeWalk in which the drone maneuvers randomly over the area of interest [15]. It can be observed that the DDPG-based strategy outperforms FreeWalk by 52% and 31% in the presence of 200 and 100 IoT devices, respectively, once the actions of the drone are sufficiently trained. The reason is that the drone under FreeWalk does not adapt its flight trajectory to prevent buffer overflows at the IoT devices. In contrast, the DDPG-based strategy learns the battery levels, data queue backlogs, and channel states of the IoT devices; and comes up with the joint cruise control and communication decision to minimize the data loss of the entire network over a long time horizon.

V. Conclusions and Future Directions

This article studied the challenges of real-time continuous cruise control and data aggregation in drone-assisted IoT networks, as well as the deep reinforcement learning solutions. A new architecture of DDPG-based continuous cruise control and communication scheduling was presented, where the actions of the drone are trained online based on the incomplete and potentially outdated knowledge of the drone on the network states. The onboard DDPG-based strategy can be implemented on Google TensorFlow. Performance analysis showed that the drone can maneuver to enhance the data aggregation and reduce packet loss pertaining to buffer overflows and channel fading. The impacts of the number of IoT devices on the network cost of the DDPG-based framework and the convergence of the framework were also discussed in terms of network cost.

As future research directions, the DDPG-based continuous cruise control and communication scheduling will be extended to drone swarm scenarios, where multiple drones can cooperate for one mission. Multi-agent DDPG can be individually trained on each of the drones (i.e., agents) based on independent state observation, to decide a joint action on collision-free cruise control and interference-free communication schedules for all the drones. In addition, a testbed of the drone-assisted IoT will be built. Experimental measurements will be collected on the testbed to validate the performance of the presented DDPG model.

Here, it is important to mention that, within a particular network state, the action of a drone not only determines the next state but also has a negligible impact on the actions of the other drones. As a result, the network state observed by a drone can be quickly outdated because of the actions that the other drones take in the meantime. Consequently, multi-agent DDPG could undergo slower convergence. A potential remedy is to share the observations and actions among all drones, so that a joint action can be trained for all the agents. However, this would require the drones to maintain consistent and reliable wireless connections, which is challenging in practice. Further to that, the enhancement of data aggregation with different drone swarm mobility patterns will be also considered in a future work.

ACKNOWLEDGEMENTS

This work was supported in part by the National Funds through FCT/MCTES (Portuguese Foundation for Science and Technology), within the CISTER Research Unit under Grant UIDP/UIDB/04234/2020, and in part by the National Funds through FCT, under CMU Portugal Partnership under Project CMU/TIC/0022/2019 (CRUAV).

REFERENCES

[1] F. Yang, L. Shu, K. Huang, K. Li, G. Han, and Y. Liu, "A partition-based node deployment strategy in solar insecticidal lamps internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 11, pp. 11 223–11 237, 2020.

[2] F. Ren, L. Zhang, and H. Li, "IoT-enabled highway safety pre warning system," in *IEEE/ACM International Conference on Internet-of-Things Design and Implementation (IoTDI)*, 2019, pp. 283–284.

[3] N. H. Motlagh, T. Taleb, and O. Arouk, "Low-altitude unmanned aerial vehicles-based internet of things services: Comprehensive survey and future perspectives," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 899–922, 2016.

[4] W. Xu, J. Y. Kim, W. Huang, S. S. Kanhere, S. K. Jha, and W. Hu, "Measurement, characterization, and modeling of lora technology in multifloor buildings," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 298–310, 2019.

[5] T. Qiu, R. Qiao, M. Han, A. K. Sangaiah, and I. Lee, "A lifetime-enhanced data collecting scheme for the internet of things," *IEEE Communications Magazine*, vol. 55, no. 11, pp. 132–137, 2017.

[6] A. R. Elias, N. Golubovic, C. Krintz, and R. Wolski, "Where's the bear?-automating wildlife image processing using iot and edge cloud systems," in *IEEE/ACM International Conference on Internet-of-Things Design and Implementation (IoTDI)*. IEEE, 2017, pp. 247–258.

[7] U. Garlando, L. Bar-On, A. Avni, Y. Shacham-Diamand, and D. Demarchi, "Plants and environmental sensors for smart agriculture, an overview," in *IEEE Sensors*. IEEE, 2020, pp. 1–4.

[8] K. Li, N. Lu, P. Zhang, W. Ni, and E. Tovar, "Multi-drone assisted internet of things testbed based on bluetooth 5 communications," in *2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 2020, pp. 345–346.

[9] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3133–3174, 2019.

[10] J. Hu, H. Zhang, L. Song, Z. Han, and H. V. Poor, "Reinforcement learning for a cellular internet of UAVs: Protocol design, trajectory control, and resource management," *IEEE Wireless Communications*, vol. 27, no. 1, pp. 116–123, 2020.

[11] W. Koch, R. Mancuso, R. West, and A. Bestavros, "Reinforcement learning for UAV attitude control," *ACM Transactions on Cyber-Physical Systems*, vol. 3, no. 2, pp. 1–21, 2019.

[12] Y. Hu, M. Chen, W. Saad, H. V. Poor, and S. Cui, "Distributed multi-agent meta learning for trajectory design in wireless drone networks," *IEEE Journal on Selected Areas in Communications*, 2021.

[13] T. Rashid, M. Samvelyan, C. Schroeder, G. Farquhar, J. Foerster, and S. Whiteson, "Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2018, pp. 4295–4304.

[14] Y. Duan, X. Chen, R. Houthooft, J. Schulman, and P. Abbeel, "Benchmarking deep reinforcement learning for continuous control," in *International conference on machine learning*. PMLR, 2016, pp. 1329–1338.

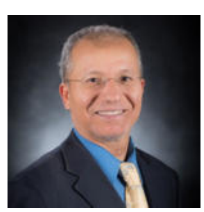
[15] P. Sun and A. Boukerche, "Performance modeling and analysis of a UAV path planning and target detection in a UAV-based wireless sensor network," *Computer Networks*, vol. 146, pp. 217–231, 2018.

VI. ABOUT THE AUTHORS

Kai Li [S'09-M'14-SM'20] (kai@isep.ipp.pt) is a senior research scientist and project leader at CISTER Research Centre, Portugal. He is currently a research fellow with Carnegie Mellon Portugal Research Program, which is jointly supported by Carnegie Mellon University, U.S., and FCT Portugal. He serves as an Associate Editor for Elsevier Ad Hoc Networks Journal, Elsevier Computer Communications Journal, and IEEE Access Journal.

Wei Ni [M'09-SM'15] (wei.ni@data61.csiro.au) received his B.E. and Ph.D. degrees from Fudan University, Shanghai, China, in 2000 and 2005, respectively. He is a Principal Research Scientist, Information Security and Privacy Group, CSIRO, Australia, and an Adjunct Professor at the University of Technology Sydney and Honorary professor at Macquarie University. He serves as the Chair of the IEEE VTS NSW Chapter, and an Editor for IEEE Transactions on Wireless Communications and IEEE Transactions on Vehicular Technology.

Alam Noor (alamn@isep.ipp.pt) is a Researcher at CISTER Research Center, Portugal, and a Ph.D. candidate at the University of Porto, Portugal. His research interests include digital signal and image recognition, deep learning, and data processing for real-time applications.



Mohsen Guizani [S'85-M'89-SM'99-F'09] (mguizani@ieee.org) received the B.S. (with distinction) and M.S. degrees in electrical engineering, and the M.S. and Ph.D. degrees in computer engineering from Syracuse University, Syracuse, NY, USA, in 1984, 1986, 1987, and 1990, respectively. He is currently a professor at the CSE Department at Qatar University, Qatar. He is an IEEE Fellow and a Senior Member of ACM.