Deep Reinforcement Learning for Persistent Cruise Control in UAV-aided Data Collection

Harrison Kurunathan*, Kai Li*§, Wei Ni†, Eduardo Tovar*, Falko Dressler‡
*CISTER-ISEP, Porto, Portugal, †CSIRO, Sydney, Australia, ‡TU Berlin, Germany

***Abstract*—Autonomous UAV cruising is gaining attention due to its flexible deployment in remote sensing, surveillance, and reconnaissance. A critical challenge in data collection with the autonomous UAV is the buffer overflows at the ground sensors and packet loss due to lossy airborne channels. Trajectory planning of the UAV is vital to alleviate buffer overflows as well as channel fading. In this work, we propose a Deep Deterministic Policy Gradient based Cruise Control (DDPG-CC) to reduce the overall packet loss through online training of headings and cruise velocity of the UAV, as well as the selection of the ground sensors for data collection. Preliminary performance evaluation demonstrates that DDPG-CC reduces the packet loss rate by under 5% when sufficient training is provided to the UAV.**

Index terms: UAV-aided WSN, Autonomous UAV, Cruise control, Deep reinforcement learning

I. Introduction

Aerial data collection using unmanned aerial vehicle (UAV) is becoming one of the major building blocks in wireless sensor networks due to its flexible deployment in harsh environments like disaster areas [1]. In such areas, steady power supply is limited and human intervention is not reliable [2]. A critical challenge in the aerial data collection with the UAV is the packet loss resulting from buffer overflows at the ground sensors and channel fading [3].

Fig. 1 depicts a typical UAV-assisted wireless Sensor Network in precision agriculture, where, a UAV hovers over a field for a finite cruise time. The data from the ground sensors are collected along the flight trajectory. These ground sensors can be equipped with solar panels to harvest renewable energy. In general, energy harvesting is highly dependent on weather conditions, e.g., a rainy or cloudy day, and is not reliable. A reduced battery level of the ground sensor can prevent data in the finite buffers from being transmitted, resulting in buffer overflow. Newly arrived packets at the ground sensors have to be discarded due to the buffer overflow.

Buffer overflows at the ground sensor can be due to two major instances: Firstly, when the network has too many sensors for data collection or when the trajectory of the UAV is not carefully planned. Secondly, when large data, such as high-resolution images or large size acoustic data are generated at the ground sensors, it can result in a buffer overflow due to the lack of enough data storage. Careful selection of the ground sensors can minimize the packet loss resulting from buffer overflows.

§ Corresponding author

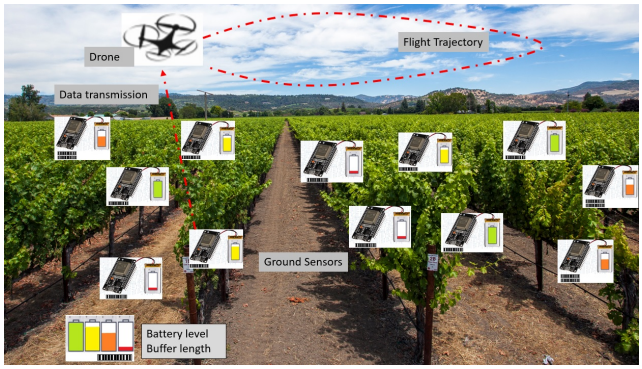

Fig. 1. UAV-assisted agriculture monitoring, where a UAV is employed to collect sensory data of the ground sensors

In [4], trajectory planning is used to reduce the communication delay by determining waypoints and alleviating data traffic congestion in accordance with the data buffer occupancy at the UAV. Several machine learning techniques, such as [9] and [11], have been used to reduce the packet loss by designing the trajectory. Considering energy consumption, it has been studied that by adjusting the flight velocity and the trajectory, it is possible to improve the energy efficiency of the UAV [5].

The trajectory planning algorithm in [6] improves the quality of service given the UAV's initial location, final destination, velocities, and maximum speed. The current research towards trajectory planning is moving towards UAVs dynamically adapting their displacement direction and distance to serve the ground users. Trajectory planning has also been used in applications where the UAV has to define strategic headings to support an extended cruise time [10], [7]. In such applications, machine learning helps in defining the actions of the UAV based on the environmental data. These algorithms have been used to minimize the communication delay of data collection. In [12], the communication delay between the UAV and the ground nodes can be reduced by proper trajectory planning and the communication scheduling.

In this paper, a joint optimization of trajectory planning for the UAV and selection of the ground sensors is proposed to minimize the overall packet loss in a UAV-assisted sensor network, where the headings and velocities of the UAV are optimized in a continuous action space.

The contributions of this work are:

- An onboard Deep Deterministic Policy Gradient based

Cruise Control (DDPG-CC) is proposed to optimize the continuous cruise control of the UAV. The onboard DDPG-CC jointly optimizes the online cruise control and communication schedule through online training of the UAV in terms of instantaneous headings, patrol velocities, and the real-time selection of the transmitting ground sensors.

- Our preliminary performance analysis demonstrates that the proposed DDPG-CC method can highly reduce the overall packet loss of the UAV aided wireless sensor network. Despite the variance in the number of nodes put to test, the packet loss rate decreases steadily and eventually converges with less that 5% packet loss around 300 learning episodes.

II. Autonomous Flight and Channel Model

In this section, we present the autonomous flight and the channel model. The flight model helps us in understanding how the proposed DDPG-CC impacts the velocities and the positions of the flight. The channel model determines the probability of the loss of sight between the UAV and the ground sensors during data collection.

A. Flight model of the autonomous UAV

In Figure 2, $(X(t), Y(t), Z)$ denotes the position of the UAV at a time t. The UAV moves with a speed of $S(t)$, where $S(t)$ varies between S_{min} (the minimum required velocity) and S_{max} (the maximum required velocity). $\Delta(t)$ is the flight duration from $(X(t), Y(t), Z)$ to $(X(t+1), Y(t+1), Z)$, the acceleration of the UAV can be given as $\Delta S(t)/\Delta t = (S(t+1) - S(t))/\Delta t$, where $0 \leq \Delta S(t)/\Delta t \leq (S_{max} - S_{min})/\Delta t$

The proposed DDPG-CC can evaluate the tangential acceleration of the UAV, i.e., $\Delta S(t)/\Delta t$, according to $\Delta S(t)/\Delta t \leq (S_{max} - S_{min})/\Delta t$. Since $\Delta S(t)/\Delta t$ has to be below the required maximum acceleration $(S_{max} - S_{min})/\Delta t$, we can obtain $(X(t+1), Y(t+1), Z)$ and $S(t+1)$. By specifying the rotation center and radius, the heading at the next location, i.e., $\theta(t+1)$, can be determined subsequently.

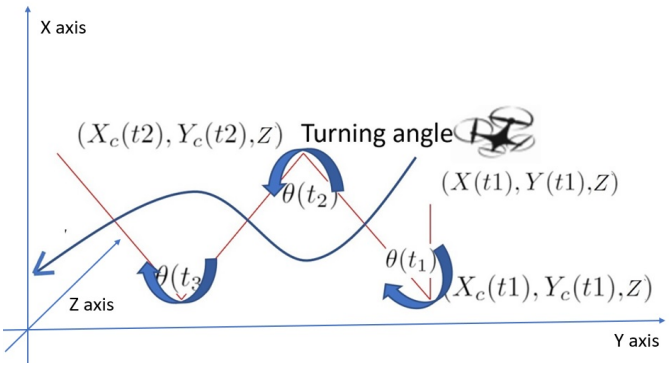

Fig. 2. The flight model of the autonomous UAV

Considering the coordinates at the rotation center is $(X_c(t), Y_c(t), Z)$, the UAV's coordinates at $t+1$ can be given as:

$$x(t+1) = x_c(t) - [(x(t) - x_c(t))cos\theta(t) - (y(t) - y_c(t))cos\theta(t)] \quad (1)$$

$$y(t+1) = y_c(t) - [(x(t) - x_c(t))cos\theta(t) - (y(t) - y_c(t))cos\theta(t)] \quad (2)$$

Instantaneous headings and cruise velocities are adapted online using our DDPG-CC framework. The UAV has to return to a command center to charge when the energy $e_{\text{UAV}}(t)$ drops below the minimum energy level $e_{\text{UAV}}^{\min}$. Additionally, beamforming can be enabled at the UAV to improve the received signal strength (RSS), eventually reducing the bit error rate (BER) of the air-ground communication.

B. Channel Model

For collecting the data from N ground sensors, the UAV moves near the ground sensors. The probability of line of sight (P_{ls}) (denoted in [13]) between the UAV and the ground sensors can be given by:

$$P_{ls} = \frac{1}{1 + a\exp(-b)[\varrho_i(t) - a]} \quad (3)$$

where a and b are the two Sigmoid parameters and ϱ_i the elevation angle between the UAV and sensor i at time t. The path-loss between the UAV and the ground sensor can be given by:

$$h_i(t) = P_{ls}(\varrho(i)(t))(\gamma_{los} - \gamma_{Nlos}) + 20log(\psi sec\varrho(i)(t)) + 20log(c_f) + 20log(4\phi/v_c) + \gamma_{Nlos} \quad (4)$$

where ψ, c_f, and v_c are the radius of the radio coverage of the UAV, the carrier frequency, and the speed of light, respectively. γ_{loS} and γ_{NloS} stand for the excessive path loss of LoS and non-LoS, respectively.

III. DDPG based Persistent Cruise Control Framework

DDPG repeatedly learns an action-value function and a policy to optimize the corresponding action. One loop of this learning is called an episode. DDPG combines the value iteration and the policy iteration to implement the proposition of the continuous state space and the continuous action space by using deep reinforcement learning.

In our work, we propose a novel DDPG-CC framework that provides a joint optimization for continuous cruise control.

The actions of the UAV in DDPG-CC can be stated as:

$$A_\alpha = (\theta(\alpha), s(\alpha), \{i_\alpha \in [1, N]\}), \quad (5)$$

where $A_\alpha \in \sum A$, and $\sum A$ comprises of every action the UAV carries out for optimization of the cruise control and communication schedule. In a environment with N ground sensors $\theta(\alpha)$ and $s(\alpha)$ are the respective headings and velocity of the action.

To change from state α to state β the network cost is denoted by $C\{\beta|\alpha, A_\alpha\}$. An experience tuple

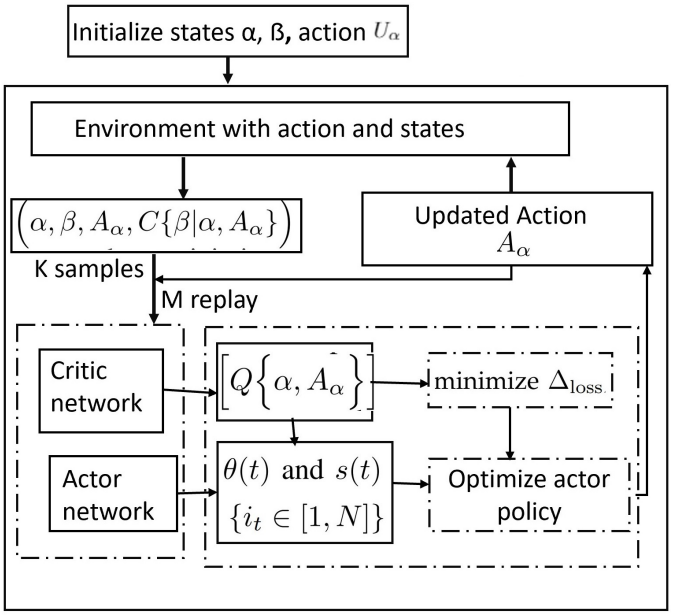

Fig. 3. DDPG-CC Algorithm

$\left(\alpha, \beta, A_\alpha, C\{\beta|\alpha, A_\alpha\}\right)$ is stored in the replay memory M replay of the UAV at every training step. K samples of this experience is utilized with the input states from the environment for the training of the DDPG-CC onboard. M episodes are carried out for a training time of t_{learning} and an action of A_α is updated at the UAV at every time step.

DDPG-CC uses the experience replay to train the policy gradients for minimizing the approximation loss between the actor-critic neural networks and the target neural networks. The experience of cruise control and sensor selection, i.e., $\left(\alpha, \beta, A_\alpha, C\{\beta|\alpha, A_\alpha\}\right)$, is stored in a memory tuple.

The critic neural network approximates the optimal action-value function $Q\left\{\alpha, A_\alpha\right\}$ that calculates the expected accumulated network cost, i.e., the overall data loss, after observing the state α and taking the action A_α. Instead of exhaustively evaluating the entire action space to minimize $Q\left\{\alpha, A_\alpha\right\}$, DDPG-CC approximates the optimal actions of the cruise control and communication schedule. At the critic network, K samples from experience replay memory are used in the training episode to minimize an approximation loss Δ_{loss}. The critic network learns the optimal $Q\left\{\alpha, A_\alpha\right\}$ for minimizing the approximation loss. The actor neural network in DDPG-CC generates the actions like the angle and speed $\theta(t)$ and $s(t)$, and also selects the ground sensor $\{i_t \in [1, N]\}$. The actor policy is updated at the UAV with the sampled policy gradients. With the optimized actor policy, the two target neural networks are updated constantly on-board at the UAV. The training episodes are carried out in the onboard DDPG-CC till an better trajectory is defined in accordance with the environment.

Each record is associated with a timestamp (TTA) in the experience replay memory. TTA is the number of slots that have elapsed since the latest node observation. When the TTA is large, there is a large data queue and there is an eventual buffer overflow. Using our proposed methodology we can efficiently reduce the TTA and improve the learning accuracy. In our method, the DDPG-CC learns the online energy arrivals, channel dynamics of the environment and patterns of data. Therefore, the DDPG-CC can optimize the actions of the UAV even at random battery energy levels and channel conditions.

IV. Performance Analysis

The performance analysis done in this work is three-fold. First, we demonstrate how the learning time helps in decreasing the overall packet loss for different number of nodes. Then, we compare the impact of the training episodes on the overall packet loss and the velocity of nodes in normal, uniform and ring shaped deployment. This helps the user to define the right deployment for the reduction of packet loss and also show the impact of training with respect to packet loss.

The proposed DDPG-CC is implemented in Python 3.5 on TensorFlow. In each experiment, DDPG-CC is trained onboard at the UAV for 300 episodes and learning time of 200 epochs. The onboard memory has 10,000 training records and every training episode uses batches of 100 samples. In Figure 4, we can see the impact of the training episodes on the packet loss rate for different number of nodes. With less training episodes the packet loss is higher. With the growth of training episodes, as the DDPG trains the actions, the packet loss significantly drops. One interesting fact we observe from this experiment is that despite the variance in the number of nodes put to test, the packet loss rate decreases steadily and eventually converges around 300 episodes.

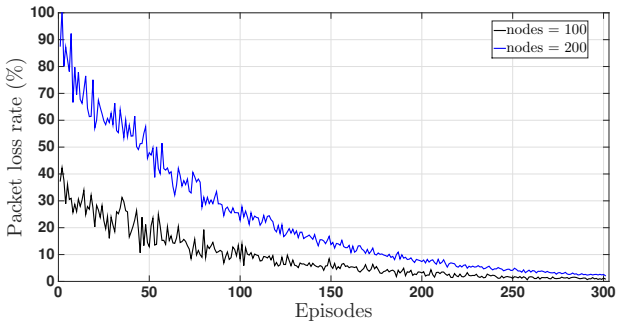

Fig. 4. Packet loss rate of DDPG-CC with regard to the learning episodes

Figure 5 presents the network costs of DDPG-CC according to normal, uniform and ring shaped distributions of the ground sensors. As the DDPG-CC optimizes the states and the actions dynamically, the overall network cost is gradually reduced. From Figure 5 it is observed that the network cost correspondent to the uniform deployment of the sensors is higher than nodes under ring-shaped and the normal deployment. Under uniform deployment the ground sensors within the radio coverage of the UAV can be easily selected for the data collection and the others deployments may suffer overflows. From this experiment we can infer that a dense deployment

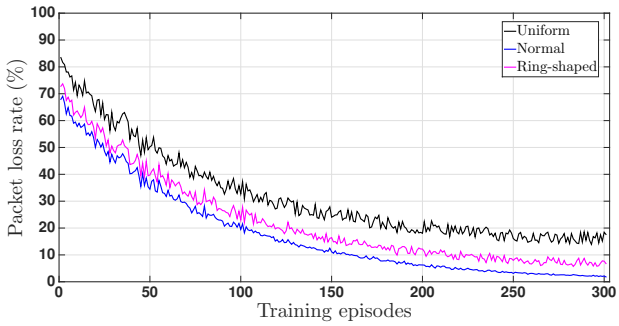

Fig. 5. Impact of various deployments on the Packet loss ratio

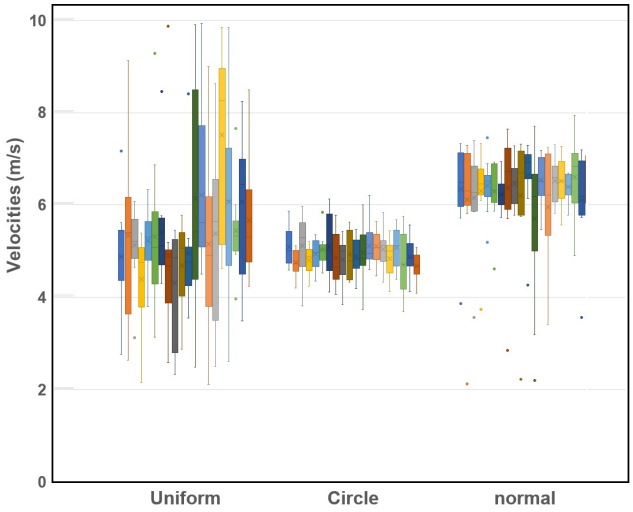

Fig. 6. Impact of various deployments on the velocity of the UAV

of the ground sensors helps in reducing the packet loss due to overflowing buffers.

Figure 6 shows that the velocity of the UAV is dynamically adjusted by DDPG-CC in accordance with the sensor deployment. Every box in this box plot represent 20 episodes and each deployment is tested for 200 episodes. When the velocity varies between 2 m/s and 10 m/s, the velocity of the UAV experiences the largest fluctuation under uniform distribution. This behavior is visible for the circle-shaped deployment and the normal deployment between 3.5 m/s and 6 m/s and between 4 m/s and 8 m/s. The variation under normal distribution is smaller than the uniform deployment, but higher than the one in the circular deployment. The superiority of the uniform distribution is due to the easier data collection than the other deployments. We can confer that the uniform shape of the sensor deployment results in a stable velocity control.

V. Conclusion

In this paper, we propose a Deep Deterministic Policy Gradient based Cruise Control (DDPG-CC) to minimize the overall data packet loss through online training of the headings and cruise velocities of the UAV. The proposed DDPG-CC optimally determines the instantaneous headings and patrol velocities as well as the selection of the ground sensor for the data collection. DDPG-CC is implemented in Python with Google TensorFlow. Based on our results we infer that DDPG-CC dynamically adapts the cruise control to reduce the packet loss under diverse deployments of the ground sensors. Additionally, by sufficiently training our proposed DDPG-CC the UAV can continuously adapt its cruise control and communication scheduling, which minimizes the packet loss rate. In this paper, the DDPG-CC scheme is provided for a single UAV networks however, this can be a potential candidate where multiple UAVs are deployed to collect the data of ground sensors and define an optimal trajectory.

Acknowledgements

This work was partially supported by National Funds through FCT/MCTES (Portuguese Foundation for Science and Technology), within the CISTER Research Unit (UIDP/UIDB/04234/2020); also by the Operational Competitiveness Programme and Internationalization (COMPETE 2020) under the PT2020 Partnership Agreement, through the European Regional Development Fund (ERDF), and by national funds through the FCT, within project ARNET (POCI-01-0145-FEDER-029074).

References

[1] Yi, M., Wang, X., Liu, J., Zhang, Y., Bai, B. (2020, July). Deep reinforcement learning for fresh data collection in UAV-assisted IoT networks. In IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops.

[2] Ejaz, W., Ahmed, A., Mushtaq, A., Ibnkahla, M. (2020). Energy-efficient task scheduling and physiological assessment in disaster management using UAV-assisted networks. Computer Communications.

[3] K. Li, Y. Emami, W. Ni, E. Tovar, and Z. Han, "Onboard Deep Deterministic Policy Gradients for Online Flight Resource Allocation of UAVs," IEEE Networking Letters, 2020

[4] Ebrahimi, D., Sharafeddine, S., Ho, P. H., Assi, C. (2020). Autonomous UAV trajectory for localizing ground objects: A reinforcement learning approach. IEEE Transactions on Mobile Computing.

[5] K. Li, W. Ni, E. Tovar, and A. Jamalipour, "On-board Deep Q-Network for UAV-assisted Online Power Transfer and Data Collection," IEEE Transactions on Vehicular Technology, 2019.

[6] Hu, Y., Chen, M., Saad, W., Poor, H. V., Cui, S. (2020). Meta-reinforcement learning for trajectory design in wireless UAV networks.

[7] K. Li, W. Ni and F. Dressler, "Continuous Maneuver Control and Data Capture Scheduling of Autonomous Drone in Wireless Sensor Networks," in IEEE Transactions on Mobile Computing, doi: 10.1109/TMC.2021.3049178.

[8] Maddikunta PK, Hakak S, Alazab M, Bhattacharya S, Gadekallu TR, Khan WZ, Pham QV. Unmanned aerial vehicles in smart agriculture: Applications, requirements, and challenges. IEEE Sensors Journal. 2021 Jan 6.

[9] K. Li, W. Ni, E. Tovar, and A. Jamalipour, "Online Velocity Control and Data Capture of Drones for the Internet-of-Things: An Onboard Deep Reinforcement Learning Approach," IEEE Vehicular Technology Magazine (VTM), 2021.

[10] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," IEEE Transactions on Wireless Communications, vol. 16, no. 6, pp. 3747–3760, 2017.

[11] K. Li, W. Ni, E. Tovar, and M. Guizani, "Joint Flight Cruise Control and Data Collection in UAV-aided Internet of Things: An Onboard Deep Reinforcement Learning Approach," IEEE Internet of Things Journal (IoTJ), 2020.

[12] Q. Wu, L. Liu, and R. Zhang, "Fundamental trade-offs in communication and trajectory design for UAV-enabled wireless network," IEEE Wireless Communications, vol. 26, no. 1, pp. 36–44, 2019.

[13] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," IEEE Wireless Communications Letters, vol. 3, no. 6, pp. 569–572, 2014.