On Evaluation of Discrete RL Agents for Traffic Scheduling and Trajectory Optimization of UAV-Based IoT Network with Multiple RIS Unit

Sheheryar[1], Sania Gul*[2], Rizwan Ahmad [1].
[1] Department of Electrical Engineering, National University of Sciences and Technology (NUST), Islamabad, Pakistan.
[2] Department of Electrical Engineering, University of Engineering and Technology (UET), Peshawar, Pakistan.
*** Correspondence:** Sania Gul saniagul@hotmail.com

Citation| Sheheryar, Gul. S, Ahmad. R, "On Evaluation of Discrete RL Agents for Traffic Scheduling and Trajectory Optimization of UAV-Based IoT Network with Multiple RIS Unit", IJIST, Special Issue. pp 275-285, May 2024
Received| May 03, 2024, **Revised**| May 21, 2024, **Accepted**| May 25, 2024, **Published**| May 29, 2024.

Unmanned Aerial Vehicles (UAVs) have been very effective for data collection from widely spread Internet of Things Devices (IoTDs). However, in case of obstacles, the Line of Sight (LoS) link between the UAV and IoTDs will be blocked. To address this issue, the Reconfigurable Intelligent Surface (RIS) has been used, especially in urban areas, to extend communication beyond the obstacles, thus enabling efficient data transfer in situations where the LoS link does not exist. In this work, the goal is to jointly optimize the trajectory and minimize the energy consumption of UAVs on one hand and satisfy the data throughput requirement of each IoTD on the other hand. As it is a mixed integer non-convex problem, Reinforcement Learning (RL); a class of Machine Learning (ML), is used to solve it, which has proven to be computationally faster than the conventional techniques to solve such problems. In this paper, three discrete RL agents i.e. Double Deep Q Network (DDQN), Proximal Policy Optimization (PPO), and PPO with Recurrent Neural Network (PPOwRNN) are tested with multiple RISs to enhance the data transfer and trajectory optimization in an Internet of Things (IoT) network. The results show that DDQN with multiple RIS is more efficient in saving communication-related energy, while a single RIS system with the PPO agent provides more reduction in the UAV's propulsion energy consumption when compared to other agents.

Keywords: Internet of Things Devices (IoTDs); Reconfigurable Intelligent Surfaces (RIS); Reinforcement Learning (RL) agents; Unmanned Aerial Vehicle (UAV); 6G communication.

Introduction

The concept of 'smart cities', 'smart agriculture', 'smart industries', 'smart transportation systems', and 'smart health care systems' rely heavily on information and communications technologies to gather the information critical for the efficient use of existing assets and resources, for increasing the land and the industrial productivity and for improving the safety and risk monitoring. The 'smart' concept requires smooth data collection from various sensors connected to the 'Internet of Things (IoT)' network, hence requiring integration of information and communication technologies. It is forecasted that by 2025, the number of IoT devices (IoTDs) may exceed 500 billion [1] thus requiring an enabling technology to handle this load. 6G is a potential technology expected to provide an enhanced user experience and better Quality of Service (QoS) for IoT networks due to its superior features over the previous network generations such as 1) ultra-low-latency, 2) extremely high throughput, 3) satellite-based customer services, and 4) massive autonomous networks [2]. Artificial Intelligence (AI), real-time intelligent edge processing, cognitive radio, the use of Unmanned Aerial Vehicles (UAVs), and Reconfigurable Intelligent Surfaces (RISs) enable the 6G networks to provide communication in difficult-to-reach areas (rural/ mountainous), as well as providing better QoS in densely populated urban areas [3]. RIS can smartly reconfigure the radio environment by incurring some change (amplitude/ phase or both) in the incident signal [4]. Compared to the active relays, RIS does not require power to amplify and transmit the signal. It just reflects the signal, making it a cost-effective solution [4]. Instead of installing multiple antenna towers to collect the data from these power-constrained IoTDs, spread over a large geographical area; UAVs have proved to be more economical. However, the UAV itself is an energy-constrained device and requires batteries or solar panels to enhance its flight time. This increases its weight, which in turn would result in more energy consumption.

Keeping in view the great potential of the 6G-IoT networks, many efforts have been put into research in this area. In a RIS-assisted UAV system for the IoT network proposed in [5], the UAV trajectory and the communication channel allocation to multiple IoTDs are jointly optimized by using the PPO agent. The RIS configuration is handled by invoking the Block Coordinate Decent (BCD) algorithm, where a finite set of phase angles is tested for each RIS element to maximize the amount of data collected from each IoTD. The objective was to provide a timely data collection service before the information became stale and was of no use. The proposed model has outperformed a similar model without an RIS, a model with randomly configured RIS, and the other two models implementing the random walk UAV and the stationary UAV configurations. In an IoT network of [6], the trajectory of UAV, RIS configuration, and traffic scheduling of IoTDs are jointly optimized by using the Double Deep Q Network (DDQN) and Deep Deterministic Policy Gradient (DDPG) Reinforcement Learning (RL) agents. It was found that the DDPG agent performs better than the DDQN agent, due to its continuous nature. The proposed systems have performed better than the system without RIS and the system with RIS but without optimal phase shifts configured. In the RIS-assisted UAV system of [7], multiple RISs are proposed to serve a single moving user. It was found that using multiple RIS units would result in increasing the communication Energy Efficiency (EE) and reducing the propulsion energy consumption. At any time, only one RIS is assumed to be active, while the others are considered to be in sleep mode to avoid the occurrence of destructive interference at the Ground Terminal (GT). The scheduling of RIS is based on its proximity to GT. In this model, Deep Q Network (DQN) and DDPG agents are used for designing the UAV trajectory and configuring the RIS phase shifts. Again, the DDPG agent outperforms the DQN agent and also produces better results than the system with fixed UAV and fixed RIS phases and another with randomly moving UAV and random RIS phases.

Our Contribution: In this paper, our main objective is to reduce the energy consumption of UAVs, by using various discrete RL agents and multiple RIS. Instead of using multiple UAVs

(as proposed by [8]), we prefer designing our proposed system with multiple RIS units, because the RIS is a passive device and its energy requirements are far lesser than the UAV. In our proposed IoT model, the UAV motion is allowed in 3D trajectory as in [6] and the system is tested with multiple agents, including PPO (as in [5]) and DDQN (as in [6]) and a newer agent PPO with Recurrent Neural Network (PPOwRNN). A comparison of these agents is made for IoT networks with single and multiple RIS. Our model uses Time Division Multiple Access (TDMA) for traffic scheduling.

The rest of the paper is organized as follows. In the next section, our proposed methodology and the experimental settings are given followed by the section on results and discussion, and finally, the paper is concluded.

Material and Methods:

Proposed Methodology:

Figure 1 shows the environment of our proposed model. As shown there, a single UAV is deployed to serve the K IoTDs, spread randomly on the ground, in the Area of Interest (AoI). The AoI is partitioned into equal-sized L cells. The location of the i^{th}cell in xy coordinates is given by $L_i^c = [x_i, y_i]^T \in \mathbb{R}^{2\times 1}$, where L_i^c is the center of cell i. The centers of the adjacent cells are separated by a distance of x_s and y_s in x and y coordinates respectively. The horizontal location of the of k^{th} IoTD is given as $\omega_k = [x_k, y_k]$, while they are on ground level, their height $z_k = 0$, and the average amount of data of the k^{th} IoTD is D_k, which needs to be uploaded to the UAV.

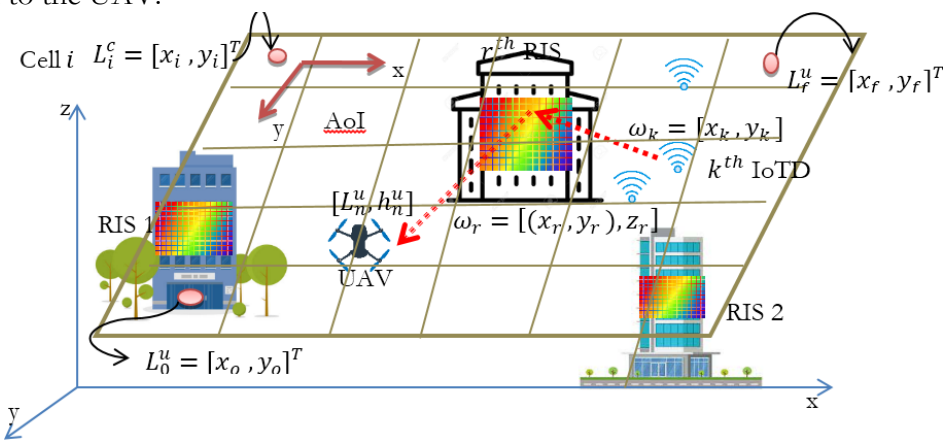


Figure 1: Network architecture with Area of Interest (AoI)

The horizontal location of the UAV at time n is given as $L_n^u \in \mathcal{L}$ where $\mathcal{L} = \{1,2,3, \ldots \ldots L\}$, $n = \{1,2,3, \ldots \ldots, N\}$, L is the total number of possible locations, and N is the maximum number of time instants, defined for an agent's training episode. At the end of each episode, the system restarts with the reset conditions, defined for any agent by the designer. Keeping the same nomenclature, L_0^u and L_f^u would be the initial and final locations of the UAV respectively. Then the horizontal trajectory of UAV in N instants is given as $\{L_0^u, L_1^u, L_2^u, L_3^u, \ldots \ldots L_n^u \ldots, L_N^u, L_f^u\}$. In the vertical direction, the UAV height level at any time instant n is given as h_n^u, where $h_n^u \in \mathcal{H} = \{1, 2, 3, \ldots \ldots, H\}$ and H is the total number of height levels. The height is also divided into discrete steps of size h_s, where $h_s = h_{max}/H$. Then the height of UAV at any instant n is given as $H_n^u = h_n^u h_s$, constrained by the condition in eq.(1)

$$h_{min} \leq H_n^u \leq h_{max} \quad (1)$$

With h_{min} and h_{max} being the maximum and the minimum heights respectively. We assume that t_n^u is the duration of time slot n, allocated to an IoTD and is given by eq. (2).

$$t_{min} \leq t_n^u \leq t_{max} \quad (2)$$

The total flight time of UAV is given as in eq. (3).

$$\tau = \sum_{n=1}^{N} t_n^u \quad (3)$$

The 3D trajectory of UAV is given by N waypoints, where a single waypoint is given as $[L_n^u, H_n^u], \forall n \in N$ along with the time slot duration $t_n^u, \forall n \in N$. With the constraint of maximum horizontal speed V_{max}^h on UAV, the number of time instants N must be large enough that the UAV location within the duration t_n^u must remain negligible as compared to the link distances between the UAV and the IoTDs. The horizontal speed of UAV at any time n is given by eq. (4).

$$v_n^h = \frac{|L_{n+1}^u - L_n^u|}{t_n^u} \leq V_{max}^h, \qquad \forall n \in N \quad (4)$$

If $v_n^h = 0$, the UAV will hover at time n. Likewise, for the maximum vertical speed V_{max}^v, UAV's vertical speed at any time n is given by eq. (5).

$$v_n^v = \frac{|H_{n+1}^u - H_n^u|}{t_n^u} \leq V_{max}^v, \qquad \forall n \in N \quad (5)$$

For rotary wing UAV, the propulsion energy at time n is given in eq. (6) as:

$$e_n^{uav} = t_n^u \left(P_0 \left(1 + \frac{3(v_n^h)^2}{u_{tip}^2}\right) + \frac{1}{2} d_0 \rho s \mathcal{b} (v_n^h)^3 + P_1 \left(\sqrt{1 + \frac{(v_n^h)^4}{4v_0^4}} - \frac{(v_n^h)^2}{2v_0^2}\right)^{1/2} + P_2 v_n^v\right) \quad (6)$$

Where P_0, P_1 and P_2 are constant blade profile power, induced power in hovering status and constant ascending/ descending powers respectively. u_{tip} is the tip speed of the rotor blade, d_0 is the main body drag ratio, s is the rotor's solidity, v_0 is the mean rotor's induced velocity while hovering, ρ the air density, and $\mathcal{b}$ denotes the rotor's disc area.

As shown in Figure 1, there are multiple RISs in the AoI. Each RIS has a large number of Passive Reflecting Units (PRUs). PRUs are arranged in $M_r \times M_c$ uniform planar array and the distance between the adjacent PRUs is d_r meters row-wise and d_c meters column-wise. The reflection coefficient $\mathrm{R}_{mr,mc}$ of each PRU is given by eq. (7) as:

$$\mathrm{R}_{mr,mc} = \alpha e^{j\theta_{mr,mc}}, \qquad \forall\, m_r \in 1, \ldots M_r, m_c = 1, \ldots M_c \quad (7)$$

Where α is the attenuation loss of PRU and $\theta_{mr,mc}$ is the phase shift of PRU, adjusted according to the direction of the IoTD being served. Suppose that there are total R RIS available in the AoI. Then, at any particular time n, only one of them is considered to be switched on, while the rest are switched off or in sleep mode. The turn-on schedule decision of the r^{th} RIS is given as $r_n^{RIS} = \{0, 1\}$, where $r_n^{RIS} = 1$ means r^{th} RIS is switched on and $r_n^{RIS} = 0$ indicates that it is switched off at time n. The RIS scheduling scheme is given in eq. (8).

$$r_n^{RIS} = \begin{cases} 1, & r^{RIS} = argmin(d_n^{uR}) \\ 0, & otherwise \end{cases} \quad (8)$$

Where $d_n^{uR} = \{d_n^{ur}, \forall r \in R\}$ denotes the set of distances between UAV and RIS at time n. Additionally, we have the condition of $\sum_{n=1}^{N} r_n^{RIS} = 1, \ \forall n \in N$, restricting only a single RIS to be turned on, which is nearest to the UAV. For the r^{th} RIS, the location of its first element is given by $\omega_r = [x_r, y_r]$ and its height above ground is given by z_r. Although, the horizontal locations of all RISs differ, their height above ground z_r is same.

Assume that the LoS does not exist between the UAV and any of the IoTDs, making it necessary for the IoTD to transmit its data to the UAV via RIS. Now the data from the IoTD is uploaded to the UAV via a cascaded channel, consisting of two LoS links (as shown by dotted

lines in Figure 1). The first LoS link is from IoTD to RIS and the other is from RIS to UAV. Then the channel gain between the UAV to r^{th} RIS (switched-on according to eq. (8)) at time n) can be denoted as $g_n^{ur} \in \mathbb{C}^{Mr \times Mc}$ and given in eq. (9) as

$$g_n^{ur} = \frac{\sqrt{\xi}}{d_n^{ur}} \left[1, e^{-j\frac{2\pi}{\lambda}d_r\phi_n^{ur}\psi_n^{ur}}, \ldots\ldots e^{-j\frac{2\pi}{\lambda}(M_R-1)d_r\phi_n^{ur}\psi_n^{ur}}\right]^T \otimes \otimes \left[1, e^{-j\frac{2\pi}{\lambda}d_c\varphi_n^{ur}\psi_n^{ur}}, \ldots\ldots e^{-j\frac{2\pi}{\lambda}(M_C-1)d_c\varphi_n^{ur}\psi_n^{ur}}\right]^T (9)$$

Where $[.]^T$ represents the transpose operation, ξ is the path loss at a reference distance $D_o = 1$ m, $d_n^{ur} = \sqrt{(H_n^u - z_r)^2 + (L_n^u - \omega_r)^2}$, λ is the carrier wavelength, $\phi_n^{ur} = \frac{x_i - x_r}{d_n^{ur}}$ is the cosine and $\varphi_n^{ur} = \frac{y_i - y_r}{d_n^{ur}}$ is the sine of the horizontal Angle of Arrival (AoA) and $\psi_n^{ur} = \frac{h_n^u - z_r}{d_n^{ur}}$ is the sine of the vertical AoA of signal at the r^{th} RIS. The symbol $\otimes$ shows the tensor product. Far-field array response vector model is assumed at RIS, where $d_n^{ur} \gg max\,(M_r\, d_r, M_c\, d_c)$. The channel gain between r^{th} RIS and k^{th} IoTD is denoted as $g_n^{rk} \in \mathbb{C}^{Mr \times Mc}$ and given as in eq. (10).

$$g_n^{rk} = \frac{\sqrt{\xi}}{d_n^{rk}} \left[1, e^{-j\frac{2\pi}{\lambda}d_r\phi_n^{rk}\psi_n^{rk}}, \ldots\ldots e^{-j\frac{2\pi}{\lambda}(M_R-1)d_r\phi_n^{rk}\psi_n^{rk}}\right]^T \otimes \left[1, e^{-j\frac{2\pi}{\lambda}d_c\varphi_n^{rk}\psi_n^{rk}}, \ldots\ldots e^{-j\frac{2\pi}{\lambda}(M_C-1)d_c\varphi_n^{rk}\psi_n^{rk}}\right]^T (10)$$

Where $d_n^{rk} = \sqrt{(z_r)^2 + (\omega_r - \omega_k)^2}$, $\phi_n^{rk} = \frac{x_r - x_k}{d_{rk}^n}$ is the cosine and $\varphi_n^{rk} = \frac{y_r - y_k}{d_n^{rk}}$ is the sine of the horizontal AoA and $\psi_n^{rk} = \frac{z_r}{d_n^{rk}}$ is the sine of the vertical AoA of signal at the k^{th} IoTD. The cascaded channel gain is then given as in eq. (11).

$$g_n^{urk} = g_n^{ur}.\boldsymbol{\theta_n}.g_n^{rk} (11)$$

Where $\boldsymbol{\theta_n} = diag\,(\theta_n) \in \mathbb{C}^{MrMc \times MrMc}$ is the r^{th} RIS reflection coefficient matrix, and $\theta_n = [e_n^{j\theta_{1,1}}, \ldots., e_n^{j\theta_{mr,mc}}, \ldots., e_n^{j\theta_{Mr,Mc}}]^T \in \mathbb{C}^{MrMc \times 1}$ According to our assumption of the Non-LoS (NLoS) path between UAV and IoTD, the channel gain at the k^{th} IoTD is given by eq. (12) as

$$g_n^k = (1 - p_n^k)\frac{\xi}{\left(d_n^{uk}\right)^2} + p_n^k g_n^{urk} (12)$$

Where $d_n^{uk} = \sqrt{(H_n^u)^2 + (L_n^u - \omega_k)^2}$ and p_n^k is the blocking probability between UAV and k^{th} IoTD at time n, given by eq. (13) as

$$p_n^k = 1 - \frac{1}{1 + \eta_1 e^{-\eta_2 tan^{-1}\left(\frac{H_n^u}{d_n^{uk}}\right) - \eta_1}} (13)$$

Where η_1 and η_2 are constants depending on the environment. The data rate R_n^k achieved at the k^{th} IoTD is given in eq. (14).

$$R_n^k = c_n^k\, B\, log_2\left(1 + \frac{P g_n^k}{B\sigma^2}\right) (14)$$

Where P is the transmission power of an IoTD, B is the channel bandwidth, σ is the noise variance and $c_n^k = \{0, 1\}$ is the scheduling decision for the k^{th} IoTD at time n. As TDMA is used, so only a single IoTD is scheduled for service at any time instant, i.e. $\sum_{n=1}^{N} c_n^k = 1,\ \forall n \in N$. The aim of this research is to minimize the propulsion energy and maximize the EE of UAVs. Let $L = \{L_n^u, n \in N\}, H = \{h_n^u, n \in N\}, \mathcal{C} = \{c_n^k, n \in N\}, T =$

$\{t_n^u, n \in N\}$ and $\theta = \{\theta_n, n \in N\}$ as already defined in the discussion above. The optimization problem can be formulated in eq. (15) as.

$$\mathcal{P}: min_{L,H,\mathcal{C},T,\theta} \sum_{n=1}^{N} e_n^{uav} \quad (15)$$

Due to being a mixed variable problem, eq. (15) is non-convex and would need extensive computational resources to be solved in real-time. As already pointed out, RL has been very effective in solving such optimization problems in real-time. The goal of eq. (15) is to minimize the fuel consumption of UAVs at all times with the constraints of: 1). $c_n^k = \{0, 1\}$, $\sum_{n=1}^{N} c_n^k = 1, \ \forall n \in N$, ensuring a single IoTD link to UAV at any time instant n, 2). $\sum_{n=1}^{N} t_n^u R_n^k \geq D_k, \forall k \in K,$ for making sure that the data uploading from k^{th} IoTD must be completed within the flight time of the UAV, 3).$v_n^h \leq V_{max}^h$, 4). $v_n^v \leq V_{max}^v$ and 5). $h_{min} \leq h_n^u \leq h_{max}, \ \forall n \in N$ for guaranteeing that the UAV's horizontal and vertical speeds and heights do not exceed the maximum limits, and finally 6). $t_{min} \leq t_n^u \leq t_{max}$ restricts the time slot duration allocation to an IoTD between t_{min} and t_{max} for data transmission.

For eq. (15), the observation and action spaces and the rewards for RL agents are defined below.

Observation Space:

The current state $s(n)$ at time n is defined as in eq. (16)

$$s(n) = \{s_u(n)\} \in \mathcal{S} \triangleq \mathcal{L} \times \mathcal{H} \quad (16)$$

Where $\mathcal{S}$ is the overall state space and $s(n) = (L_n^u, h_n^u) \in \mathcal{L} \times \mathcal{H}$ is the current horizontal and vertical location of the UAV.

Action Space:

The action space can be continuous or discrete, depending on the type of agent used. However, as we are using only the discrete agents, so, we will define only the discrete action space here. Ref [6] provides continuous space for a similar problem. The discrete action $a(n)$ at time n is defined as in eq. (17).

$$a(n) = \{l_n, h_n, c_n^k, t_n^u\} \in A \triangleq \mathcal{L}_u \times \mathcal{H}_u \times \mathcal{C} \times T \quad (17)$$

Where A is the overall action space and $a(n) = (l_n, h_n, c_n^k, t_n^u) \in \mathcal{L}_u \times \mathcal{H}_u \times \mathcal{C} \times T$ is the current action chosen from A. $\mathcal{L}_u \times \mathcal{H}_u$ is the action space from which the current UAV flying actions in horizontal and vertical directions l_n, h_n are chosen. $\mathcal{C}$ is the IoTD scheduling space $\mathcal{C} = \{c_n^k, \forall k, n\}$ and $T = \{t_{min} : 0.1ms : t_{max}\}$ is the action space of discrete flight time durations from which t_n^u is chosen between t_{min} and t_{max} with the step size of 0.1ms. Assume that during a one-time instant n, the UAV is only allowed to move to one of its adjacent cells from its current cell in the horizontal plane and also it is allowed only to move one step in the vertical direction. Then the horizontal and vertical locations of UAV in the next time instant $n + 1$ is given as in eq. (18) and (19):

$$L_{n+1}^u = L_n^u + l_n \quad (18)$$

$$H_{n+1}^u = H_n^u + h_n \quad (19)$$

Where l_n is the horizontal flying action of a UAV defined as $l_n \in \mathcal{L}_u \triangleq \{(0, y_s), ((0, -y_s), (x_s, 0), (-x_s, 0), (0, 0)\}$ with $\mathcal{L}_u$ being the horizontal action space of UAV comprising 5 actions including moving to north, south, east, west, or hovering at its current location respectively, and h_n is the vertical flying action of a UAV defined as $h_n \in \mathcal{H}_u \triangleq \{h_s, -h_s, 0\}$ with $\mathcal{H}_u$ being the vertical action space of UAV comprising 3 actions including moving upward, downward, or staying there respectively.

Step Reward:

The reward of taking an action $a(n)$, when the system is in state $s(n)$ at time n, is given by eq. (20):

$$rwd\big(s(n),a(n)\big) = \sum_{k=1}^{K}\sum_{\acute{n}=1}^{N}\frac{t_n^u R_n^k}{e_{\acute{n}}^{uav}} - p_0 (20)$$

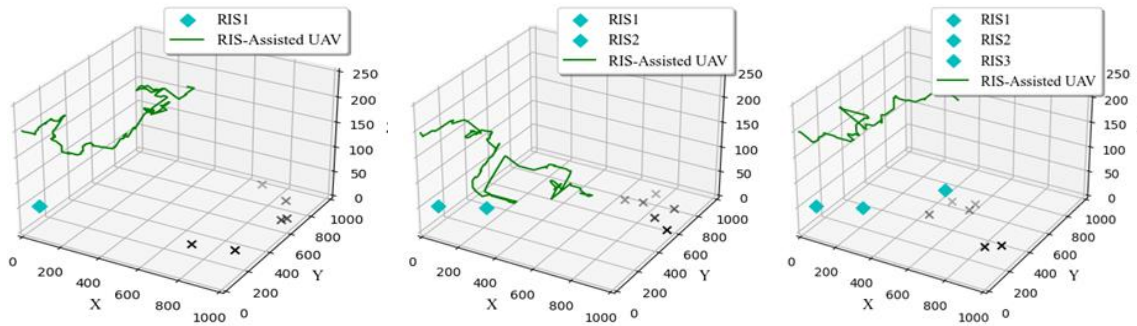
Where the reward function rwd is the ratio of the total data uploaded by the K IoTDs up to time instant $n+1$, to the propulsion energy of the UAV consumed till that time. However, penalty p_0 is applied to reward, if the data transferred to UAV at time n is less than the average data rate of any IoTD i.e. $t_n^u R_n^k < \frac{D_k}{N}$.

The reward would be highest if the data rate R_n^k is maximized. R_n^k can be maximized by maximizing the device gain g_n^k, which depends on the cascaded gain g_n^{urk}, which in turn is maximized, when the phase shift of each PRU of the active RIS (having $r_n^{RIS} = 1$) is adjusted to produce the maximum beamforming towards the IoTD scheduled during the current time n. The maximum reward for the state-actor pair $s(n), a(n)$ is given as in eq. (21).

$$rwd^{max}\big(s(n),a(n)\big) = \sum_{k=1}^{K}\sum_{\acute{n}=1}^{N}\frac{t_n^u R_n^{kmax}}{e_{\acute{n}}^{uav}} - p_0 (21)$$

The DDQN, PPO, and PPOwRNN agents are used for solving the complex optimization problem given in eq. (15). The first agent in the list belongs to the critic and the last 2 belong to the actor-critic categories of RL agents respectively. We will use the implementations of [9] for DDQN, and [10] for the PPO and PPOwRNN agents.

Experimental Settings:



(a) **(b)** **(c)**

Figure 2: Trajectory of UAV with a). Single RIS, b). Two RIS, c). Three RIS

The UAV's trajectory in the presence of multiple RIS is depicted in Figure 2 and the settings for the environment, observation and action spaces, and the hyper-parameters for the Neural Networks (NNs) of RL agents along their training settings are listed in tables 1 to 3 respectively.

Table 1: Environment settings

Environment	Symbol	Value
IoTDs and AoI	AoI	1000m x 1000m
	L	10000
	K	6
	$[\omega_k, z_k]$	[Random, 0]
	h_s	2 m
	x_s	10 m
	y_s	10 m
UAV	u_{tip}	120
	d_0	0.6
	s	0.05

	v_0	4.3
	ρ	1.225
	$\acute{o}$	0.503
	P_0	$\frac{12 \times 30^3 \times 0.4^3}{8} \rho s \acute{o}$
	P_1	$\frac{1.1 \times 20^{3/2}}{\sqrt{2\rho \acute{o}}}$
	P_2	11.46
	h_{min}	30m
	h_{max}	100m
	V_{max}^h	10 m/s
	V_{max}^v	10 m/s
	t_{min}	1s
	t_{max}	3s
RIS	R	3
	M	100
	z_r	50m
	ω_r	See the result section
	d_r, d_c	$\lambda/2$
Channel	η_1, η_2	9.61, 0.16
	B	2 MHz
	D_k	1024 bits
	f (carrier frequency)	900MHz
	P	500mW
	σ	169 dBm/Hz
Reset conditions	h_0^u	100m
	$L_0^u = [x_0^u, y_0^u]$	[0,0]
	θ_i	0^o
	h_0^u	100m

Table 2: Observation and Action Space

$\mathcal{S}$	3→$\{[L_n^u, h_n^u] = [x_n^u, y_n^u, h_n^u]\}$
$\mathcal{L}_u$	5 → {'north', 'east', 'west', 'south', 'hover'}
$\mathcal{H}_u$	3 → {'upward', 'downward', 'stay there'}
$\mathcal{C}$	6 → {(0,0,0,0,0,1), (0,0,0,0,1,0), (0,0,0,1,0,0), (0,0,1,0,0,0), (0,1,0,0,0,0), (1,0,0,0,0,0)}
$T = \frac{t_{max} - t_{min}}{0.1ms}$	21 → {1:0.1:3}
A	5x3x6x21 = 1890

Table 3: RL agents' NN and training parameters

Evaluation and target networks	**DDQN**	**PPO**	**POwRNN**
Total hidden layers in each NN	2	2	2
Total neurons in hidden layer 1	20	200	200
Total neurons in hidden layer 2	20	100	100
Number of LSTM cells inside each hidden neuron	NA	NA	2
Optimizer	RMS prop	Adam	Adam
Weights Initialization	$\mathcal{N}(0,0.003)$	$\mathcal{N}(0,0.003)$	$\mathcal{N}(0,0.003)$

Number of epochs O	NA*	25	25
Number of episodes E	60	120	120
Number of time instants N	600	600	600
Learning rate	0.005	0.001	0.001
Experience buffer F	3200	10000	10000
Mini batch size G	32	8	8
Discount factor γ	0.9	0.9	0.9
Exploration factor ε	0.9	NA*	NA*
Clip ratio ε,	NA*	0.2	0.2
Lambda (for advantage estimation function)	NA*	0.95	0.95
Value function loss coefficient	NA*	0.5	0.5

NA* =Not Applicable

PPOwRNN has feedback connections in their NNs, which are composed of Long Short Term Memory (LSTM) cells, which impact the current outputs of the hidden layer's neurons from their previous outputs and the previous outputs of the next layers' neurons. All agents start with a random policy π, and after every action $a(n)$, the agent stores the current state $s(n)$, action pair $[s(n), a(n)]$, action's reward $rwd(.)$, and the state of the system $s(n+1)$, after the action in its experience buffer F. During the training stage, mini-batches of G random samples from the experience buffer are selected to train the actor and critic's NNs. The discount factor γ determines how the rewards at the individual time steps are weighted. An action's influence over the future states of the environment typically decreases over time. A few parameters listed in Table 3 are specific to an agent and are not applicable to others. For example, the exploration factor ε is specific to DDQN and decides how much the agent would explore. Similarly, the number of epochs O, the clipping ratio ε, lambda, and the value function loss coefficients are specific to the PPO agents (both PPO and PPOwRNN) and are used for calculating its loss functions.

Result and Discussion:

The results for the DDQN, PPO, and PPOwRNN with single and multiple RISs are given in Table 4. The best results and the agents generating them are boldfaced. For multiple RIS, many random locations are tested, but among them, only those which generated the best results for any of the 3 agents, are reported in Table (4).

Table 4: Performance comparison of different agents with multiple RIS

# of RIS	Location $[\omega_r, z_r]$ m	Agent	Propulsion energy (kJ)	(EE) bits/J
1	[50 50 50]	DDQN	135.25	71.29
		PPO	**83.21**	88.67
		PPOwRNN	94.55	84.69
2	[50 50 50], [900 900 50]	DDQN	147.16	75.42
		PPO	128.05	48.58
		PPOwRNN	112.56	49.80
3	[50 50 50], [250 150 50], [500 500 50]	DDQN	109.49	**94.06**
		PPO	125.11	44.99
		PPOwRNN	161.94	53.05

As clear from Table 4, the PPO agent with a single RIS provides more propulsion energy savings and the DDQN agent with 3 RIS outperforms other agents in EE. Among all the RL agents used for our experiments, installing multiple RIS has benefitted the DDQN agent the most, while there is either little or no improvement in the performances of the PPO agent and

the PPOwRNN in the presence of multiple RISs. The DDQN agent with 3 RIS surpasses the PPO agent, which is the best available RL agent to date [11].

The computational complexity of the 3 agents is estimated in terms of the average training and evaluation time taken by them on Intel Core i5, 2.81 GHz CPU for multiple RIS units. On average, the DDQN agent takes 4 min and 30 s, the PPO 10h, 20 min, and 50 s, while the PPOwRNN requires 15h, 40 min, and 20s for convergence. Although the PPO and PPOwRNN agents save more propulsion energy in single and double RIS cases, they are far more complex than the DDQN agents.

Conclusion:

To collect data from a widely spread IoT network, one solution is to use multiple UAVs, to provide coverage to a large area, as in [8], but owing to the cost of fuel, it would be an expensive option. RIS on the other hand is a low-powered, easy-to-install, and lightweight device, which can be easily installed on buildings without posing any danger to the public as is caused by the huge antenna structures in case of bad weather conditions or earthquakes. So, we have tested different RL agents for the IoT data collection in the presence of multiple RISs and found that the presence of additional RIS units supports the otherwise weak agents (e.g. DDQN) more than the strong ones (e.g. PPO), which although does not require more RIS units, are computationally more expensive. It is shown in [6] that the continuous RL agents perform better than the discrete ones. So, in the future, these agents must also be tested with multiple RIS units. Also, more bandwidth-efficient access methods e.g. Orthogonal Frequency Division Multiple Access (OFDMA); must be implemented to estimate the maximum potential of the proposed algorithm.

Acknowledgement:

The authors acknowledge the support of the Artificial Intelligence in Healthcare (AIH) lab, UET Peshawar for providing the computational facilities.

Author's Contribution:

Sheheryar: Conceptualization, Experimentation, Analysis. **Sania Gul:** Experimentation, Analysis, Paper writing. **Rizwan Ahmed:** Conceptualization, Supervision, Paper review.

Conflict of Interest: The authors declare there exists no conflict of interest for publishing this manuscript in IJIST.

Project Details: This was the thesis work of Sheheryar for his MS degree.

References:

[1] M. Samir, S. Sharafeddine, C. M. Assi, T. M. Nguyen, and A. Ghrayeb, "UAV Trajectory Planning for Data Collection from Time-Constrained IoT Devices," IEEE Trans. Wirel. Commun., vol. 19, no. 1, pp. 34–46, Jan. 2020, doi: 10.1109/TWC.2019.2940447.

[2] "6G Internet of Things: A Comprehensive Survey,." Accessed: May 04, 2024. [Online]. Available: https://arxiv.org/pdf/2108.04973

[3] H. Tataria, M. Shafi, A. F. Molisch, M. Dohler, H. Sjoland, and F. Tufvesson, "6G Wireless Systems: Vision, Requirements, Challenges, Insights, and Opportunities," Proc. IEEE, vol. 109, no. 7, pp. 1166–1199, Jul. 2021, doi: 10.1109/JPROC.2021.3061701.

[4] Y. Zhao, J. Zhao, W. Zhai, S. Sun, D. Niyato, and K. Y. Lam, "A Survey of 6G Wireless Communications: Emerging Technologies," Adv. Intell. Syst. Comput., vol. 1363 AISC, pp. 150–170, Apr. 2020, doi: 10.1007/978-3-030-73100-7_12.

[5] A. Al-Hilo, M. Samir, M. Elhattab, C. Assi, and S. Sharafeddine, "RIS-Assisted UAV for Timely Data Collection in IoT Networks," IEEE Syst. J., vol. 17, no. 1, pp. 431–442, Mar. 2023, doi: 10.1109/JSYST.2022.3215279.

[6] H. Mei, K. Yang, Q. Liu, and K. Wang, "3D-Trajectory and Phase-Shift Design for RIS-Assisted UAV Systems Using Deep Reinforcement Learning," IEEE Trans. Veh. Technol., vol. 71, no. 3, pp. 3020–3029, Mar. 2022, doi: 10.1109/TVT.2022.3143839.

[7] L. Wang, K. Wang, C. Pan, and N. Aslam, "Joint Trajectory and Passive Beamforming Design for Intelligent Reflecting Surface-Aided UAV Communications: A Deep Reinforcement Learning Approach," IEEE Trans. Mob. Comput., vol. 22, no. 11, pp. 6543–6553, Nov. 2023, doi: 10.1109/TMC.2022.3200998.

[8] M. Samir, D. Ebrahimi, C. Assi, S. Sharafeddine, and A. Ghrayeb, "Leveraging UAVs for Coverage in Cell-Free Vehicular Networks: A Deep Reinforcement Learning Approach," IEEE Trans. Mob. Comput., vol. 20, no. 9, pp. 2835–2847, Sep. 2021, doi: 10.1109/TMC.2020.2991326.

[9] "Simulation code for DDQN", [Online]. Available: https://github.com/HaiboMei/UAVRIS-DRL.git

[10] "Welcome to Spinning Up in Deep RL! — Spinning Up documentation." Accessed: May 04, 2024. [Online]. Available: https://spinningup.openai.com/en/latest/

[11] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," Jul. 2017, Accessed: May 04, 2024. [Online]. Available: https://arxiv.org/abs/1707.06347v2