Decentralised Reinforcement Learning in Fog-based Internet-of-Things

XXX YYYY, and XXX ZZZZ

***Abstract*—Reinforcement learning (RL) algorithms offers more insights about the overall futuristic functionalities for intelligent Internet-of-Things (IoT) devices. With the explosive growth in the number of IoT devices, as well as the highly-distributed deployments of these devices today, managing the IoT devices centrally becomes infeasible. As such, several disruptive paradigms have emerged, one of which is the fog computing-based IoT, which aim towards shifting computation, control, and decision-making closer to the network edge. However, mobility and power-constrain of these fog devices remains an issue of concern. In this paper, we apply a q-learning algorithm to minimize the outage in communication within a fog-based IoT network, by optimizing the power-control parameter of the agent, as well as optimizing the physical position. Furthermore, the agent was able to efficiently minimize the energy consumed by the fog relay and the IoT sensor, while guaranteeing efficient transmission.**

***Index Terms*—Reinforcement learning (RL), Fog-based Internet-of-Things (IoT), q-learning, communication outage, energy management.**

I. Background

THE fog computing-based IoT paradigm aims at moving computation, control, and decision-making within the IoT ecosystem closer to the network edge [1]. The key driver of this paradigm are fog devices, which may be energy-constrained or not, and can either be mobile or static. The deployment and efficient utilization of these fog devices will contribute to the success of future IoT systems [2], one of which is serving as relays to overcome communication outages due to obstacles or long distances between a source node and a remote destination node where IoT services may be rendered.

However, in order for devices to communicate efficiently with minimal outage (loss of transmitted packets), several bottlenecks may arise, one of which is the efficient utilization of energy by power-constrained IoT devices. Energy can be used up when these devices unnecessarily increase their power-level in order to communicate with neighbouring devices within the network, conversely, energy can be saved when the devices regulate their transmission power, especially in situations when they are relatively close to the communicating parties. For example, an IoT end-device that transmits at high power irrespective of the channel conditions or proximity of its neighbours may drastically deplete its energy and die-out, hence, resulting in communication breakdown due to a point-of-failure within the network. Another challenge to be addressed is the energy consumed due to mobility. In order to deliver on some acceptable quality-of-service (QoS) within the IoT ecosystem, it is imperative for devices to be mobile, however, these devices are at risk of draining most of their energy on movement. For instance, energy will be inefficiently used if a power-constrained fog device decides to move closer to the communicating party only to relay very few numbers of packets, though having a guarantee that all the packets are delivered to the destination. However, if the size of the number of packets to be relayed is large, it may be optimal for the fog device to move closer, hence, conserving the energy of the IoT end-device, who can now transmit at a lower power level.

Moreover, since power-control mechanisms and smart mobility are critical in minimizing outages in communication, these devices should be able to learn when to increase, decrease, or maintain their power-levels or to move efficiently in order to increase the long-term performance of the network. More so, near-optimal actions from these devices are required to drive several smart cities applications, most importantly the Industrial IoT (IIoT), where industrial robots are deployed to act intelligibly in a dynamic industrial setting, and intelligent monitoring applications, where surveillance drones are deployed in militarised zones to meet stringent quality-of-service (QoS) requirements [3].

The work in [3] used an iterative algorithm based on the steepest descent method to address the problem in a multi-tier fog-based IoT architecture with a fog device which could adjust its position and power-control parameter in order to minimize outage in communication. However, optimization was carried out using the gradient descent method with slow convergence and unspecified states and actions set. In [5], a clustering algorithm was used on eight (8) unmanned aerial vehicle (UAVs) to collect data from ground IoT devices with an objective of minimizing the energy consumed by the IoT devices during uplink communications. However, this work considered a centralized network where the locations of every IoT device and UAV were known to the controller. Considering the highly-distributed nature of deployed IoT devices, it becomes infeasible to manage devices centrally [4]. As such, reinforcement learning can be effectively deployed on fog devices to allow them to act independently based on their local experiences in the environment, i.e. each fog device should be able to learn independently without a central entity.

A decentralised stateless Q-learning approach was proposed in [4] to improve aggregate throughput in four coexistent wireless networks (WN). Each WN was considered to be an agent running the stateless Q-learning algorithm with agents having action space as channel number, and transmit power (dBm). A lightweight distributed learning approach was proposed in [7] to increase energy efficiency and reliability of IoT communications. There was significant performance

Manuscript received March 14, 2019; revised July X, 2019.

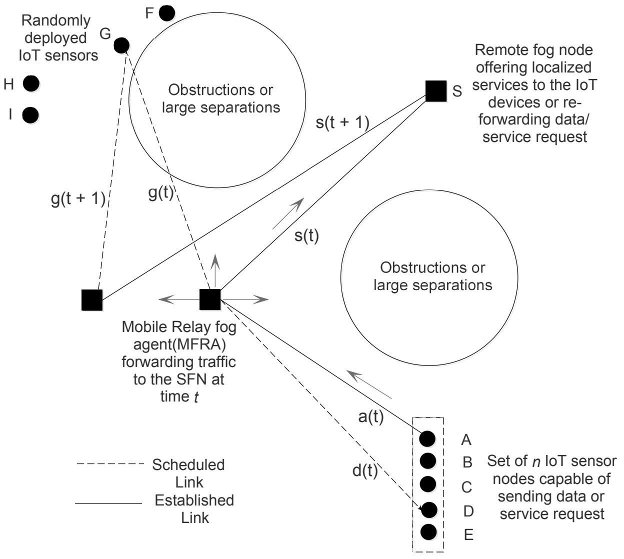
Randomly deployed IoT sensors
F
G
H
I
Obstructions or large separations
Remote fog node offering localized services to the IoT devices or re-forwarding data/ service request
S
s(t + 1)
g(t + 1)
g(t)
s(t)
Obstructions or large separations
Mobile Relay fog agent(MFRA) forwarding traffic to the SFN at time *t*
a(t)
d(t)
A
B
C
D
E
Set of *n* IoT sensor nodes capable of sending data or service request
Scheduled Link
Established Link

Fig. 1. Single-agent fog-based IoT system.

improvement when the proposed algorithm was compared to a centralized optimized strategy. Transmit power, sub-channel, and spreading factor made up the action space. However, the system model in both works was rather hierarchical than distributed, ie. each WN was assumed to be an independent central entity with no specifications to what is learnt within each sub-network. Though IoT is defined as a large-scale network where various sub-networks coexist [1], applying RL to end-devices within sub-network may bring about meaningful performance improvement in the overall IoT network.

The main contribution of this paper is to propose a decentralised reinforcement learning approach as in [6] that addresses communication performance within a fog-based IoT architecture. First, we assumed a single-agent scenario of possible state-action pair for each communication scenario as seen in Fig. 1, where an agent may be faced with a unique topology and environment. Next, in order to guarantee energy-efficient communications for the fog devices in the dynamic environment, we apply a decentralised q-learning algorithm, where each agent observes its position with respect to the communicating party, as well as its present transmit power level and learns to take actions that minimize loss of packets, as well as efficient energy utilization.

The remainder of this work is organized as follows. In Section II, we reviewed related works, and present our proposed approach in Section III. In Section IV, we evaluate the proposed fog-based IoT system, and present the results in Section V. Section V concludes the paper and outlines future directions.

II. Problem Definition

In this section, we provide full description of the system model, as well the RL approach used to address the problem. The Mobile Fog Relay Agent (MFRA) and its environment are discussed below.

A. MFRA environment

States: The states are defined as a tuple, ⟨Outage communication cost ($\mathcal{P}_{out}$) /Energy status of the fog relay (J) /Energy status of the IoT sensor (J)⟩.

- Outage communication cost: Outage observations from the environment is estimated using (1) from [3], which gives an estimate of the communication outage when the agent takes an action, such as changing power levels or location, or both.
- Energy expended by fog relay: This observation gives the agent insight on how much energy by the fog agent when following policy $\omega_i \in \omega_{fog}$. If the fog agent continues to take sub-optimal actions, it depletes its energy and dies out.
- Energy expended by IoT sensor: This observation gives the agent insight on how much energy the IoT sensor has used up by following a policy $\omega_i \in \omega_{IoT}$. If the IoT sensor continues to take sub-optimal actions, it depletes its energy and dies out.

$$\mathcal{P}_{out} = 1 - (1 + 2\Psi^2 \ln \Psi) \exp\Big(-\frac{N_0\tilde{\kappa}}{P_I(D_I+\delta)^{-\sigma}}\Big), \quad (1)$$

where $\Psi = \sqrt{(N_0\tilde{\kappa})/(P_R(D_S+\delta)^{-\sigma})}$, and $\mathcal{P}_{out}$ is an expression for the outage probability with values between 0 and 1. We assume a predefined threshold $\tilde{\kappa}$ which determines the outage in communication, P_I is transmit power of the IoT sensor, P_R is transmit power of the fog relay agent, D_I is the distance between IoT sensor and fog relay agent, and D_S is the distance between fog relay agent and destination node. We assume a small change in the position of the fog relay agent, $\delta = \pm 0.25m$, N_0 to be the channel noise, and σ to be the path-loss exponent.

B. MRFA agent

We apply a the Q-Learning algorithm, an RL approach which requires no prior knowledge of the environment by the agent. In Q-learning, the agent interacts with the environment over periods of time according to a policy ω. At every time-step $k \in N$, the environment produces an observation $s_k \in \mathbb{R}^{D_s}$. By sampling, the agent then picks an action a_k over $\omega(s_k)$, $a_k \in \mathbb{R}^{D_a}$, which is applied to the environment. The environment consequently produces a reward $r(s_k, a_k)$ and may end the episode at state s_N or transits to a new state s_{k+1}. The agent's goal is to minimize the expected cumulative cost, $\min_\omega \mathbb{E}_{s_0,a_0,s_1,a_1,\ldots,s_N}\Big[\sum_{i=0}^{N}\gamma^i\mathcal{C}(s_i)\Big]$, where $0 \leq \gamma \leq 1$ is the discount factor, and $\mathcal{C}$ is the overall cost function of our model.

First, the agent takes an initial random action a_k and gets observations from the environment which corresponds to that action, as well as a reward. It then discretizes the continuous observations emanating from the environment into a $50 \times 5 \times 5$ state space corresponding to the tuple, ⟨Outage communication cost ($\mathcal{P}_{out}$) /Energy status of the fog relay (J) /Energy status of the IoT sensor (J)⟩. The agent then updates it's Q-values at each time-step k following (2).

$$Q(s_k, a_k) := Q(s_k, a_k) + \alpha\Big[r_{k+1} + \gamma \max_a Q(s_{k+1}, a) - Q(s_k, a_k)\Big], \tag{2}$$

where α is the learning rate, which determines the impact of new experience on the Q-value, r_{k+1} is the reward the agent receives by being in s_{k+1} from s_k. Based on the policy followed by the agent, it gets observations and rewards from the environment.

Action space: The actions are move and transmit by fog relay, and select a power-level and transmit by IoT end-device, which make up eight possible actions.

- Mobility: Move by $\pm\delta$ and transmit, where $\delta = \pm 0.25m$ and mobility range (m) = [-30, 30]
- Power-level: Choose power-level and transmit, P, where transmit power ranges (W) = [0.001, 0.01, 0.15, 0.2, 0.25, 0.3]

Goal: The goal is for the agent to learn to minimize the overall cost $\mathcal{C}$ in the tuple, $\langle$Outage communication cost ($\mathcal{P}_{out}$) /Energy status of the fog relay (J) /Energy status of the IoT sensor (J)$\rangle$, by keeping all nodes within the link alive while ensuring that the packets received in each transmission does not fall below the pre-defined threshold, which was set at 95%.

Rewards: The reward function used is given in (3) as

$$R = \begin{cases} 100, & \text{if } goal == Reached \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Metrics: Outage probability, i.e. the ratio of the number of packet lost to those transmitted, which we measure in percentages, and energy status of the fog relay agent and the IoT sensor, i.e. the ratio of depleted energy to the initial capacity in Joules, which we measure as a percentage.

The MFRA's learning process is summarized in Algorithm 1. A new learning episode is terminated when the agent attains the pre-defined goal of minimizing the communication outage in the link, or when either the fog relay or the IoT sensor dies out due to taking sub-optimal actions without getting to the goal. When an fog relay moves closer to the communicating parties, the IoT sensor uses a lower power level as compared to when it is far away, hereby saving IoT sensor energy. However, mobility have some cost and if the fog relay continues to move in order to minimize the communication outage, it may die out soon, hereby causing a point-of-failure to the network. As each episode is completed, a reward of 100 points is given to the agent if it reaches its goal and a 0 points otherwise. The reward is updated in the Q-learning table, with environmental information updated as well.

III. Experimental Setup

We carried out experimentation using the Python IDE 3.7.2, and considered a single agent to help us compare our proposed approach with the baseline.

Algorithm 1 MFRA Learning Process

```
Initialize: Power levels (W) = [0.001, 0.01, 0.15, 0.2, 0.25, 0.3], $\delta = \pm 0.25m$ and mobility range (m) = [-30, 30]
top:
ResetEnvironment()
state ← MapLocalObservationToState(env)
action ← QLearning.SelectAction(state)
if action == "move close and Tx" then
    Env.EstimateOutage (1)
    Env.EstimateFogEnergyStatus
else if action == "move away and Tx" then
    Env.EstimateOutage (1)
    Env.EstimateSensorEnergyStatus
else if action == "choose power level and Tx" then
    Env.EstimateOutage (1)
    Env.EstimateSensorEnergyStatus
endif
InvokePolicy(ExponentialDecay)
UpdateQLearningProcedure() (2)
CurrentState ← NewState
if goal == "Reached" then return Reward = 100,
else if goal != "Reached" or Agent == Death then return Reward = 0
endif
EndEpisode goto top.
```

A. Baselines

The baseline [3] used for the comparative analysis applied a gradient descent approach to arrive at a local optima in minimizing the communication outage within the network. The baseline approach is known to arrive at a local optima, however, may take significant amount of time to converge.

B. Indicators

To evaluate our proposed approach, we examine the convergence of our approach in minimizing the steps it takes the agent to get to its goal, the energy utilization of the fog relay and IoT sensor.

IV. Results and Discussions

In this section, we present the results of the fog-based IoT system. First, we compare the proposed approach with the baseline, then we examine the number of steps required by the agent to reach it's goal over learning episodes. Finally, we evaluate the energy utilization of both the fog relay agent and the IoT sensor.

A. Proposed approach vs. baselines

We carried out comparison of the proposed approach, applying the Q-learning algorithm and the baseline, using the gradient descent algorithm. Simulations were done using Python IDE 3.7.2, and Table I shows a summary of the parameters used. Fig. 2 shows the percentage of packets successfully transmitted, with each algorithm learning to minimize the

TABLE I
SIMULATION PARAMETERS

Parameter	*Values*
D_I	40 metres
P_I	[0.001, 0.3] Watts
D_S	35 metres
P_R	0.3 Watts
δ	±0.25 metres
Mobility bound	[-35, 35] metres
Noise power N_0	2×10^{-7} Watts
Path-loss exponent σ	3
Pre-defined threshold κ	1
Discount factor γ	0.9
Learning rate α	0.1
Episodes N	1000
Iteration runs	100000
Policy ϵ	$e^{-0.0015N}$

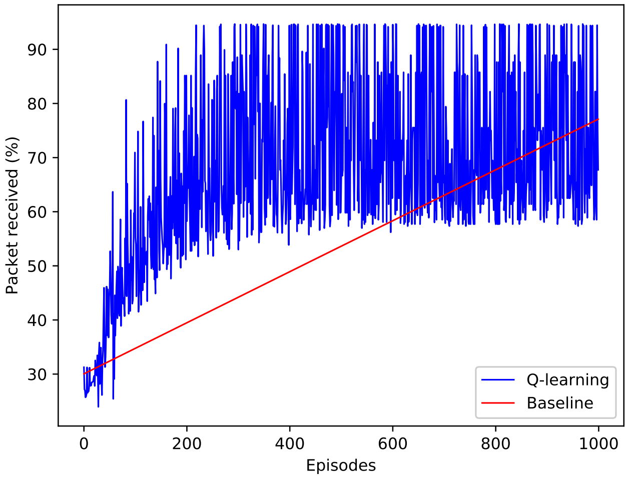

Fig. 2. Percentage of packets successfully transmitted.

number of packets lost over each episode. We observe that in the first 20 episode, the baseline had better performance than the proposed approach, however, for the remaining part of the episodes, the proposed approach out-performed the baseline. This signifies that the agent's exploration within the environment plays some significant role in improving the packets successfully transmitted. Convergence can be observed around 200th episodes where about 55% - 100% of transmitted packets are successfully received at the destination. Interestingly, we know that gradient descent methods may be slow to converge, but always surely converge to a local optima [3]. This is well depicted Fig. 2, where around 1000th episode the baseline approach is able to successfully transmit about 75% of packets.

Overall, the performance of the proposed approach yields better results than the baseline. In Fig. 3, we observe the number of iterations required for the agent to reach it's goal following policy ϵ. After 200 episodes, we achieve convergence. This implies that the agent learns to reach the goal

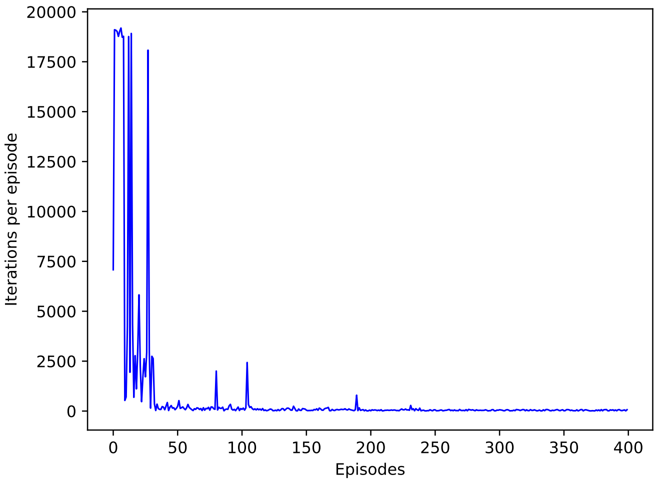

Fig. 3. Number of iteration over episodes.

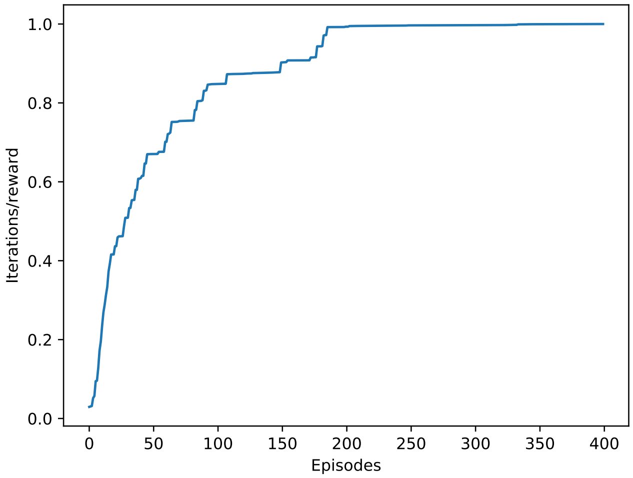

Fig. 4. Normalized Iterations per reward over 1000 episodes.

state faster and more efficiently. Similarly, in Fig. 4, which shows the number of iterations per reward, convergence was attained at about 200 episodes. This implies that the agent learns to follow policies that maximize its total expected reward. Therefore, the proposed RL approach has shown good performance behaviour as compared to the previous approach.

B. Energy utilization

We examine the impact of the learning process on the energy utilization of both the fog relay and the IoT sensor. Fig. 5, shows the energy consumed by the fog relay during transmission. We observe that at about 50 episodes, the fog relay consumes less than 10% of its energy in moving and transmitting. This has significant impact to increasing the longevity of fog devices within the network. A similar behaviour is also observed for the IoT sensor shown in Fig. 6, where the IoT sensor after about 60 episodes, spends less

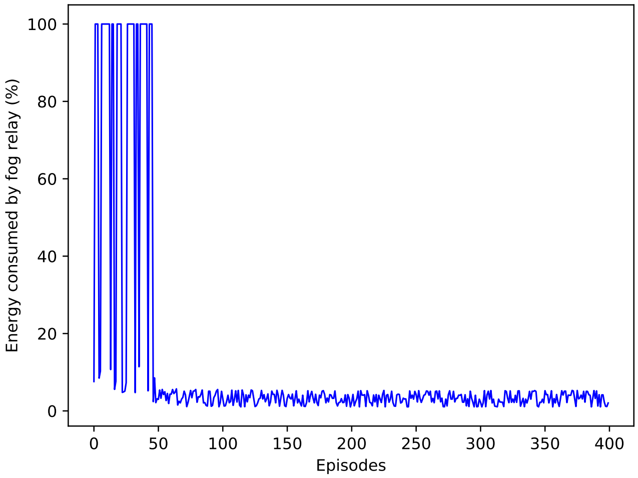

Fig. 5. Energy consumed by fog agent over episodes.

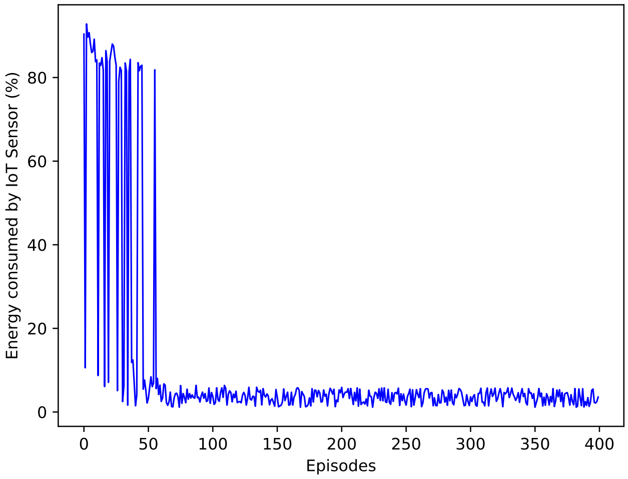

Fig. 6. Energy consumed by IoT sensor over episodes.

than 10% of its energy on transmission. The outcome of this experiments reveal that energy management within the IoT domain can be efficiently tackled using RL approach.

V. Related Works

Considering the dynamics and heterogeneity within the ultra-distributed IoT environment, it is important for communication devices which are mobile to move seamlessly without degrading quality of service (QoS). More so, the constrained devices should communicate efficiently without depleting all their energy by transmitting at high-power, which may have long-term consequences to the network. Several non-RL-based approaches have been proposed to optimize the communication performance in IoT based networks, however, when some parameters in the network are changed, these approaches may fail. The work in [3] considered a multi-tier fog-based IoT architecture where a mobile/static fog node acts as an amplify and forward relay that transmits received information from a IoT sensor node to a higher hierarchically-placed fog device, which offers some localized services. In order to minimize the outage in communication, an iterative algorithm based on the steepest descent method (SDM) was proposed to jointly optimize the mobility pattern and power-control parameters. However, the single-agent based work did not take into consideration the performance of the approach in a highly-decentralised IoT environment. Furthermore, each time the topology is changed, the agent will need to recompute the model in other to act optimally.

RL can be applied to a new environment, since the it allows the agent to learn, by take actions that will improve its long-term return. Furthermore, RL can be applied to multiple agents. In [4], a decentralised, multi-agent-based stateless Q-learning approach was proposed, where no information about neighbouring nodes are available to agents. Though the paper was able to improve aggregate throughput in the network, by allowing the networks modify both the transmission power and the channel used, however, high variability was observed in the throughput of the individual networks. Only eight agents were observed using a stateless Q-learning approach. The work did not adequately depict the distributed nature of the IoT network. Moreover, mobility of agents was completely ignored.

Several works in WSN have applied the decentralised RL technique. In [10], a multi-agent reinforcement learning-based multi-hop mesh cooperative (MRL-CC) mechanism for the improvement of some QoS metrics in the WSNs. Though the mobility of the cooperative nodes were taken into account when learning the optimal policy, the MRL-CC failed to consider the power dissipated by the power-constrained devices, as well as the overall outage in communication.

In [9], a reliable and energy-efficient routing (REER) protocol was proposed using a geographic routing approach. The work considered the idea of a central entity, called a reference node (RN), which is assumed to be situated at an ideal location between source and destination. Several other cooperative nodes, which contend to relay data, are assumed to be situated around the RN. The work was able to examine the trade-off between reliability and energy-efficiency when the distances between RNs was adjusted.

We present a decentralised reinforcement learning approach that adequately addresses some key performance issues within the IoT domain. The main task of our work is to minimize global outage in communication within a fog-based IoT network, by optimizing the power-control parameter of the potential mobile fog-relay agent (MFRA), as well as optimizing the position of each relaying agents in the network. As such, each MFRA is compelled to take certain actions that may influence its environment. However, the duration it takes the MFRA to learn is significantly influenced by the state space, as well as the possible set of actions [8]. The variables for the state, action and reward of an agent may be discrete or continuous, with the former represented as small interval of values which imply distinct levels [11], and can easily be represented in a tabular form. However, it is difficult to represent continuous space using Q-learning tables. The work in [12] considered a RL agent that explores continuous state and action space using Gaussian unit search behaviour. Other works [8], [14] considered the reduction of states by eliminating states that are unlikely to occur. However, this may pose a big risk especially in a highly dynamic environment. RL can be effective for learning action policies in discrete stochastic environments, but its efficiency can decay exponentially with increasing state space [13]. Our proposed problem can be observed to have continuous state-action pairs, and is approached by discretizing the state and action space.

In this paper, we assume the following.

1) The MFRA is completely oblivious of its environment, and as such, has no prior knowledge of the overall cost function.
2) The MRFA may change its position in order to ensure better communication. Also, the MFRA may change its position (2D/3D) depending on the scenario considered.
3) The MFRA has an objective of learning to make actions that yield better outcomes within its local view of the environment.
4) The states are be divided into discrete levels to overcome the exponential decay in the efficiency of the proposed approach due infinite state space.
5) The MFRA independently tries to optimize power usage and moves in a direction that maximizes the communication outage.

VI. Conclusion and Future works

We aim to apply the q-learning algorithm on a multi-agent fog-based IoT system where multiple agents compete to transmit reliably in a highly dynamic environment, where interference contributes significantly to communication outages with the network. For instance, agents may take actions that can have direct consequences on neighbouring agents, which may have further impact on other agents within the network. For instance, if an agent decides to increase the transmit power beyond some threshold value, in order to boost its communication capabilities, its action may result in channel interference to its immediate neighbours, and worst, it may deplete its energy fast, and die-out, leading to link failure that can affect the performance of the entire network. This will be looked at in our future work.

References

[1] B. Omoniwa, R. Hussain, M. A. Javed, S. H. Bouk and S. A. Malik, "Fog/Edge Computing-based IoT (FECIoT): Architecture, Applications, and Research Issues," in IEEE Internet of Things Journal.

[2] M. Chiang and T. Zhang, “Fog and IoT: An Overview of Research Opportuinities,” IEEE Internet of Things, vol. 3, no. 6, pp. 854-864, Dec. 2016.

[3] B. Omoniwa et al., "An Optimal Relay Scheme for Outage Minimization in Fog-based Internet-of-Things (IoT) Networks," in IEEE Internet of Things Journal.

[4] F. Wilhelmi, B. Bellalta, C. Cano and A. Jonsson, "Implications of decentralized Q-learning resource allocation in wireless networks," 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, 2017, pp. 1-5.

[5] M. Mozaffari, W. Saad, M. Bennis and M. Debbah, "Mobile Internet of Things: Can UAVs Provide an Energy-Efficient Mobile Architecture?," 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, 2016, pp. 1-6.

[6] M. Gueriau and I. Dusparic, "SAMoD: Shared Autonomous Mobility-on-Demand using Decentralized Reinforcement Learning," 1558-1563. 10.1109/ITSC.2018.8569608.

[7] A. Azari and C. Cavdar, "Self-organized low-power IoT networks: A distributed learning approach," arxiv

[8] I. Dusparic and V. Cahill, “Distributed W-Learning: Multi-Policy Optimization in Self-Organizing Systems,” 2009 Third IEEE International Conference on Self-Adaptive and Self-Organizing Systems, San Francisco, CA, 2009, pp. 20-29.

[9] M. Chen, T. Kwon, S. Mao, Y. Yuan and V. C. M. Leung, “Reliable and energy-efficient routing protocol in dense wireless sensor networks,” Int. J. Sen. Netw. vol. 4, no. 1/2, pp. 104-117, July 2008.

[10] Xuedong Liang, Min Chen, Yang Xiao, I. Balasingham and V. C. M. Leung, “A novel cooperative communication protocol for QoS provisioning in wireless sensor networks,” 2009 5th International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities and Workshops, Washington, DC, 2009, pp. 1-6.

[11] K.-L. A. Yau, P. Komisarczuk, P. D. Teal, “Reinforcement learning for context awareness and intelligence in wireless networks: Review, new features and open issues,” Journal of Network and Computer Applications, vol. 35, no. 1, pp. 253-267, Jan. 2012.

[12] N. Vucevic, J. Perez-Romero, O. Sallent and R. Agusti, "Reinforcement Learning for Active Queue Management in Mobile All-IP Networks," 2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications, Athens, 2007, pp. 1-5.

[13] W. T. B. Uther and M. M. Veloso, “Tree Based Discretization for Continuous State Space Reinforcement Learning,” *Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence,* 1998, Madison, Wisconsin, USA, pp. 769-774.

[14] H. Cuayahuitl, S. Renals, O. Lemon, and H. Shimodaira. “Learning multi-goal dialogue strategies using reinforcement learning with reduced state-action spaces,” In Int. Journal of Game Theory, pp. 547âĂŞ565, 2006.