# MACHINE LEARNING IN MOBILE CROWD SENSING USING DEEP REINFORCEMENT LEARNING WITH DOUBLE Q-NETWORK

[1] Garidepalli Revathi, [2] Vallem Sushma Latha

[1] g.revathi57@gmail.com , [2] sushmacjits@gmail.com

[1, 2] Assistant Professor Department of CSE, Talla Padmavathi College of Engineering,Warangal, Telangana, India.

**ABSTRACT: As a key technique for enabling artificial intelligence, machine learning (ML) is capable of solving complex problems without explicit programming. Mobile crowd sensing (MCS) is a technique of sensing huge amount of data through a large group of mobile devices which has the ability to sense various physical parameters. However, there are some unique challenges has to be overcome in mobile crowd sensing due to the potential limitations of the mobile devices and services, which involves bandwidth, energy and computational power. To overcome these limitations, this paper proposes Deep Reinforcement learning based Cell selection mechanism for Sparse MCS called DR-Cell. Experiments on various real-life sensing datasets verify the effectiveness of DR-Cell over the state-of-the-art cell selection mechanisms in Sparse MCS by reducing up to 15% of sensed cells with the same data inference quality guarantee. Based on dueling network architectures for deep reinforcement learning (Dueling DQN) and deep reinforcement learning with double Q-learning (Double DQN), a dueling architecture based double deep Q-network (D3QN) is adapted in this paper.**

**KEY WORDS: Sparse mobile crowd sensing, deep reinforcement Learning and Double deep Q-network.**

## I. INTRODUCTION

In the beginning, ML has been seen as sub field of computer science that was isolated from other sciences. Today the role of ML in other sciences is realized and it is deployed in many sciences including optimization and control, data mining, medical diagnosis, stochastic modeling and also in wireless communication. An inclusive survey on application of deep learning (DL) in mobile and wireless networking has been published in [3], where a very descriptive explanation about DL is provided with various applications of DL in wireless mobile and network system. More recently, to reduce data collection cost, Sparse MCS [4] is proposed, which collects data from only a few cells while intelligently inferring the data of remaining cells with quality guarantees (i.e., inference error is lower than a threshold). In Sparse MCS, one key issue affecting how much cost can be practically saved is cell selection which sells the organizer decides to collect sensed data from participants [4].

The difficulty of cell selection lies in the fact that collecting data from different cells may lead to diverse inference data quality due to complicated spatio-temporal correlations [2], while it is hard to foreknow inference quality because the ground truth of un-sensed cells is not known. Hence, it is quite challenging to design a good cell selection strategy. In this paper, a new cell selection framework for Sparse MCS, called DR-Cell, with Deep Reinforcement learning techniques will be designed. In general, deep reinforcement learning (RL) can benefit a large set of decision making problems which can be abstracted as ‘an agent needs to decide the action under a certain state’. Our

cell selection problem can also be interpreted as 'an MCS server (agent) needs to choose the next cell for sensing (action) considering the data already collected (state)'. Tai in [5] utilized Deep Q-Networks (DQN) with depth image from RGB-D sensor by estimating q value corresponding to all moving commands realizing a robot exploration without any collision. Tai in [6] proposed a DQN approach using features trained by Convolution Neural Networks from depth image information as input. After training a certain number of times, the robot can travel in new environment autonomously. To solve the disadvantage of over estimation of DQN, Deep Reinforcement Learning with Double Q-learning (Double DQN) was proposed in 2015 by Hasselt in [7], which reduces the observed overestimations by decouple the selection from the evaluation. In this regard, it is promising to apply deep RL on the cell selection problem. To effectively employ deep RL in cell selection for minimizing number of sensed cells, several issues are still being faced.

## II. LITERATURE REVIEW

In reinforcement learning (RL), the agent aims to optimize a long term objective by interacting with the environment based on a trial and error process. Specifically, the following reinforcement learning algorithms are applied in surveyed studies.

**1) Q-Learning:** One of the most commonly adopted reinforcement learning algorithms is Q-learning. Specifically, the RL agent interacts with the environment to learn the Q values, based on which the agent takes an action. The Q value is defined as the discounted accumulative reward, starting at a tuple of a state and an action and then following a certain policy. Once the Q values are learned after a sufficient amount of time, the agent can make a quick decision under the current state by taking the action with the largest Q value. More details about Q learning can be referred to [10]. In addition, to handle continuous state spaces, fuzzy Q learning can be used.

**2) Multi-Armed Bandit Learning:** In a multi-armed bandit (MAB) model with a single agent, the agent sequentially takes an action and then receives a random reward generated by a corresponding distribution, aiming at maximizing an aggregate reward. In this model, there exists a tradeoff between taking the current, best action (exploitation) and gathering information to achieve a larger reward in the future (exploration). While in the MAB model with multiple agents, the reward an agent receives after playing an action is not only dependent on this action but also on the agents taking the same action. In this case, the model is expected to achieve some steady states or equilibrium.

**3) Actor-Critic Learning:** The actor-critic learning algorithm is composed of an actor, a critic and an environment with which the actor interacts. In this algorithm, the actor first selects an action according to the current strategy and receives an immediate cost. Then, the critic updates the state value function based on a time difference error and next, the actor will update the policy. As for the strategy, it can be updated based on learned policy using Boltzmann distribution. When each action is revisited infinitely for each state, the algorithm will converge to the optimal state values.

**4) Joint Utility and Strategy Estimation Based Learning:** In this, algorithm shown in Fig. 7, each agent holds an estimation of the expected utility, whose update is based on the immediate reward. The probability to select each action, named as strategy, is updated in the same iteration based on the utility estimation [9]. The main benefit of this algorithm lies in that it is fully distributed when the reward can be directly calculated locally, as, for example, the data

rate between a transmitter and its paired receiver. Based on this algorithm, one can further estimate the regret of each action based on utility estimations and the received immediate reward and then update strategy using regret estimations. In surveyed works, this algorithm is often connected with some equilibrium concepts in game theory like Log it equilibrium and coarse correlated equilibrium.

**5) Deep Reinforcement Learning:** In [8], authors propose to use a deep NN, called deep Q network (DQN), to approximate optimal Q values, which allows the agent to learn from the high-dimensional sensory data directly and reinforcement learning based on DQN is known as deep reinforcement learning (DRL). Specifically, state transition samples generated by interacting with the environment are stored in the replay memory and sampled to train the DQN and a target DQN is adopted to generate target values, which both help stabilize the training procedure of DRL. Recently, some enhancements to DRL have come out and readers can refer to the literature [1] for more details.

## III. PROPOSED SYSTEM

Mobile Crowd Sensing (MCS) is a technique of sensing huge amount of data through a large group of mobile devices which has the ability to sense various physical parameters. Sensed data is collectively shared to extract information of interest. In the same fashion like mobile block-chain, mobile crowd sensing also has many similarities with IoT. However, there are some unique challenges has to be overcome in mobile crowd sensing due to the potential limitations of the mobile devices and services, which involves bandwidth, energy and computational power. ML can be deployed to intelligently utilize the resources, empower security and increase the quality of service (QoS). Considering that, deep reinforcement learning is used to identify the best cell for obtaining the sensed data in sparse mobile crowd-sensing, which will reduce the amount of data to be sensed while maintaining the same quality. Additionally, mobile crowd sensing face more security threats while sensing and exchanging information. Moreover, to increase the QoS of MCS, Multi-agent Reinforcement Learning can be used, which helps to learn optimal sensing policies and increase users' payoffs.

### A. Problem Formulation

**Problem [Cell Selection]:** Given a Sparse MCS task with m cells and n cycles, using compressive sensing as data inference method and leave-one-out based Bayesian inference as quality assessment method, a minimal subset of sensing cells will be aimed to select during the whole sensing process (minimize the number of non-zero entries in the cell-selection matrix S), while satisfying $(\epsilon, p)$-quality:

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} S(i,j) \text{ ----- (1)}$$

$$s.t., \text{Satisfy } (\epsilon, p) - \text{Quality}$$

A block diagram that illustrates this model in more details is shown in Figure (1). (1) Suppose we have five cells and the current is the 5th cycle; (2) the cell is selected for collecting data and then assess whether the current cycle can satisfy $(\epsilon, p)$ −quality; (3) As it is found that the quality requirement is not satisfied, collecting data from cell 5 will be continued; (4) The quality requirement is now satisfied, so the data collection is terminated for the current cycle and the data of the un-sensed cells is inferred. In this example, there are totally 11 data submissions from participants which can be seen after five cycles. The objective of our cell selection problem is to minimize the number of such data submissions.

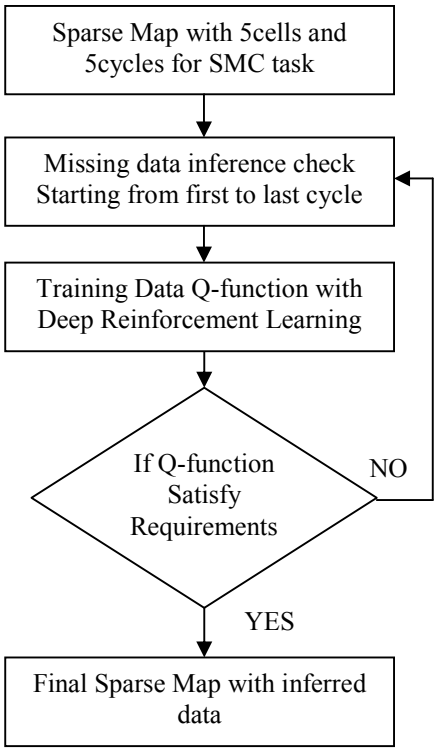


**Fig. 1: FRAMEWORK FOR SPARSE MCS WITH DEEP REINFORCEMENT LEARNING**

### B. Modeling State, Action and Reward

To apply deep RL on cell selection, the key concepts should be needed to model in terms of state, action and reward. Figure 2 illustrates the relationship between the three key concepts in DR-Cell. Briefly speaking, in DR-Cell, based on the current data collection state, it is needed to learn a Q-function which can output reward scores for each possible action. The action of choosing which cell is best as the next sensing cell while rewards indicate how good a certain action is.

(1) State represents the current data collection condition of the MCS task. In Sparse MCS, cell selection matrix (Definition4) can naturally model the state of Sparse MCS, as it records both where and when data is collected from the target area. While keeping the whole historic data collection matrix may lead to a too large matrix, in practice, it is kept that the recent k cycles' cell selection matrix as the state, denoted as $S = [S_{-k+}, \cdots, S_{-1}, S_0]$, where $S_0$ represents the cell selection vector (length is the number of cells) of the current cycle, $S_{-1}$ represents last cycle and so on.

(2) Action means all the possible decisions that are produced in cell selection. Suppose there are totally m cells in the target sensing area, then our next selected cell can have choices, leading to the whole action set $\mathbb{A} = \{1, 2, \cdots m^m\}$ Note that while in practice one cell will not be selected for more than once in one cycle, to make the action set consistent under different states, let us assume that the possible action set is always the complete set of all the cells under any state. If some cells have already been selected in the current cycle, then the probability of choosing these cells is zero.

(3) Reward is used to indicate how good an action is. In each cycle, actions will be selected one by one until the selected cells can satisfy the quality requirement for the current cycle (i.e., inference error $\leq \epsilon$). Satisfying this quality requirement by minimizing the selected cells is the goal of cell selection and should be reflected in the reward modeling. Hence, a positive reward, denoted by R, would be given to an action under a state if the quality requirement is satisfied after the action is taken. In addition, as selecting participants to collect data incurs cost; negative score $-1$ will be also put in the reward modeling of an action. Then, the reward can be written as$R = q \cdot R - c$, in which $q \in \{0,1\}$ means whether the action makes the current cycle satisfy the inference quality requirement. With the above modeling, it is then needed to learn the Q-function which can output the reward score of every possible action under a certain state.

### C. Training Q-function with DRL of Dueling Double Q-Network

Dueling Double DQN is a combination of Double DQN and Dueling DQN. As a result, it overcomes the estimation problem and

improves the performance. As the standard reinforcement learning method, the learning sequences s is regarded as a Markov Decision Process (MDP). The robot makes decisions by interacting with the environment. At each time step, the robot chooses an action $a_t$ according to the current observation $s_t$ at time t firstly, where the observation is a depth image stack consists of four image frames. And then observes a reward signal $r(s_t, a)$produced by reward function. In this paper, the actions contain moving forward, turning half left, turning left, turning left right, turning right. Lastly, the robot transits to the next observations$_{t+1}$. The accumulative future reward is

$$R_t = \sum_r^T \gamma^{r_T} \text{ ---------- (2)}$$

The robot is aimed to maximum the discounted reward. In this formulation, $\gamma$ is a discount factor between 0 and 1 that trades-off the importance of immediate and future rewards? The smaller $\gamma$ is, the more important immediate will be, and vice visa. T means the termination time step. The target of the algorithm is to maximize the action value function Q. Compared with DQN, the Q function of Dueling DQN is

$$Q(s, a; \theta, \alpha, \beta) = V(s, \theta, \beta) + \big(A(s, a; \theta, \alpha) - \max(s, a; \theta, \alpha)\big) \text{----- (3)}$$

Here, denotes the parameters of the convolution layers while and are the parameters of the two streams of fully connected layers. The loss function to train the Q-network is

$$L(\theta) = \frac{1}{n}\sum_k^n \big(y_k - Q(s, a; \theta)\big)^2 \text{ ---- (4)}$$

The network structure is composed of the perception network and the control network. The perception network layer is a 3-layer convolution neural network (CNN), including convolution and activation in every layer.

## IV. RESULTS

The DR-Cell is evaluated with temperature (Sensor Scope) as shown in Figure (2). In the temperature scenario of Sensor-Scope, for the predefined $(\epsilon, p)$-quality, error bound $\epsilon$ is set as 0.2°C and p as 0.8 or 0.9. This quality requirement is that the inference error is $\leq$ 0.2°C for around 80% or 90% of cycles.

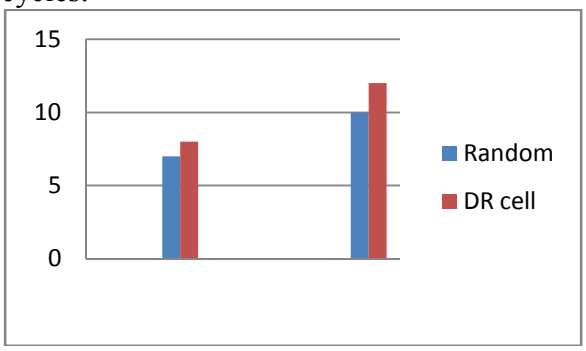


**Fig. 2: NUMBER OF ALLOCATED TASKS**

## V. CONCLUSION

This paper proposes a Deep Reinforcement learning based Cell selection mechanism for Sparse MCS, namely DR-Cell. The three key concepts are properly model in reinforcement learning, i.e., state, action, and reward and then propose Deep Reinforcement Learning of Dueling Double Q-network to learn the Q-function that can output the reward score given an arbitrary state-action pair. Then, under a certain state, it is chosen that the cell with the largest reward as the next cell for sensing. Experiments on various real sensing datasets verify the effectiveness of DR-Cell in reducing the data collection costs.

## VI. REFERENCES

[1] Y. Li, "Deep reinforcement learning: An overview," arXiv:1701.07274v6, Nov. 2018, accessed on Jun. 13, 2019.

[2] L. Wang, D. Zhang, D. Yang, A. Pathak, C. Chen, X. Han, H. Xiong, and Y. Wang, "Space-ta: Cost-effective task allocation exploiting intradata and interdata

correlations in sparse crowd sensing," ACM Transactions on Intelligent Systems & Technology, vol. 9, no. 2, pp. 1–28, 2018.
[3] Chaoyun Zhang, Paul Patras, Hamed Haddadi "Deep Learning in Mobile and Wireless Networking: A Survey," available online arXiv:1803.04311, Sep 2018.
[4] L. Wang, D. Zhang, Y. Wang, C. Chen, X. Han, and A. M'hamed, "Sparse mobile crowdsensing: challenges and opportunities," IEEE Communications Magazine, vol. 54, no. 7, pp. 161–167, 2016.
[5] Tai, L. and M. Liu, Towards Cognitive Exploration through Deep Reinforcement Learning for Mobile Robots. 2016.
[6] Tai L, Liu M . Mobile robots exploration through CNN based reinforcement learning[J]. Robotics and Biomimetics, 2016,3(1):24.
[7] van Hasselt, H., Guez, A., and Silver, D. Deep reinforcement learning with double Q-learning. arXiv preprint arXiv: 1509.06461, 2015.
[8] V. Mnih et al., "Human-level control through deep reinforcement learning," Nature, vol. 518, no. 7540, pp. 529–533, Feb. 2015.
[9] S. M. Perlaza, H. Tembine, and S. Lasaulce, "How can ignorant but patient cognitive terminals learn their strategy and utility?" in Proceedings of SPAWC, Marrakech, Morocco, Jun. 2010, pp. 1–5.
[10] R. Sutton and A. Barto, Reinforcement learning: An introduction, Cambridge, MA: MIT Press, 1998.

**Garidepalli Revathi** is currently working as an Assistant Professor in the Department of Computer Science & Engineering, Talla Padmavathi College of Engineering,Warangal, Telangana, India. She is having 4+ years of experience in teaching and her area of interest includes Networking, Machine Learning, IOT, Cloud Computing etc.

**Vallem Sushma Latha** is currently working as an Assistant Professor in the Department of Computer Science & Engineering, Talla Padmavathi College of Engineering,Warangal, Telangana, India. She is having 9+ years of experience in teaching and her area of interest includes Machine Learning, IOT, Cloud Computing etc.