Avoiding Jammers: A Reinforcement Learning Approach

Serkan Ak and Stefan Brüggenwirth
Physics and Radar Techniques (FHR)
Fraunhofer Institute for High Frequency
53343 Wachtberg, Germany
E-mail: {serkan.ak, stefan.brueggenwirth}@fhr.fraunhofer.de

Abstract—**This paper investigates the anti-jamming performance of a cognitive radar under a partially observable Markov decision process (POMDP) model. First, we obtain an explicit expression for the uncertainty of jammer dynamics, which enables us to discover new insights into the performance metric of the probability of being jammed for the radar beyond a conventional signal-to-noise ratio (SNR) based analysis. Considering two frequency hopping strategies developed in the framework of reinforcement learning (RL), this performance metric is analyzed with deep Q-network (DQN) and long short term memory (LSTM) networks under various uncertainty values. Finally, the requirement of the target network in the RL algorithm for both network architectures is replaced with a softmax operator. Simulation results show that this operator improves upon the performance of the traditional target network.**

I. Introduction

The number of data-hungry wireless devices such as smart phones, tablets, laptops and M2M has been dramatically increasing, recently [1]. Hence, weather and military-radars in the C-band tend to become more vulnerable to the co-channel interference produced by these communication devices [2]. Radar waveform design has been an effective approach to mitigate the impact of other devices [3], [4]. Another technique for alleviating the co-channel interference is to make use of dynamic frequency selection [5]. A cognitive radar [6]–[8] envisioned as an intelligent radar can optimize its operational parameters with respect to data gathered through a feedback loop as a result of the interaction with the surrounding environment. Thus, for a cognitive radar, it is desirable to sense and avoid frequency bands used or intentionally jammed by other transmitters.

Deep reinforcement learning (DRL) has recently become one of machine learning paradigm's crowning achievements [9]. It has recently been used for developing frequency hopping strategies. Han et al. [10] considered a DQN based anti-jamming communication to improve signal-to-interference-plus-noise ratio (SINR). Kang et al. [11] utilized Q-learning and DQN to learn jammer's dynamics and in turn used it to avoid the jammer. Yue et al. [12] considered a multi-user environment where the double DQN algorithm with frequency hopping strategy is used against jamming attack. We utilize the machinery of DRL in order to develop strategies for intelligent frequency hopping, which will in turn reduce the probability of being jammed.

In this paper, we focus on the probability of being jammed performance of a radar under two different strategies based on RL algorithm, named as KARAA strategy and LARA strategy. Our results provide new insights into this performance, i.e., it depends on both the extent of random nature of the jammer and SNR value prior to the detection process of the noisy received signal. To the best of our knowledge, this paper is the first for shedding light on this fact. Utilizing Bellman's optimality in DRL, we show that proposed strategies are considerably better than a purely random hopping strategy in terms of the probability of being jammed. A DQN and LSTM, taking the detection process for a variety of SNR values as input, are separately considered to compute the optimal policy for each strategy, where both of them effectively utilize two significant tricks proposed by Google DeepMind for Deep Q-learning [9]: experience replay and using a target network. The last but not the least, we replace the target network with a softmax operator proposed in [13]. This operator eliminates the need for extra memory in the system required for weight matrices and bias coefficients of the target network, and it also facilitates computations of a target value in the DRL algorithm. Our simulation results show that the performance of a neural network using the softmax operator is at least as good as the performance of a neural network using the target network as the uncertainty of jammer dynamics increases.

Notation: We use calligraphic letters to denote sets. $\mathbb{1}_{\{.\}}$ denotes the indicator function. $|.|$ notation is used to denote the cardinality of a set while $\|.\|_2$ denotes the Euclidean norm of a (complex) number.

II. System Model

In this section, we introduce the details of the studied anti-jamming model. We consider a jammer and radar both of which operate in the same set of N channels. Dynamics of the jammer are generated by a Markovian mechanism studied in Section III-A. At each time slot, the radar desires to intelligently select an unoccupied channel to successfully transmit. To this end, the radar employs and trains a neural network with imperfect observations of all the channels in order to learn this mechanism, and in turn it exploits this

978-1-7281-6813-5/20/$31.00 ©2020 IEEE

information intelligently for avoiding the jammer at every time slot.

The signal model of the system is comprised of two phases: the training phase and the implementation phase. In the training phase, the radar does not transmit any signal but observes the all channels to collect data, which will be used to train the employed neural network in Section III-C via the state-action value function. Formally, we model the detection problem of observation as choosing between the hypothesis of the absence of the jammer in channel k with $\mathcal{H}_0$ and the hypothesis of the presence of the jammer in channel k with $\mathcal{H}_1$. Thus, the received signal in channel k at each time slot at the radar can be written as

$$y_k = \begin{cases} w_k & \text{under} \quad \mathcal{H}_0 \\ g_k x_k + w_k & \text{under} \quad \mathcal{H}_1 \end{cases} \tag{1}$$

where $k \in \{1, \ldots, N\}$, $x_k \in \mathbb{C}$ is the signal in channel k transmitted by the jammer, $g_k \in \mathbb{C}$ is the gain of channel k and w_k is independent and identically distributed Gaussian random variable, i.e., $w_k \sim \mathcal{CN}(0, \mathrm{N}_0)$. For the sake of simplicity, we assume that g_k is unity in the rest of the paper.

The implementation phase consists of two steps, i.e., the initial step and the operation step, respectively. The former step is taken once in the beginning of this phase, and it is used to determine the channel occupied by the jammer at time slot t and in turn the radar employs one of the proposed strategies in Section IV in which it utilizes this information to take an action in the next time slot. In the latter step, according to the chosen strategy, the radar transmits at time slot $t+1$ and at all future time slots. In this step, we assume that the jammer and radar transmit simultaneously at the beginning of each time slot. Although this assumption seems a bit restrictive, it is important not to let either the radar or the jammer take advantage of observing the transmission of the other in the same time slot.

III. Problem Formulation

In this section, we will introduce the details of the uncertainty of jammer dynamics, the studied signal model as a partially observable Markov decision process (POMDP) and the learning algorithm, which we will leverage to propose two strategies for avoiding a jammer in Section IV.

A. Uncertainty of Jammer Dynamics

We start our discussion with uncertainty of jammer dynamics. The study of uncertainty, which may seem a bit of artificial at first glance, will set the stage for us to shed light on the probability of being jammed performance of the radar under a specific strategy.

The distribution of jammer dynamics is modeled as a Markov chain with the state space $\mathcal{S}$, which is statistically independent of the Gaussian random variable in (1). In practice, this model of jammer dynamics can be considered plausible if the transition probabilities are either time-invarient or slowly time-varient. A measure of predictability of sequences generated by a Markov source, which is also called entropy, was studied in Shannon's groundbreaking paper [14]

$$H_i\left(p_{i1}, \ldots, p_{iN}\right) = -\sum_{j=1}^{N} p_{ij} \log_2 p_{ij}, \quad 0 \leq p_{ij} \leq 1 \tag{2}$$

where H_i is the uncertainty of state i, p_{ij} is the transition probability from state i to state j and $i, j \in \{1, \ldots, N\}$. In a similar fashion, the predictability of jammer dynamics, i.e., hopping sequences, may be calculated. However, it is often difficult to obtain uncertainty of a Markov source unless we have the unifilar property [14], [15]. We will make use of two assumptions to utilize this property. The following assumption ensures that we have an ergodic Markov chain.

Assumption 1: For each state $s_i \in \mathcal{S}$, $s_j \in \mathcal{S}$ can be reached in one step from s_i, i.e., $p_{ij} > 0$ and $i, j \in \{1, \ldots, N\}$.
The jammer should be able to hop to any one of the available channels in the next time slot for not leaving a safe channel for the radar, which also justifies Assumption 1 from a practical point of view. We will make the following assumption in order to establish the unifilar property in the Markov chain.

Assumption 2: Each state $s_k \in \mathcal{S}$ is associated with a distinct label.
We will consider orthogonal channels, i.e., frequency bands, as labels to fulfill Assumption 2, e.g., the label of kth state s_k is channel k. In other words, each channel is assigned to a different frequency band, and those frequency bands never overlap due to the orthogonality property.

According to Shannon [14], the uncertainty of sequences X of a unifilar Markov chain is the sum of state uncertainties each of which is weighted by a steady-state probability

$$H\{X\} = \sum\nolimits_{i=1}^{N} \psi_i H_i \tag{3}$$

where ψ_i is the steady-state probability of state i. Note that the convergence of the steady-state probability is independent of the initial state due to Assumption 1. Depending on the size of the transition matrix, the largest H value may also be different. To circumvent this problem, we introduce the normalized uncertainty equation, which is defined as

$$\tilde{H} \triangleq \frac{H\{X\}}{\log_2 \lambda_{\max}} \tag{4}$$

where λ_{max} is the largest eigenvalue of the connection matrix, a simple transformation of the transition matrix [15], and $\log_2 \lambda_{\max}$ corresponds to the maximum uncertainty of the Markov chain [14].

B. Partially Observable Markov Decision Process

We model the received signal in (1) as a discounted POMDP defined by the tuple $(\mathcal{S}, \mathcal{O}, \mathcal{A}, R, P, \gamma)$.

The true state of the system in channel k at time slot t is denoted by $s_k(t) \in \mathcal{S}$, which is defined as

$$s_k(t) \triangleq \mathbb{1}_{\left\{\|x_k\|_2 > 0\right\}}. \tag{5}$$

The noisy observation in channel k at time slot t is denoted by $o_k(t) \in \mathcal{O}$, which is defined as

$$o_k(t) = \{y_k\}. \tag{6}$$

The action of the radar at time slot t denoted by $a_t \in \mathcal{A}$, where $\mathcal{A} = \{1, ..., N\}$, is taken to predict the occupied channel in the training phase or to transmit in a certain channel in the implementation phase.

The reward function $r_t \in R$, where $R = \{0, 1\}$, is defined as

$$r_t \triangleq \mathbb{1}_{\{a_t = \zeta_t\}} \tag{7}$$

where $\zeta_t = k\mathbb{1}_{\{s_k(t)=1\}}$ corresponds to the hypothesis $\mathcal{H}_1$ for channel k. Eq. (7) states that if the jammer is active in the kth channel, i.e., the hypothesis $\mathcal{H}_1$ is true only for channel k, and the kth action has already been taken, i.e., $a_t = k$, then this means a correct prediction of the occupied channel and in turn a positive reward is given.

Without loss of generality, we consider a NxN circulant probability transition matrix with an exponential decay given by (8), where $\vartheta \in [0, 1)$, ϵ is a small positive real number to preserve Assumption 1 as ϑ tends to zero and ρ is $(N-1)/2$ where N is an odd number. Sum of each row of $P_{\vartheta,\rho}$ is one, from which κ should be obtained. However, any stochastic matrix satisfying Assumption 1 may be used. Note that the ith row of $P_{\vartheta,\rho}$ corresponds to the ith state in the Markov chain. Due to the unifilar property in Assumption 2, the label of the ith state is assumed to be channel i. When the jammer starts to operate, it will use a random permutation of the rows of $P_{\vartheta,\rho}$, which will randomly change the transition probabilities of each channel, i.e., randomly changing most (and least) frequent hopping sequence of the jammer. We assume that the transition probabilities in each row of $P_{\vartheta,\rho}$ do not change during both the training and the implementation phases. The discount parameter denoted by $\gamma \in (0, 1)$ is used to put weights on future rewards. For the sake of brevity, we may use s_t and o_t, or s_k and o_k instead of $s_k(t)$ and $o_k(t)$ respectively in the rest of the paper when the content is obvious.

C. Reinforcement Learning Algorithm

We assume that jammer dynamics are obtained by a policy $\pi : \tilde{s}_t \rightarrow \mathcal{A}$, where $\tilde{s}_t$ is the detected channel at time slot t given in (9) and $\xi_{k,t}$ is a function that takes the noisy observation for channel k at time slot t in (6) as an input and determines whether channel k is occupied or not as an output. Hence, as an intermediate step, $\xi_{k,t}$ also performs a detection by utilizing a detector, e.g., the energy detector or the generalized likelihood ratio test. The output of $\xi_{k,t}$ is unity if channel k is occupied at time slot t; otherwise, it is zero. In the training phase, the radar collects the sample $(\tilde{s}_t, a_t, r_t, \tilde{s}_{t+1})$ at each time slot from the environment and stores them into the replay memory $\mathcal{M}$. The state-action value function $Q^\pi : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ is defined as the aggregate discounted reward obtained by when policy π is used to take actions, that is,

$$Q^\pi(s_t, a_t) = \mathsf{E}\left[\sum\nolimits_{m=t}^{\infty} \gamma^m r_m \middle| s_t, a_t\right] \tag{10}$$

where E is the expectation operator. The objective of reinforcement learning algorithm is to find the optimal policy π^*, which obtains the largest aggregate discounted reward, that is,

$$\pi^*(s_t) = \sup_{\pi} Q^\pi(s_t, a_t) \tag{11}$$

where the supremum operator takes all policies into account. However, if the state space is large, there are two approaches to deal with this issue. The first solution is to use a neural network including a target network with parameter θ and a minibatch of independent samples $(\tilde{s}_t, a_t, r_t, \tilde{s}_{t+1})$ from the replay memory $\mathcal{M}$ as in [9] and in turn to minimize the following loss function

$$L(\theta) = \mathsf{E}\left[(\varsigma_t - Q(s_t, a_t; \theta))^2\right] \tag{12}$$

where

$$\varsigma_t = r_t + \gamma \max_{\tilde{a} \in A} Q(s_{t+1}, \tilde{a}; \theta) \tag{13}$$

is the target value. In the second solution, the maximum operator in (13) is replaced by a softmax operator, Mellowmax as proposed in [13]. One intriguing property of this operator is that it works similar to the online RL algorithm.

IV. Strategies for Avoiding Jammer

In this section, we will propose two strategies, which may be employed by the radar for avoiding the jammer: (i) knowledge-based random access agent (KARAA) strategy and (ii) least aggregate reward agent (LARA) strategy.

A. KARAA Strategy

Our discussion begins with KARAA strategy. The KARAA strategy denoted by $\pi_K(s)$ is essentially two-fold: (i) the radar searches for the most frequent hopping sequence of the jammer by the RL algorithm, and (ii) it avoids this sequence in a random fashion; thus, it may reduce probability of being jammed, effectively.

Formally, given the reward function in (7), the most frequent hopping sequence of the jammer corresponds to the optimal policy π^* given in (11) under Bellman's optimality, which yields the largest aggregate discounted reward.

As is defined in Section III-C, the characteristic of actions taken in the first step is deterministic. In the following step, the radar generates a random sequence of actions $\pi_K(s)$, which is defined in (14), where each action $\tilde{a}_{s_i} \triangleq \pi_K(s_i)$, in contrast to the first step, is taken with probability $P(\tilde{a}_{s_i}) = \frac{1}{|\mathcal{A}|-1}$, provided that $a^*_{s_i} \neq a^*_{s_j}, i \neq j, i, j \in \{1, \ldots, N\}$ due to the unifilar property. Note that the inequality $\tilde{a}_{s_i} \neq a^*_{s_i}$ in (14) indicates that the selected action $\tilde{a}_{s_i}$ in state s_i is different from $a^*_{s_i}$ belonging to $\pi^*(s_i)$ for the same state. Also note that it is not necessary to satisfy unifilar property in the second step since the uncertainty of the jammer's Markov source is independent of $\pi_K(s)$.

$$P_{\vartheta,\rho} = \left[\begin{array}{ccccccc} & & & \ddots & & & \\ \kappa\vartheta^{\rho}+\varepsilon & \cdots & \kappa\vartheta+\varepsilon & \kappa+\varepsilon & \kappa\vartheta+\varepsilon & \cdots & \kappa\vartheta^{\rho}+\varepsilon \\ \cdots & \kappa\vartheta^{\rho}+\varepsilon & \cdots & \kappa\vartheta+\varepsilon & \kappa+\varepsilon & \kappa\vartheta+\varepsilon & \cdots \\ & & & & \ddots & & \end{array} \right] \tag{8}$$

$$\tilde{s}_t \triangleq \arg\max_{k} \left\{ \mathbb{1}_{\{\xi_{k,t}=1\}} \middle| \, \xi_{k,t} : o_k\left(t\right) \to \{0,1\}, k \in \{1,\ldots,N\}, t \in \mathbb{Z}_+ \right\} \tag{9}$$

$$\pi_K\left(s\right) \triangleq \left\{ \left(\tilde{a}_{s_1},\ldots,\tilde{a}_{s_N}\right) \middle| \, \tilde{a}_{s_i} \neq a^*_{s_i}, \tilde{a}_{s_i} \in \mathcal{A}, a^*_{s_j} \in \mathcal{A}, i,j \in \{1,\ldots,N\} \right\} \tag{14}$$

B. LARA Strategy

Now, we introduce the LARA strategy denoted by $\pi_L\left(s\right)$. The intuition behind this strategy is to exploit the fact that there is at least one hopping sequence of the jammer, which yields the smallest aggregate discounted reward under Bellman's optimality.

Formally, given the reward function in (7), the least frequent hopping sequence of the jammer corresponds to the optimal policy $\pi_L\left(s\right)$ given in (15) under Bellman's optimality, which yields the smallest aggregate discounted reward, and it is defined as

$$\pi_L\left(s\right) = \inf_{\pi} \mathsf{E}\left[\sum_{m=t}^{\infty} \gamma^m r_m \middle| s_t, a_t \right] = \inf_{\pi} Q^{\pi}\left(s_t, a_t\right) \tag{15}$$

where the infimum operator takes all policies into account. The radar may directly employ $\pi_L\left(s\right)$ in the implementation phase when it hops to a frequency band at each time slot. In comparison to the KARAA strategy, the LARA strategy has two advantages: (i) In the case of full observation, i.e., y_k without noise in (1), it yields the optimum result for the probability of being jammer performance under the Bellman's optimality. (ii) Taking actions in a random fashion is not necessary. In fact, this would lead to a suboptimal result as is shown in the following section.

V. Simulation Results

For the DQN architecture, we consider a fully connected neural network with three hidden layers. Each hidden layer has 32 neurons. The ADAM algorithm [16] is used for the stochastic optimization of the loss function of DQN in (12) on a minibatch of independent samples. The double DQN (DDQN) is utilized to train the DQN since the maximum operator in (13) uses the same values for both selecting and evaluating an action, which may lead to overoptimistic value calculations. Separately, we also consider a single LSTM with 32 hidden units. As is suggested in [9], both architectures make use of a target Q-learning network and ϵ-greedy algorithm for selecting actions. We take 16 samples for training and finish the training in 300,000 time slots. In the backward propagation, the learning rate is fixed at 0.00007 and 0.13 for DQN and LSTM, respectively while LSTM has a single output layer whose learning rate is fixed at 0.01. γ is set to 0.95 and 0.1 for DQN and LSTM, respectively. The Mellowmax coefficient for both DQN and LSTM is set to 15 and 45 in the cases of 5-channel and 9-channel, respectively.

We assume that SNR at the radar varies between 5dB and 10dB in the training phase. Input of both networks is the signal obtained after a detection process, for which we use a simple energy detector since signal x_k in (1) is completely unknown to the radar. Accordingly, detection error may slow down the training of the networks. In order to successfully implement the introduced strategies in Section IV, $\pi^*\left(s_t\right)$ given in (11) needs to be computed as accurately as possible in the training phase.

In Fig. 1, the performance of each type of network is investigated for 9 channels in terms of number of computed $\pi^*\left(s_t\right)$ elements in error in the cases of normalized uncertainty of values 0.85 and 0.9. Here, LSTM and DQN may use either a target Q-learning network in the loss function (12) or Mellowmax operator. In the case of normalized uncertainty of value 0.85, in the lefthand side figure, LSTM using Mellowmax network perfectly computes elements of π^*, LSTM using a target network exhibits a few errors, and DDQN network's performance appears inferior to DQN using Mellowmax. Now, in the case of normalized uncertainty of value 0.9, in the right-hand side figure, DQN with Mellowmax operator outperforms DDQN, however, both types of LSTM network exhibit the same performance, but they are more robust to the increase in uncertainty in comparison to their DQN competitors. This is not surprising because LSTM has a memory, which can extract more information about the past trace of the true state despite the enhanced challenge of uncertainty.

In Fig. 2, we demonstrate the average probability of being jammed as a function of normalized uncertainty $\tilde{H}$ given in (4), where we employ a LSTM using Mellowmax operator to compute (14) and (15). Except the purely random access strategy, we first observe a downward trend in this probability as a function of decreasing values of the $\tilde{H}$, which is an anticipated result since the smaller value of $\tilde{H}$, the more

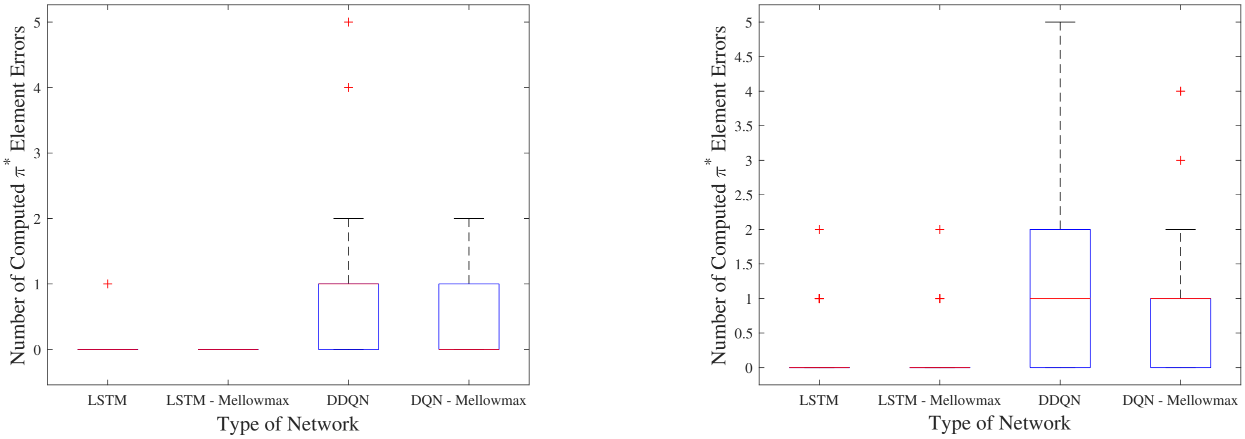

Fig. 1. Computed π^* elements in error for normalized uncertainty of values 0.85 and 0.9 in the lefthand side and righthandside figures, respectively.

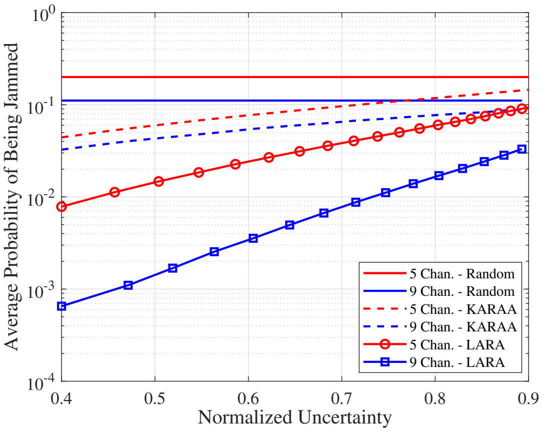

Fig. 2. Average probability of being jammed for a variety of values of normalized uncertainty. In the training, SNR ranges between 5dB and 10dB.

predictability of hopping sequences of the jammer, which is in accordance with the discussion in Section III-A.

We note that both KARAA and LARA exhibit exciting performance even in high $\tilde{H}$ values, especially, LARA provides a great deal of performance improvement. Another interesting observation is that there is a threshold $\tilde{H}$ value, that is approximately 0.77, below which the performance of KARAA for 5-channel turns out to be superior to the random strategy for 9-channel.

Perhaps it's not surprising to see that the performance of KARAA is suboptimal in comparison to LARA since the optimal policy π^*, according to Bellman's optimality, indicates the state with the largest aggregate reward where the jammer will be most probably in the next time slot, but it does not provide any information about $|\mathbb{S}| - 1$ number of other states. As a result of this, selecting one of them in a uniformly random fashion leads to a suboptimal solution.

VI. Conclusions

In this paper, we have studied the probability of being jammed performance of a radar under a POMDP model. To this end, we have utilized DQN and LSTM networks trained under a range of SNR values to compute the optimal policy. Inspired by Shannon's landmark paper, we have proposed a novel approach to analyze the probability of being jammed in terms of the extent of uncertainty of jammer dynamics under two specific strategies - KARAA strategy and LARA strategy. Beyond a traditional SNR based analysis approach, the proposed analysis is of the prime importance for shedding light on the performance achievable by general strategies. Simulation results have confirmed the potential and success of the proposed strategies. LSTM using Mellowmax operator has appeared more robust to uncertainty than the other networks in simulations.

Acknowledgment

The authors would like to thank Fahrettin Gokgoz for helpful comments.

References

[1] "Cisco visual networking index: Global mobile data traffic forecast update, 2017–2022," Cisco white paper, Accessed Nov. 2019.

[2] "Recommendation on C-band meteorological radars design to ensure global and long-term coexistence with 5 GHz RLAN," EUMETNET Recommendation on C-BAND radars, 2008.

[3] A. Aubry, A. De Maio, M. Piezzo and A. Farina, "Radar waveform design in a spectrally crowded environment via non-convex quadratic optimization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 2, pp. 1138–1152, Jul. 2014.

[4] A. Aubry, V. Carotenuto and A. D. Maio, "Forcing multiple spectral compatibility constraints in radar waveforms," *IEEE Signal Processing Letters*, vol. 23, no. 4, pp. 483-487, Apr. 2016.

[5] F. Hessar and S. Roy, "Spectrum sharing between a surveillance radar and secondary wi-fi networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 3, pp. 1434–1448, Jul. 2016.

[6] S. Haykin, "Cognition is the key to the next generation of radar systems," in *Proc. IEEE 13th Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop*, Marco Island, FL, USA, Jan. 2009, pp. 463–467.

[7] S. Bruggenwirth, “Design and implementation of a three-layer cognitive radar architecture,” in *Proc. 50th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, USA, Nov. 2016, pp. 929–933.

[8] A. Farina, A. De Maio, and S. Haykin, “The impact of cognition on radar technology,” SciTech Publishing, New Jersey, USA, 2017.

[9] V. Mnih, K. Kavukcuoglu, D. Silver, A.A. Rusu, J. Veness, M.G. Bellemare, A. Graves, M. Riedmiller, A.K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[10] G. Han, L. Xiao, and H. V. Poor, “Two-dimensional anti-jamming communication based on deep reinforcement learning,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, New Orleans, LA, USA, Mar. 2017, pp. 2087-2091.

[11] L. Kang, J. Bo, L. Hongwei, and L. Siyuan, “Reinforcement learning based anti-jamming frequency hopping strategies design for cognitive radar,” in *Proc. IEEE International Conference on Signal Processing, Communications and Computing*, Qingdao, China, Sept. 2018, pp. 1-5.

[12] Y. Bi, Y. Wu, and C. Hua, “Deep reinforcement learning based multi-user anti-jamming strategy,” in *Proc. International Conference on Communications*, Shanghai, China, May 2019, pp. 1-6.

[13] S. Kim, K. Asadi, M. Littman, and G. Konidaris, “Removing the target network from deep Q-networks with the Mellowmax operator,” in *Proc. the 18th International Conference on Autonomous Agents and MultiAgent Systems*, Montreal, Canada, May 2019, pp. 2060-2062.

[14] C.E. Shannon, “A mathematical theory of communication,” *Bell Syst. Tech. J.*, vol. 27, pp. 379-423, Jul. 1948.

[15] R.B. Ash, “Information theory,” Inter-science Publishers, New York, USA, 1965.

[16] D.P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. Int. Conf. Learn. Represent.*, Dec. 2015, pp. 1–15.