Coexistence Management for URLLC in Campus Networks via Deep Reinforcement Learning

Behnam Khodapanah*, Tom Hößler*†, Baris Yuncu*, André Noll Barreto†, Meryem Simsek‡, Gerhard Fettweis*
*Vodafone Chair Mobile Communications Systems, Technische Universität Dresden, Germany
Email:{behnam.khodapanah, tom.hoessler baris.yuncu gerhard.fettweis}@tu-dresden.de
†Barkhausen Institut, Dresden, Germany; Email: andre.nollbarreto@barkhauseninstitut.org
‡International Computer Science Institute, Berkeley, USA; Email: simsek@icsi.berkeley.edu

***Abstract*—Increased usage of wireless technologies in unlicensed frequency bands inevitably increases the co-channel interference. Hence, for applications such as ultra-reliable-low-latency-communications (URLLC) in factory automation, the interference should be avoided. An intelligent coexistence management entity, which dynamically distributes the time and frequency resources, has been shown to be greatly beneficial in boosting efficiency and avoiding crippling interruptions of the wireless medium. This entity also supports multi-connectivity schemes, which are crucial for industry-level reliability requirements. The proposed governing technique of the coexistence management is a deep reinforcement learning (DRL) method, which is a model-free framework and channel allocation decisions are learned merely by interactions with the environment. The simulation results have shown that the employed method can greatly increase the reliability of the wireless network, when compared with legacy methods.**

***Index Terms*—Coexistence Management, URLLC, Campus Network, Deep Reinforcement Learning, 5G**

I. Introduction

URLLC is regarded as one of the most innovative features brought in the fifth-generation mobile networks (5G) for mission-critical communications such as industrial automation [1]. The fourth generation (4G) cellular networks cannot satisfy the strict delay and reliability requirements of the URLLC applications. The hybrid automatic repeat request procedure can guarantee reliability but comes with cost of delay. Moreover, factory automation requires multi-user, low-cost, and worldwide applicability. Thus, the 5 GHz ISM bands are a promising candidate for wireless automation applications. However, since everybody is allowed to use these unlicensed frequencies, co-channel interference is a major challenge. The interference problem will be inevitably amplified as the number of independent networks are increased. A "campus network" is an exclusive mobile network which is designed to meet the specific needs of users and satisfy the future requirements of industry 4.0 [2]. With increased use of unlicensed bands in these networks, their coverage could easily overlap. In the norm IEC 62657-2 [3] for industrial radio communication systems it is recommended to use an active coexistence management for reliable channel utilization. To ensure that such entity can be agile and reliable enough, an automatic cooperative coexistence management concept is proposed.

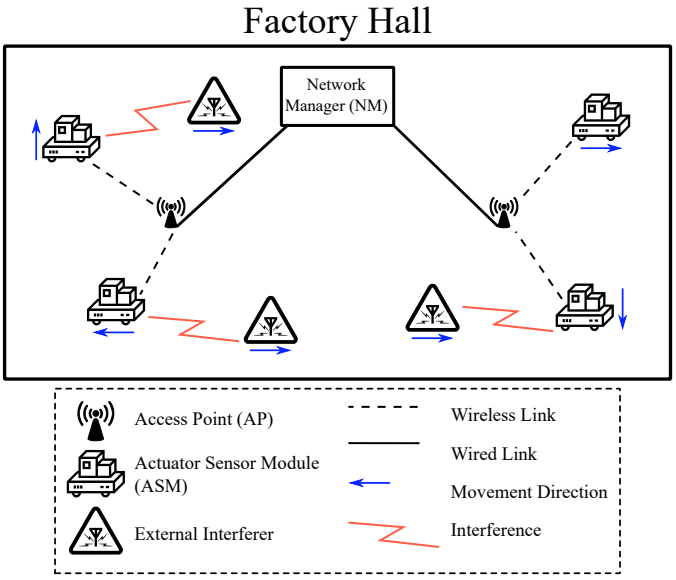

Fig. 1. In a factory hall, the actuator sensor modules (ASMs) are connected to the access points (APs) wirelessly. However, there might well be external interferers that are using the same time-frequency block and can cause collisions. To ensure that the external interferers do not cause interruptions in the wireless link, the network manager (NM) is dynamically altering channel allocations of ASMs.

Beside the interference problem, the coexistence management scheme should be able to cope with the special fading characteristics of the industrial environment. Due to the more open building layout, the presence of machinery and highly reflective materials (e.g. aluminium or steel), the radio channel in an industrial environment will act differently from, for example, office buildings. This emphasizes the need for specifically developed wave propagation models for industrial environments. In this article the large- and small-scale fading models are based on [4] and [5] respectively, which are based on the channel measurements in different frequencies and different transmitter/receiver setups.

It is well known that introducing redundancy to the wireless system can improve reliability. In this context, multi-connectivity is a justifiable solution, where the redundancy is realized by means of having multiple links carrying the same information. In [6], the benefits of multi-connectivity for the secondary users in a cognitive radio environment has been

demonstrated. The coexistence management entity should be able to accommodate the multi-connectivity aspect as well.

Many cooperative techniques have been proposed in the past years. For example, [7] proposed a method for cooperative load resource optimization by considering inter-domain co-channel interference for multi-domain Het-Net. This proposed inter-domain load balancing scheme is focusing on balancing the radio resource cost and co-channel interference. However, that paper neglects external interference sources. In [8], a cooperative load-balancing framework for multi-domain WLANs operating in an interference environment is proposed. This work is focused on improving network utilization over multiple WLANs by controlling the available channels at each AP. However, the proposed load balancing scheme does not dynamically adjust channel assignment at each AP.

In this paper, the coexistence management problem is studied for a campus network in the area of industrial automation (see Fig. 1). In order to address the self-coexistence problem in a campus network, an adaptive interference- and fading-aware dynamic channel allocation strategy based on deep reinforcement learning (DRL) is presented. Reinforcement learning provides model-free control policies that are learnt merely by interactions with the environment. In [9], it has been shown that such algorithms are able to exhibit human-level control in certain difficult tasks. In recent years, some successful applications of DRL in the field of wireless communication were developed. In [10], a dynamic channel allocation (DCA) method based on DQN for multi-beam satellite systems was proposed. Their simulation results show that the proposed method outperforms fixed channel allocation and location based dynamic channel allocation methods by means of lower blocking probability and increased spectrum efficiency.

This paper is structured as follows. In Section II, we describe the system model for simulating an industrial automation environment. Next, in Section III, we introduce different coexistence management schemes, specifically the deep reinforcement learning based approach. The evaluations can be found in Section IV, where the performances of different coexistence management algorithms are compared. Finally, in Section V we conclude the paper.

II. System Model

In an automated factory hall, we assume there are M actuator-sensor modules (ASMs), which are the industrial equipment that are connected to the factory's wireless network. Each of these ASMs is able to measure the total received power over C different channels and reports them back to the connected access points (APs), which itself relays it to the network manager (NM). They are connected to one of N APs and the APs themselves are connected to a central entity named NM, who makes decisions on channel allocations for APs. To enable multi-connectivity, it is assumed that the ASMs are connected via two different channels to the APs. Furthermore, K external interferers are assumed to be in the vicinity, which are using a subset of the C available channels and therefore interfering with the ASMs. Fig. 1 illustrates how different entities are connected.

A. Interference and Fading

The vulnarability of the wireless link is mainly caused by the interference and fading of the channel. The interference source can be a neighbouring AP, when both of them are transmitting to their respective ASM in the same time-frequency block. This adverse effect can be minimized if the NM properly distributes the resources. Another source of interference are the external devices, i.e., the devices that use the unlicensed band time-frequency resources, but are not controlled via the NM. The signal-to-interference-plus-noise-ratio (SINR) of ASM m connected to AP n on channel c can be written as

$$\gamma_{m,n}^{c} = \frac{P_{m,n}^{c}}{\sum\limits_{n' \neq n}^{N} P_{m,n'}^{c} + \sum\limits_{k \in \mathbb{K}^c} P_{k}^{c} + \eta}, \tag{1}$$

where $P_{m,n}^{c}$ denotes the received power of ASM m from AP n on the channel c, P_{k}^{c} is the received power from interferer k, $\mathbb{K}^c$ is the set of interferers on channel c and η is the noise power. When the SINR of a channel drops below a guard threshold ρ_g, this channel is considered blocked, since the probability of experiencing deep fade in the next time steps is higher.

To model the fading of wireless channels between the ASMs, APs and interferers, large- and small-scale fading models are considered. In the industrial environment, large shadowing is expected due to the presence of heavy machinery. This effect is modeled by a zero-mean log-normal distribution [4]. Furthermore, to model the small-scale fading, the links between the ASMs and APs are modeled by a Rician fading because it is reasonable to assume a dominant line-of-sight or a dominant reflection path inside the industrial environment [5]. Moreover, if the APs are attached to the ceilings, the assumption of line-of-sight component is more realistic. However, to model the links between the ASMs and external interferers, a Rayleigh fading is assumed.

B. Multi-Connectivity

To ensure a reliable communication within the factory hall, in this work we consider a multi-connectivity scheme, where two channels are assigned to each ASM simultaneously. For the combining scheme we choose selection combining (SC), which is a fast and low-complexity method. In this scheme, the transmissions from the channel with highest SINR is chosen and the transmission of the other channel is discarded. Therefore, the effective SINR of the ASM m in the AP n is

$$\gamma_{m,n} = \max\{\gamma_{m,n}^{c_1}, \gamma_{m,n}^{c_2}\} \tag{2}$$

where $\gamma_{m,n}^{c_1}$ and $\gamma_{m,n}^{c_2}$ are the SINR of the ASM m experienced over channels c_1 and c_2, respectively.

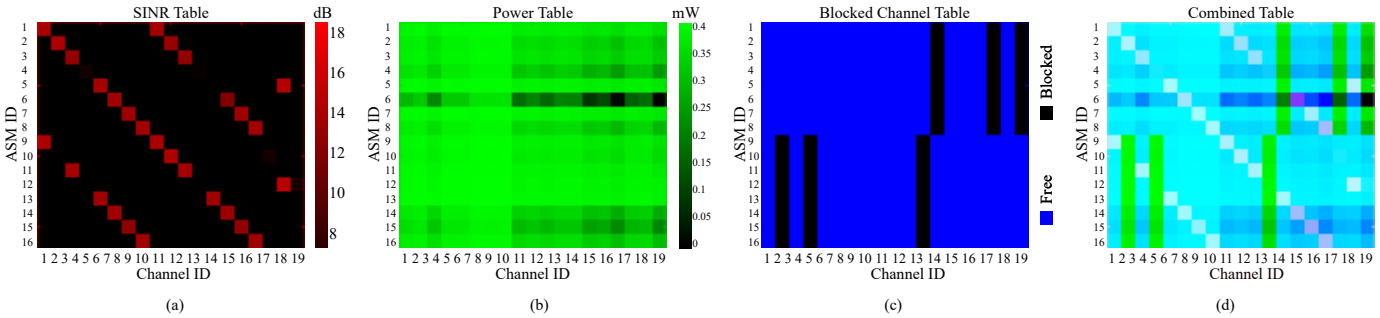

Fig. 2. Exemplary SINR (a), power (b) and blocked channels (c) tables. These tables can be encoded in red, green and blue channels and combined in one image (d).

C. Performance metric: Outage Probability

The metric for evaluating different coexistence management schemes in this work is the outage probability, which we define in the following. In our setup, outage occurs when both of the channels associated with each ASM are experiencing deep fade or large interference, i.e.,

$$P_{m,n}^{\text{out}} = P((\gamma_{m,n}^{c_1} < \rho_g) \cap (\gamma_{m,n}^{c_2} < \rho_g)). \tag{3}$$

To calculate the outage over all the ASMs, during a simulation with duration of T timesteps, the number of times ASM m has experienced an outage is counted and represented as o_m. Thereafter, we sum up all the outages and divide it with the number of ASMs M and simulation duration T, i.e.,

$$P^{\text{out}} = \frac{1}{M \cdot T} \sum_{m=1}^{M} o_m. \tag{4}$$

III. Coexistence Management Schemes

In this section, firstly, we describe the NM entity and its input and output. Thereafter, we introduce the legacy coexistence management techniques, along with the strategy based on deep reinforcement learning.

A. Network Manager (NM)

The NM is responsible for regulating the channel allocations of the ASMs. Therefore, during the run time the NM sends the channel allocation table to the APs, which is the mapping between C channels and M ASMs. This table can be updated periodically or as a reaction to blockages experienced by the ASMs. Each ASM senses the spectrum and measures the received power of that ASM over all frequencies. Therefore, each ASM sends a vector of size C to the NM at each time step. Furthermore, each ASM is measuring the SINR in the two channels that are allocated to it. In the NM, the SINRs received over M ASMs are compared with the guard threshold ρ_g and the ones which are below the threshold are marked as blocked channels. Fig. 2 illustrates examples of the power table, SINR table and blocked channel tables. We can accommodate each table as a color channel of a RGB image and get a combined table which has all of the relevant information. These tables are periodically made available to the NM and, based on them, the NM can dynamically allocate proper assignments, i.e., manage their coexistence.

B. Legacy Coexistence Management

The legacy strategies could be static or dynamic, meaning that the channel allocation can remain constant or be changed dynamically, either periodically or triggered by an outage event.

1) Static Channel Allocation (SCA): In the SCA technique, a set of channels are statically allocated to each ASM at the initialization phase and do not changed during the run time. This technique is not able to cope with the time-varying interference and fading. This coexistence management scheme is only investigated to constitute a comparison basis.

2) Random Channel Allocation (RCA): To cope with the time-varying nature of the wireless link, it is essential that the NM has the ability to dynamically change the channel allocations to avoid incoming interference and channels that will face deep fade. Therefore, in this technique, when a channel is marked blocked by the NM, a randomly selected free channel will be assigned to the ASM that is experiencing blockage.

C. Deep Reinforcement Learning based Channel Allocation

Firstly, the concept of Reinforcement Learning (RL) is shortly introduced. Thereafter, the RL method is applied to the problem of coexistence management.

1) Concept of Reinforcement Learning: Reinforcement learning is a subject of machine learning, in which an agent takes actions in an environment and receives rewards. The aim is to learn a policy for the agent, such that the cumulative reward of the agent is maximized. To formulate the interactions between the environment and the agent, Markov Decision Process (MDP) is used, i.e., the agent at time step t selects an action a_t and the environment responds to these actions with moving to new states s_{t+1} and giving rewards r_{t+1}. The action-value function is defined as

$$Q(s_t, a_t) = \mathbb{E}\{R|s_t, a_t\} = \mathbb{E}\{\sum_{k=0}^{\infty} \eta^k \cdot r_{t+k+1}|s_t, a_t\}, \tag{5}$$

where R is the cumulative reward, r_t is the reward received at time step t and $\eta \in [0, 1]$ is the discount factor. If the state

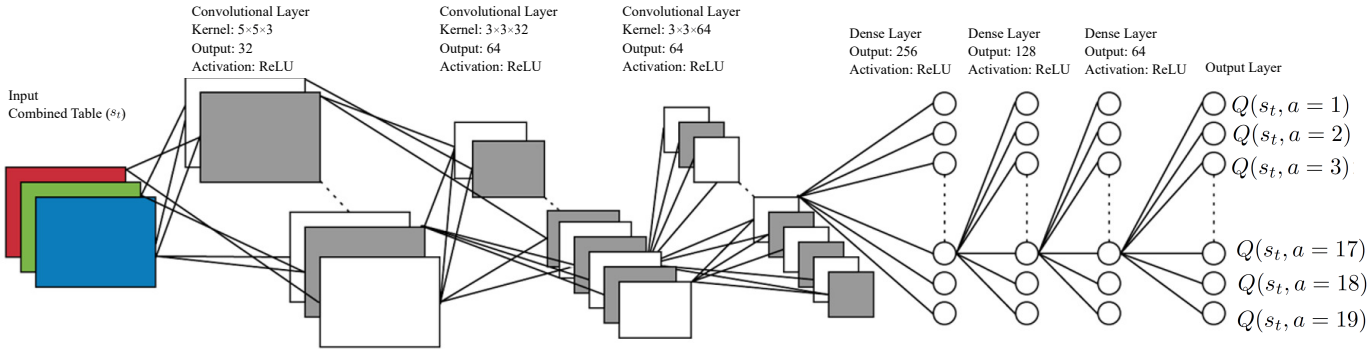

Fig. 3. Architecture of the DRL.

space is discrete and finite, off-policy Q-learning with time difference (TD) update rule for $Q(s, a)$ is [11]

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[R + \eta \max_a Q(s_{t+1}, a) - Q(s_t, a_t)]. \tag{6}$$

After the training is finished, we choose the action that maximizes this function, so that we can collect maximum cumulative rewards.

In practice, however, the states can have high dimensions with continuous space. Therefore, the use of an approximate method that can handle large and continuous state dimensions is necessary. The deep Q-network (DQN) architecture uses artificial neural networks to approximate the $Q(s, a)$ function as $Q(s, a; \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a vector containing all of the weights and biases in the neural network and parametrizes Q. $\boldsymbol{\theta}$ is the parameter that is updated during learning. At time step t, the DQN takes the state s_t as input and outputs $Q(s_t, a; \boldsymbol{\theta}) \quad \forall a \in \mathbb{A}$, where $\mathbb{A}$ is the action space set. This means that the number of output nodes of the DQN is equal to number of possible actions.

Each of the experiences between the agent and the environment can be summarized in a tuple $< s_t, a_t, s_{t+1}, r_{t+1} >$. According to [9], to avoid the correlation between the consecutive samples from the environment, we store the experiences in a replay buffer and in each training iteration a random mini-batch of the experiences is chosen for training the network. Furthermore, to avoid instabilities during training caused by a moving target in (6), it is suggested to create two neural networks, one which is updated every training step $Q(s, a; \boldsymbol{\theta})$ and another one, the target network $Q(s, a; \boldsymbol{\theta}^-)$, which is updated every τ steps [9], [12]. Besides, to avoid the influences of the replay buffer memory size on training, combined replay has been proposed in [13]. In this method, the most recent experience is added to the random mini-batch samples. The labels for training are

$$L(s_t, a) = \begin{cases} r_{t+1} + \eta Q(s_{t+1}, \arg\max_{a'} Q(s_{t+1}, a'; \boldsymbol{\theta}); \boldsymbol{\theta}^-) & \text{if } a = a_t \\ Q(s_t, a, \boldsymbol{\theta}) & \text{if } a \neq a_t \end{cases}, \tag{7}$$

Initialize Q-Net. weights $\boldsymbol{\theta}_0 \leftarrow \boldsymbol{\theta}_{\text{Random}}$;
Initialize Target Net. weights $\boldsymbol{\theta}^- \leftarrow \boldsymbol{\theta}_0$;
Populate replay buffer with minimum samples;
for $t \leftarrow 1$ **to** t_{max} **do**
  if $mod(t, \tau) = 0$ **then**
    Update Target Net. weights $\boldsymbol{\theta}^- \leftarrow \boldsymbol{\theta}_t$;
  end
  Sample a random number $\rho \leftarrow rand(0, 1)$;
  Based on greedy policy calculate $\epsilon(t)$;
  if $\rho < \epsilon(t)$ **then**
    Sample a random action $a_t \in \mathbb{A}$;
  else
    Q-Net. action $a_t = \arg\max_{a \in \mathbb{A}} Q(s_t, a; \boldsymbol{\theta}_t)$
  end
  Pass the a_t to the environment;
  Observe r_{t+1}, s_{t+1} from environment;
  Recent experience $\leftarrow < s_t, a_t, r_{t+1}, s_{t+1} >$;
  Add the recent experience to the replay buffer;
  Sample a mini-batch from the replay buffer;
  Augment the mini-batch with recent experience;
  Calculate the label according to (7);
  Train the network $\boldsymbol{\theta}_t \leftarrow \boldsymbol{\theta}_{t+1}$;
end

Algorithm 1: DRL Training.

where $L(s_t, a)$ is the calculated label for all $a \in \mathbb{A}$ [12]. To encourage exploration in the beginning phase of training, random actions are taken with probability of $\epsilon(t)$, which is defined as

$$\epsilon(t) = \begin{cases} 1 - \frac{1-\epsilon_f}{t_{max}} t & \text{if } t < t_{max} \\ \epsilon_f & \text{otherwise} \end{cases}, \tag{8}$$

where ϵ_f is the final exploration rate after t_{max}. After t_{max}, the agent is taking actions based on the learned policy, i.e., exploration phase.As time moves on this probability decays, so that the exploitation phase takes over. Algorithm 1 illustrates the process of training in more detail. Furthermore, Table I lists the parameters of the DRL.

2) Application of DRL to Coexistence Management: To apply the framework of DRL to the coexistence management

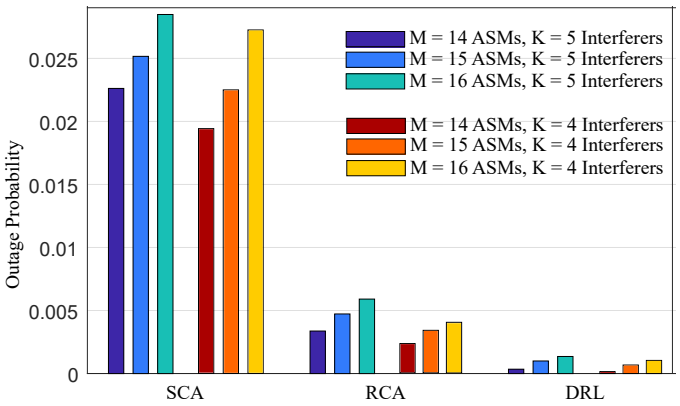

Fig. 4. Comparison of outage probabilities of different coexistence management techniques.

problem, we need to define the state, action and rewards as follows

- *State*: To enable the agent to fully observe the environment, we define the state s to be all the information available to the NM. Therefore, the state s consists of the SINR table, power table and blocked channel table. One can reformulate these tables as an $M \times C \times 3$ array (see Fig. 2).
- *Action*: The actions are the choice of the channel to be selected, i.e., there are C channels available and therefore, there are C different actions that the agent can take.
- *Reward*: Since we want to decrease the outage probability, after each action a negative reward is assigned if after the change in the channel allocation, there is an outage. On the contrary, if after the channel allocation, there were no outages, a positive reward is assigned. Thus the agent is motivated to learn to select the channels that will avoid outages.

Since the states are in the form of images, as suggested in [10], we employ a convolutional neural network (CNN) as the function approximator which is accompanied by the fully-connected layers. Fig. 3 illustrates the architecture of the neural network. Furthermore, since the channel assignment should be carried our for each of the ASMs, the first row of the $M \times C \times 3$ image is swapped with the ASM that is being decided on. Thus the agent is always making the best decision for the ASM in the first row and separate agents for separate ASMs is not required.

IV. Simulation Setup and Evaluation

The factory hall is assumed to have a width of 50 meters (in y-axis) and length of 100 meters (in x-axis). The height of this building is assumed to be 6 meters (in z-axis). There are two APs attached to the ceiling and they are located at $\{x = 25, y = 25, z = 6\}$ and $\{x = 25, y = 75, z = 6\}$ meters. Furthermore, the ASMs are at a height of $z = 1$ meter and can move throughout the hall. However, we pose an additional constraint that they are not allowed to cross a border at $y = 50$ meters. This constraint is required to avoid simulating the handover process, since it is out of scope of this paper. Half of of the ASMs are served by the AP on the left and the other half by the AP on the right. The interferers, on the other hand, are generated at the left edge of the factory hall and move to the right and occupy one of the $C = \{19\}$ channels. They are assumed to be at a height of $z = 7$ meters, to resemble a pedestrian in the second floor. The interferer is removed if it exits the rights side of the factory or if it is randomly selected for removal. At each time step, the interferer will be removed with probability of 0.001. When an interferer is removed, a new one appears on the left side of the factory and occupies a randomly selected channel. The ASMs' locations are randomly selected in the beginning of the simulations and they pick a random direction to move, i.e., north, east, south or west. Moreover, to follow the standard in [14], the transmission power of each channel is different. The transmission power of channels $c = 1, \cdots, 4$ is 23 dBm, $c = 5, \cdots, 8$ is 20 dBm and $c = 9, \cdots, 19$ is 27 dBm. The simulation scenarios consider the number of ASMs of $U = \{14, 15, 16\}$, and the number of interferers $K = \{4, 5\}$. Each simulation corresponds to 100,000 steps because we simulate 100 seconds with the resolution of 1000 samples per second. Table I summarizes the simulation and DRL parameters.

TABLE I
Simulation and DRL Parameters

Simulation Parameters	Value
Simulation Duration (s)	100
Interval between steps (ms)	1
Total number of steps T	100.000
Factory Hall (m)	50×100
Timed update event frequency t_{f}	100
ASM speed $(m \cdot s^{-1})$	1
Interferer speed $(m \cdot s^{-1})$	5
Guard threshold ρ_g (dB)	7
Carrier frequency (GHz)	5.2
Bandwidth (MHz)	20
G_t and G_r (dBi)	2
Rician K-factor (dB)	14.7
Number of multi-links	2
Number of channels	19
DRL Parameters	**Value**
Final exploration rate ϵ_f	0.01
Max. exploration step t_{max}	10000
Replay memory size	50000
Mini-batch size	32
Target network update freq.	80
Reward for correct assignment	+10
Reward for incorrect assignment	-10
Discount factor η	0.9

To compare the performance of different coexistence management techniques, their outage performance has been calculated and shown in Fig. 4. For each of the test cases, 50 different realizations have been conducted. First of all, as we increase the number of interferers, the outage probability increases. That is because with more interferers, they are occupying more channels and therefore, increasing the chance of collision with channels that are already assigned to the

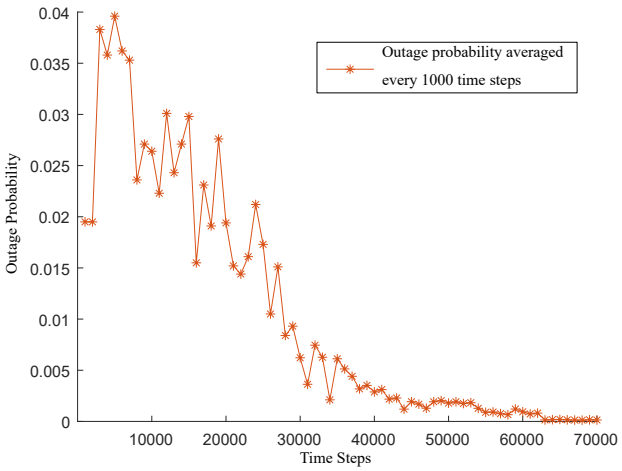

Fig. 5. Outage probability during the training of the DRL.

ASMs. Moreover, as the number of ASMs is increased, the number of free channels decreases and the NM does not have as much freedom to allocate more suitable channels to the ASMs that are experiencing deep fade or intense interference. Comparing the coexistence management schemes, the static channel allocation (SCA) has the highest outage probability. This is because in this technique there is no reaction to the incoming interferers or the deep fade. This results in a high number of outage cases. This clearly shows that dynamic channel allocation is necessary to ensure reliable communications for the URLLC cases. The next technique is the random channel allocation (RCA), where a blocked channel triggers the NM to allocate a new channel. This newly assigned channel is chosen randomly from the pool of free channels. Clearly with comparison to the SCA, reacting to the blocked channels will increase the reliability of the system greatly.

Finally, we can see that the DRL approach has the best performance. This is because the NM, at each time step, has learned to choose the channel that minimizes the outage. Therefore, this method is not waiting for an outage to happen and then react to it, but is always dynamically changing the channel allocations. If an external interferer approaches the hall or some channels begin to experience deep fades, the NM will have learned to quickly identify those and react to them before it is too late. The more interesting observation that can be made is that this knowledge has been learnt via only trial and error and no modeling or domain expertise was required. Fig. 5 shows the performance of the DRL agent during the training. Every 1000 steps the average outage probability is calculated. During the first 10000 steps, the agent is mostly taking random actions to explore the environment (exploration phase) and hence, the outage probability is increased. As the time goes on, we start taking the actions based on the DRL (exploitation phase), where the channels that are assigned will cause least amount of outage.

V. Conclusion

In this paper, a cooperative automatic coexistence management system based on a centralized unit (called NM) is proposed for URLLC in unlicensed frequency bands. This system is based on the deep reinforcement learning framework, which does not require any modeling or domain expertise and only depends on the interactions with the environment to learn a policy. This approach considers all of the received power and SINR tables collected from the ASMs and decides on the channel allocation. We have shown that compared with the legacy techniques, the DRL based technique is superior. To further increase the reliability, we expect a more sophisticated combining scheme can have a big influence. Furthermore, employing a recurrent neural network that can produce a channel allocation table for all of the ASMs at each time step, can increase the responsiveness of this technique.

Acknowledgments

This work was supported by the project “fast automation” and the Federal Ministry of Education and Research of the Federal Republic of Germany (BMBF) within the initiative “Region Zwanzig20” under project number 03ZZ0510F. This research was co-financed by public funding of the state of Saxony/Germany. We would also like to thank the Center for Information Services and High Performance Computing (ZIH) at TU Dresden for their generous allocations of computer time.

References

[1] P. Popovski *et al.*, “Wireless Access for Ultra-Reliable Low-Latency Communication: Principles and Building Blocks,” *IEEE Network*, vol. 32, no. 2, pp. 16–23, March 2018.

[2] Telekom 5g technology in campus network. [Online]. Available: https://www.telekom.com/en/company/details/5g-technology-in-campus-networks-556692

[3] International Electrotechnical Commission, “Industrial communication networks – wireless communication networks – part 2: Coexistence management,” 2013.

[4] T. S. Rappaport and C. D. McGillem, “UHF fading in factories,” *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 1, pp. 40–48, Jan 1989.

[5] E. Tanghe *et al.*, “The industrial indoor channel: large-scale and temporal fading at 900, 2400, and 5200 MHz,” *IEEE Transactions on Wireless Communications*, vol. 7, no. 7, pp. 2740–2751, July 2008.

[6] H. Li and L. Qian, “Enhancing the reliability of cognitive radio networks via channel assignment: risk analysis and redundancy allocation,” pp. 1–6, March 2010.

[7] B. Li *et al.*, “Multi-domain load resource optimization for heterogeneous network in LTE-A,” pp. 215–219, Sep. 2012.

[8] J. Xie and I. Howitt, “Multi-domain wlan load balancing in wlan/wpan interference environments,” *IEEE Transactions on Wireless Communications*, vol. 8, no. 9, pp. 4884–4894, September 2009.

[9] V. Mnih *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, pp. 529–33, 02 2015.

[10] S. Liu *et al.*, “Deep Reinforcement Learning Based Dynamic Channel Allocation Algorithm in Multibeam Satellite Systems,” *IEEE Access*, vol. 6, pp. 15 733–15 742, 2018.

[11] R. S. Sutton and A. G. Barto, “Reinforcement learning: An introduction,” *IEEE Transactions on Neural Networks*, vol. 16, pp. 285–286, 1988.

[12] H. V. Hasselt, “Double q-learning,” in *Advances in Neural Information Processing Systems 23*. Curran Associates, Inc., 2010, pp. 2613–2621.

[13] S. Zhang and R. S. Sutton, “A Deeper Look at Experience Replay,” *CoRR*, vol. abs/1712.01275, 2017.

[14] European Telecommunications Standards Institute, “ETSI EN 301 893 v2.0.7,” 2016.