Reinforcement Learning Based Backoff Management for Industry 5.0

Alessia Tarozzi *, Mahin Ahmed †, Hans-Peter Bernhard † ‡, Roberto Verdone *
* University of Bologna (DEI) / WiLab (CNIT), Italy
† Silicon Austria Labs GmbH, Austria
‡ Johannes Kepler University, Austria

***Abstract*—Industry 5.0 marks a transition from the digitalization focus of Industry 4.0 to a paradigm emphasizing resilience, sustainability, and human-centric processes. In such dynamic networks, reinforcement learning (RL) algorithms can play a crucial role in enhancing performance. The paper proposes a novel approach using a centralized RL algorithm to optimize the medium access control for a moving autonomous guided vehicle (AGV) on an industrial shop floor. This ensures uninterrupted production flow by dynamically managing the network traffic. The use case scenario considers a mobile AGV transmitting data to a base station (BS) within a harsh industrial environment. It uses the RL algorithm to dynamically select an optimal backoff (BO) time for an ALOHA-like channel access protocol. This enables accurate data transmission without prior knowledge of the industrial environment. The results show an improvement of up to 34.6% in success probability compared to traditional BO design approaches. The RL model achieves outstanding performance, guaranteeing a minimum success probability of 99.46%.**

***Index Terms*—RL, Industry 5.0, Backoff, THz**

I. Introduction

Industrial applications pose significant challenges due to the presence of both static and moving devices, which demand rigorous and heterogeneous requirements in terms of data rate, latency, and reliability [1]. Future networks will depend on 6G technologies, leveraging terahertz (THz) frequencies to satisfy these demands [2]. However, THz frequencies face challenges such as high attenuation and molecular absorption [3]. In this context, it is envisioned that the emerging paradigm of the Industry 5.0 would enable the automation of production processes, enhancing production efficiency, logistics, and storage. It integrates three key components: connectivity, autonomous cyber-physical systems, and the human in the loop. For connectivity, sensors access the network via medium access control (MAC) protocols, including contention-based protocols, such as ALOHA [4] and carrier sensing multiple access (CSMA) [5], as well as scheduled protocols [6]. For what regards, contention-based protocols, ALOHA is distinguished by numerous collisions, partially mitigated in the CSMA protocol by the introduction of the sensing phase and a waiting period. This waiting period is denoted as the backoff (BO) time [7]. We refer to BO time as BO in the rest of the paper.

The topic of BO selection has been partially addressed in the literature, through mathematical methods [8], [9] and artificial intelligence (AI) algorithms [10]–[13]. On the one hand, the authors in [10] present a deep reinforcement learning (DRL) model in which an agent seeks to select the optimal BO for slotted ALOHA, and in [11] the same evaluation is performed for a CSMA protocol. On the other hand, the authors in [12] and [13] propose a Q-learning based CSMA/collision avoidance protocol that employs a BO selection scheme by adjusting the contention window (CW). Nevertheless, the prior works on BO selection neither consider the simple but powerful medium access protocol ALOHA nor do they apply these evaluations to industrial scenarios or THz frequencies. Furthermore, these works consider only a limited number of nodes in the network, implying a low level of flexibility. Finally, the impact of the BO selection and its subsequent effect on latency is not addressed. In this paper, we address the aforementioned shortcomings in related work on BO selection. We consider a shop floor where an autonomous guided vehicle (AGV) is used for transporting products between the production line and the warehouse. It uses an ALOHA-like protocol for medium access at THz frequencies to send data to the base station (BS) to guarantee a satisfactory production flow. The network is highly contested, with numerous sensors and workers transmitting data to the BS for production and logistic tasks. Therefore, the AGV experiences an extremely congested network, where it has to meet the industrial processes' requirements. The AGV has to select an optimal BO to ensure a high success probability and a low latency. In this highly dynamic scenario where the AGV moves while the sensors produce dynamic traffic, it is impossible to rely on traditional optimization algorithms. Therefore we introduce an adaptive model that relies on AI.

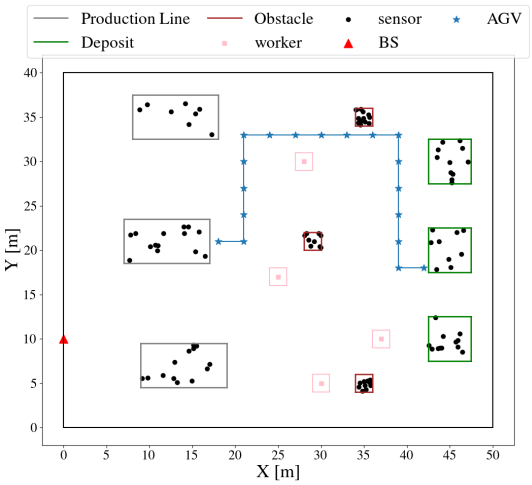

Fig. 1. Industrial plant comprising production lines, obstacles, workers, deposits, and an AGV.

The main contributions of our work are as follows:

- We develop a novel RL-based BO model implemented on the AGV which is able to learn the network traffic,

and optimize the medium access by selecting the appropriate BO to wait before accessing the channel using an ALOHA-like protocol. The goal is to ensure the correct reception of data at the BS, in a highly congested network, without having any prior information on the number of active nodes competing for the channel or the industrial environment.
- We extensively evaluate the proposed algorithm exploiting the 3DScat ray tracing (RT) tool to provide an accurate channel model valid also at THz frequencies.
- Furthermore, we compare its performance with state-of-the-art (SOTA) solutions, like random BO selection and binary exponential backoff (BEB), as well as an ideal benchmark.

The obtained results prove that the implemented model is able to learn the traffic dynamics and select an optimal BO to access the network. Specifically, the RL model outperforms the SOTA benchmarks and obtains performance comparable to the ideal ones. The paper is organized as follows; the system model and the RL model are presented in Secs. II and III. Sec. IV illustrates the simulation results, while Sec. V concludes the paper.

II. System Model

A. Scenario

We assume a shop floor F comprising of P production lines, D deposits, and O obstacles. The walls of the shop floor are made of concrete, whereas the ceiling and the floor are made of SOTA materials.

The production lines consisting of a variety of machines for assembly, packaging, and measurements, are manufactured from metal. The obstacles, which include tables and shelves, are made of wood, while the deposits are constructed of plywood. Furthermore, we assume the presence of W workers, an AGV, and S sensors on the shop floor, as illustrated in Fig. 1. The sensor nodes are distributed uniformly within the shop floor and the production lines. The sensor nodes distributed uniformly within the whole shop floor are called uniform nodes (UN), whereas the sensor nodes distributed only within the production lines are called machine nodes (MN). We assume a BS on the shop floor. The sensor nodes, the workers, and the AGV need to communicate their data to the BS for the correct functioning of the production process. Hence, all entities (including sensors, workers, and the AGV) that require access to the BS, are classified as user equipment (UE).

B. Ray Tracing Tool

To simulate signal propagation within the shop floor we used the 3DScat ray tracing (RT) tool. We rely on RT simulation rather than exploiting the 3GPP channel model [14] in order to capture the specific environment at THz frequencies. The RT tool is described with more detail in [15], [16]. It requires a detailed description of the shop floor layout, including building material, obstacles, the transmitter (TX), and the receiver (RX) nodes. The 3DScat employs a geometric approach to simulate signal propagation, applying ray theory to derive the received power P_{rx} for each link. Electromagnetic parameters of the materials in the shop floor are described using the relative dielectric constant ϵ_r and the electrical conductivity σ_r as listed in Table I, and derived from [17]. RT simulation involves a total of 5 interactions, comprising 3 reflections, 1 diffraction, 1 reflection with scattering, and 4 reflections with diffraction. This interaction setting was found to be a suitable compromise between accurate channel prediction and computational expenditure, since performance saturates when at least 3 reflections are considered [15]. The 3DScat operates in two phases, firstly the trajectories of the optical rays between TX and RX are calculated applying an image-RT approach according to the geometrical optics theory and its extensions, such as the uniform theory of diffraction and diffuse scattering models. Subsequently, field propagation and multipath effects are evaluated.

TABLE I
Ray Tracing Materials Characterization

Material	ϵ_r	σ_r
Concrete	5.24	4
Ceiling	1.52	1.03
Floor	2.73	1.80
Metal	0	10^7
Plywood	2.71	0.33
Wood	1.994	2.12

C. Channel Model

Signal propagation in the shop floor is modeled using a narrow-band channel model, under far field assumptions. The signal-to-noise ratio SNR in dB is formulated as $SNR = P_{tx} + G_{tx} + G_{rx} - PL_s - P_n$, where P_{tx} represents the transmitted power, G_{tx} and G_{rx} are the TX and RX gains, respectively. PL_s stands for the path loss, with s representing the line-of-sight (LoS)/non-LoS (NLoS) condition. Lastly, P_n defines the noise power. Specifically, the path loss is calculated by adopting the close-in free space reference distance path model, since it has proven to be superior for modeling path loss over many environments and frequencies. It is formulated as: $PL_s = c + 10\alpha_s \log_{10}(d) + \xi_s$, where $c = 20\log_{10}\left(\frac{4\pi}{\lambda}\right)$ is the path loss at a reference distance $d_{ref} = 1m$. α_s is the path loss exponent. d represents the 3D distance between TX and RX. Lastly, ξ_s is used to account for shadow fading. ξ_s is modeled as a zero mean Gaussian distribution $\xi_s \sim \mathcal{N}(0, \sigma_s^2)$ with standard deviation $\sigma_s = \sqrt{\frac{\sum_{i=1}^{N}(x_i-\mu)^2}{N}}$, where N is the total number of values, x_i represents the i-th value, and μ is the expected value. The large-scale channel parameters α_s and σ_s are obtained from the RT simulator, as illustrated in Table II for uplink (UL) and downlink (DL) communications considering UN and MN sensors' distribution. The noise power is expressed as $P_n = kT_{eq}B$ where k represents the Boltzmann constant, T_{eq} is the system temperature, and B is the bandwidth. This channel model is also applicable at high frequencies because it has been demonstrated that THz frequency effects, such as molecular absorption, have a limited impact on the channel in a such moderate-sized scenario [18].

D. Traffic

It is assumed that all UEs are synchronized in time. The UEs follow an ALOHA-like medium access protocol. They are initiated in an idle state. Considering a real-time system, the workers and the sensors transmit their data to the BS as

TABLE II
LARGE SCALE CHANNEL PARAMETERS

Parameter	UL			DL		
	UN-BS	MN-BS	AGV-BS	BS-UN	BS-MN	BS-AGV
α_{LoS}	1.94	2.08	1.96	1.92	2.02	1.96
α_{NLoS}	3.35	3.75	3.34	3.47	3.59	3.43
σ_{LoS}	5.68	5.61	6.06	5.39	6.1	5.99
σ_{NLoS}	11.65	14.28	12.98	13.04	11.47	15.23

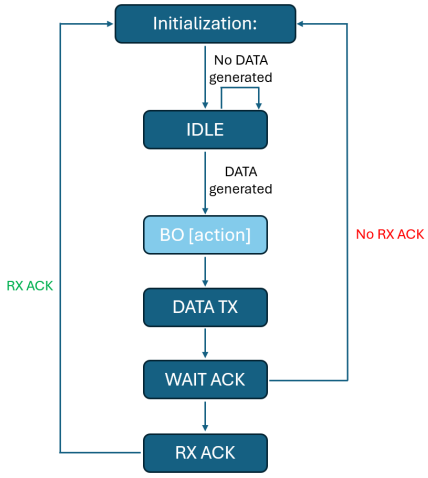

Fig. 2. Flow diagram of the ALOHA-like MAC layer protocol for the AGV.

soon as it is generated. The workers generate periodic data with a periodicity T_w. The sensors generate sensing data by dynamically changing their data generation periodicity following a two-state Markov chain. The two states are designated as fast and slow with data generation periodicity T_f and T_s, respectively. At time $t' = 0$, the sensor operates in a fast mode with probability p_{in} and in a slow mode with probability $1 - p_{in}$. A sensor in fast mode at time t' remains in the fast mode with probability p or changes to a slow mode with probability $1 - p$ at the successive time $t' + 1$. Similarly, a sensor in slow mode at time t' remains in the slow mode with probability q or changes to a fast mode with probability $1 - q$ at the successive time $t' + 1$. The AGV follows a similar medium access protocol as the workers and the sensors, with one exception. The AGV does not immediately transmit its data as soon as it is generated, rather it implements a waiting period, i.e. a BO before transmitting its data. The AGV generates data at regular intervals along its path within the shop floor. A transmission is considered successful if the received SNR at the BS is above a certain threshold SNR_{th} and no collisions occur. If the transmission is successful, an acknowledgment (ACK) is sent by the BS. Once a transmission is completed, the UEs wait for an ACK. Whether an ACK is received or not, the UEs transition to an idle state until new data is generated. In the case of unsuccessful transmissions, no retransmissions are performed. The AGV is considered an intelligent entity in the scenario. Hence, the duration of the BO is selected using a RL algorithm. This ALOHA-like MAC protocol for the AGV is depicted in Fig. 2.

III. REINFORCEMENT LEARNING MODEL

A. RL Model on the AGV

RL algorithms comprise of two principal components: the agent and the environment [19]. The agent is the entity that performs the actions, which in this case is the AGV. The environment, on the other hand, encompasses everything external to the agent. In the considered scenario, it includes the shop floor, in particular the sensors and the workers. The learning process starts with the agent in an initial state S_t, wherein it explores the surrounding environment. Then, it performs an action A_t that corresponds to the selection of the BO before transmitting the data. Subsequently, the action results in the agent being moved to a new state S_{t+1}. While the environment offers to the agent a reward R_t [20] accounting for the goodness of the action on the system. The time is divided in $t = 1, 2, ..., T_{ep}$ discrete intervals which compose an episode. Each interval t comprises a set of ticks $t' = 1, 2, ..., T_{tick}$ and has a duration T_S. In an episode, the AGV moves from the production line to the warehouse following a defined path. At each interval, the AGV moves and generates data, while sensors and workers transmit based on their traffic model as described in Sec. II-D. The learning process is designed using a Markov decision process (MDP), which is described by the vector (S_t, A_t, T_t, R_t).

- $S_t \in \mathcal{S}$ defines the space of all possible states for an agent. In particular, $S_t = \{x_t, y_t, t, g_i\}$ contains the coordinates of the AGV at time t (x_t, y_t) obtained considering a right-handed reference system, the time instant t, with $t = 1, 2, ..., T_{ep}$, and the produced good g_i, with $g_i = 1, 2, ..., g_{max}$, that has been manufactured by the production line $P_i \in P$ and needs to be transported to the correct deposit $D_j \in D$. Indeed, each good requires to be transported to a specific deposit. The location of the AGV at time t is determined as $(x_t, y_t) = (x_{t-1}, y_{t-1}) + (\Delta x_{t-1}, \Delta y_{t-1})$. Here (x_{t-1}, y_{t-1}) represents the previous location of the AGV at time $t - 1$. $(\Delta x_{t-1}, \Delta y_{t-1})$ is the change in it's location and it is calculated as the product between the velocity vector and the interval of time $\Delta x_{t-1} = \vec{v}_{t-1} \cdot \Delta t$.
- $A_t \in \mathcal{A}$ represents the set of actions that the AGV can perform. It consists of a discrete set of BOs $A_t = \{1, 2, ..., T_{BO,max}\}$ expressed in tick, with duration t'.
- T_t is the transition probability function. It models the dynamics of the environment by returning the probability of transitioning to a new state S_{t+1}, given an action A_t and an initial state S_t.
- $R_t \in R$ is the reward and it is a fundamental parameter for the learning process. In particular, -1 is assigned in case the message sent by the AGV to the BS collides, and

$-\frac{\tau}{\tau_{tot}}$ otherwise. The reward was designed ad-hoc with the aim of minimizing the selected BO and therefore the medium access delay and latency while maximizing the success probability.

$$R_t = \begin{cases} -1 & \text{if AGV message collides} \\ -\frac{\tau}{\tau_{tot}} & \text{otherwise} \end{cases} \quad (1)$$

with $\tau = \tau_{BO} + \tau_{tr,DATA} + 2 \cdot \tau_{prop} + \tau_{proc} + \tau_{tr,ACK}$ and $\tau_{tot} = \tau_{BO,MAX} + \tau_{tr,DATA,MAX} + 2 \cdot \tau_{prop,MAX} + \tau_{proc} + \tau_{tr,ACK,MAX}$. Specifically, τ_{BO} represents the BO selected by the RL agent, $\tau_{tr,DATA} = \frac{P_{DATA}}{R_b}$ is the time needed to transmit a message of size P_{DATA} with a data rate R_b, τ_{prop} defines the propagation delay which depends on the relative distance between BS and AGV, τ_{proc} represents the processing time at the BS, and $\tau_{tr,ACK}$ stands for the time needed to transmit an ACK of size P_{ACK} with a data rate R_b. As for the normalization parameters, $\tau_{BO,MAX}$ represents the maximum BO which can be selected by the RL agent, $\tau_{tr,DATA,MAX}$ is the time needed to transmit a message of maximum size $P_{DATA,MAX}$ with a data rate R_b, $\tau_{prop,MAX}$ defines the propagation delay referred to the farthest location with respect to the BS, and $\tau_{tr,ACK,MAX}$ stands for the time needed to transmit an ACK message with data rate R_b. The goal of the RL algorithm is to identify the optimal policy $\pi_\theta(S_t, A_t)$ which maximizes the cumulative discounted return $G_0 = \sum_{t=1}^{T_{ep}} \gamma R_t$, with $\gamma \in [0, 1)$ being the discount factor, a parameter which balances the importance of immediate and future rewards.

B. RL Algorithm

The RL algorithm exploited in this work is the advantage actor critic (A2C) [21] which is a DRL algorithm that uses a synchronous gradient descent to optimize deep neural networks. This algorithm's network comprises two sub-networks: the actor and the critic. This architecture is helpful in reducing the variance of the RL algorithm by stabilizing and fastening the training process. Hence, the simultaneous operation of the actor and critic facilitates the exploration of diverse regions of the environment through the exploitation of policy-based and value-based methods. Specifically, the training process evolves as follows; at each time step t the RL agent comprising the actor and the critic observes the current state S_t, then the actor performs an action A_t according to its policy $\pi_\theta(S_t, A_t)$ which is the mapping from the state S_t to the action A_t. The action is used by the critic to compute the value of taking that action at that state, also known as the V-function, $V(S_t)$. Therefore, the critic observes the action and provides feedback, while the actor updates its policy parameters using the advantage function. The latter is preferred with respect to the action value function to enhance the stability of the learning process. The advantage function calculates the advantage obtained in taking that action in that state with respect to the average value of the state. It is obtained by subtracting the mean value of the state from the state action pair: $A(S_t, A_t) = Q(S_t, A_t) - V(S_t) = G_0 + \gamma V(S_{t+1}) - V(S_t)$. Therefore, the parameters of the actor are updated as follows: $\Delta\theta = \eta \nabla_\theta (\log \pi_\theta(S_t, A_t)) A(S_t, A_t)$, with η being the learning rate. While those of the critic are updated exploiting the following formulation $\Delta\phi = \eta A(S_t, A_t) \nabla_\phi V(S_t)$. Thus, in the actor-critic there are two function approximations and therefore two neural networks: a policy $\pi_\theta(S_t, A_t)$ that controls the agent actions and a value function $V(S_t)$ used to assist the policy updates by measuring the goodness of the action performed.

IV. Simulation Results

This section describes the results obtained by exploiting the A2C algorithm as RL model to predict the traffic dynamics, selecting an appropriate BO, and guaranteeing the accurate reception of data by the BS. Table III lists the parameters utilized to simulate the channel model, the MAC layer, and the RL algorithm in the considered scenario. Specifically, the implemented RL-based BO model was compared to a series of benchmarks:

(i) First, an ideal benchmark (termed as oracle) that assumes that the AGV always chooses the optimal BO;
(ii) Second, a random BO model (named as random) where the BO is chosen randomly;
(iii) Third, the BEB model [22], which is characterized by a variable CW depending on the collisions experienced by the AGV. In detail, the AGV selects a random BO value within the interval $[1, 2^i C]$, where $i = 2, 3, 4$ determines the maximum number of acceptable collisions and C defines the size of the CW. To perform a fair comparison, it is necessary to ensure that $2^i C$ is equal to the maximum BO $T_{BO,max}$ that can be selected by the RL-based BO model.

Fig. 3 illustrates the evolution of the cumulative reward R as a function of the number of episodes N_{ep} for the proposed RL-based BO model and the different benchmarks for the UN and MN sensor distributions. Here, for the BEB models the maximum number of acceptable collisions is equal to $i = 2$ (BEB2), $i = 3$ (BEB3), and $i = 4$ (BEB4). The cumulative reward is calculated as $R = \sum_{t=1}^{T_{ep}} R_t$, by summing the individual rewards R_t obtained at each time step t of an episode with duration T_{ep}, as given in Eq. (1). It can be seen that the reward achieved by the RL-based BO model is only slightly lower as compared to that for the oracle (ideal benchmark), with a difference of $\Delta_R^{RL,oracle} = 2.39\%$. As a result, the model can learn the BO by adapting to the changes in the scenario. Moreover, after converging the RL-based BO model is able to achieve a significantly enhanced performance as compared to the non-ideal benchmarks. Their behavior remains constant over the number of simulated episodes. In particular, the RL model is characterized by an improvement of $\Delta_R^{RL,random} = 76.74\%$ with respect to the random BO model, $\Delta_R^{RL,BEB2} = 67.01\%$ with respect to the BEB2 model, $\Delta_R^{RL,BEB3} = 40.1\%$ with respect to the BEB3 model, and $\Delta_R^{RL,BEB4} = 82.82\%$ with respect to the BEB4 model. It can be observed that the BEB4 model exhibits the poorest performance. The reason is that the BEB4 begins with a CW significantly smaller than the other BEB models, resulting in a shorter BO and consequently a higher collision probability. The random BO model performs slightly better than the BEB4 model. Whereas the BEB2 and BEB3 models perform better than the random BO model and the BEB4 model. This result is quite intuitive, as an adaptive model, if not characterized by a low CW, leads to better results than a random BO selection.

Additionally, one can observe a higher reward for the non-ideal BO models for the MN sensor distribution when compared to the UN sensor distribution. This may be attributable to a higher success probability resulting from the particular distribution of sensors in clusters within the production lines. However, this difference is not observed for the oracle, which exhibits the same performance for both sensor distributions. The same applies to the RL-based BO model. This means that the RL-based BO model can choose the optimal BO and is robust to different sensors' distribution.

Fig. 4 depicts the variation of the success probability p_{suc} against the number of episodes for the different BO models. The success probability is defined as the ratio between the number of correctly acknowledged transmissions and the total attempts $p_{suc} = \frac{N_{suc}}{N_{tx}}$. It can be seen that the RL-based BO model highly differs from the non-ideal benchmarks, leading to an improvement of $\Delta_{p_{suc}}^{RL,random} = 31.03\%$ with respect to the random BO selection, $\Delta_{p_{suc}}^{RL,BEB2} = 27.50\%$ with respect to the BEB2 model, $\Delta_{p_{suc}}^{RL,BEB3} = 17.09\%$ with respect to the BEB3 model, and $\Delta_{p_{suc}}^{RL,BEB4} = 34.6\%$ with respect to the BEB4 model. The RL-based BO model achieves a performance highly comparable to that of the oracle, with a difference of just $\Delta_{p_{suc}}^{RL,oracle} = 0.51\%$. Particularly, the RL model is able to guarantee a minimum success probability of $p_{suc} > 0.9946$.

Fig. 5 shows the mean selected BO in an episode along an entire simulation for the algorithms described above. The mean selected BO is formulated as $\overline{BO} = \frac{\sum_{t=1}^{T_{ep}} BO_t}{T_{ep}}$ and measured in ticks, where BO_t is the BO chosen at time instant $t \in T_{ep}$. As anticipated, the random BO model demonstrated an average value of selected BO equal to half ($\overline{BO^{random}} = 250$ ticks) of the maximum BO ($T_{BO,max} = 500$ ticks). In contrast, the BEB2, BEB3, and BEB4 models exhibited a lower selection of BO which is $\overline{BO^{BEB2}} = 166.39$ ticks, $\overline{BO^{BEB3}} = 86.11$ ticks, and $\overline{BO^{BEB4}} = 54.77$ ticks, respectively, and that clearly depends on the CW. It is worth mentioning that although BEB3 and BEB4 models offer the possibility of selecting a shorter BO with respect to the proposed RL-based BO model, they are not able to provide adequate performance. Particularly, they offer a lower success probability equal to $p_{suc}^{BEB3} = 0.82$ and $p_{suc}^{BEB4} = 0.64$, which is insufficient for industrial applications that impose extremely high reliability. The proposed RL-based BO model is instead able to provide adequate success probability with a limited mean selected BO of $\overline{BO^{RL}} = 90.1$ ticks.

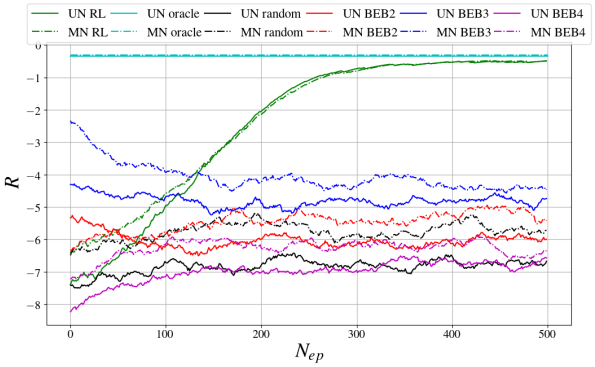

Fig. 3. Comparison of the reward evolution with the number of episodes considering the implemented RL algorithm for intelligent BO selection, oracle, random BO, BEB2, BEB3, and BEB4 models.

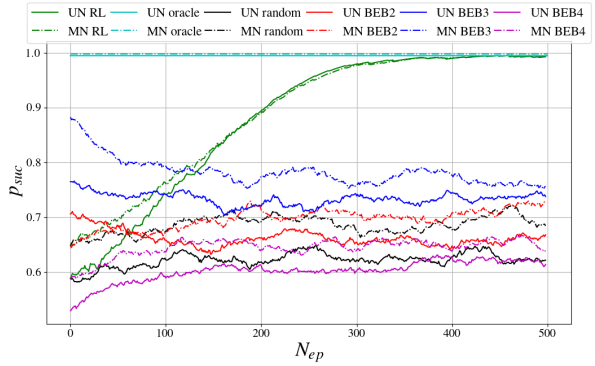

Fig. 4. Comparison of the evolution of the success probability with the number of episodes considering the implemented RL algorithm for intelligent BO selection, oracle, random BO, BEB2, BEB3, and BEB4 models.

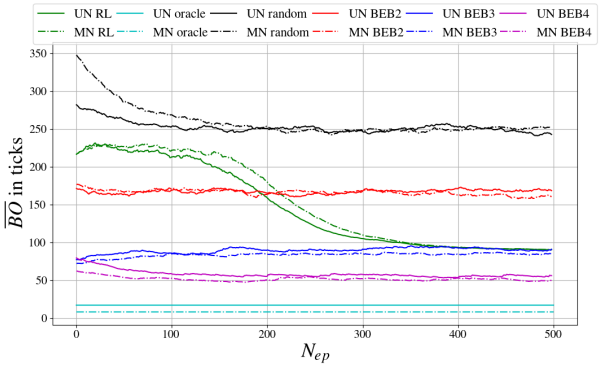

Fig. 5. Comparison of the evolution of the mean BO chosen with the number of episodes considering the implemented RL algorithm for intelligent BO selection, oracle, random BO, BEB2, BEB3, and BEB4 models.

V. Conclusions

This paper implements an innovative RL-based ALOHA-like MAC protocol with BO selection. The main goal is to optimize the network access to guarantee a continuous and satisfactory production flow in the shop floor of Industry 5.0. The channel model is simulated using an RT tool to accurately model the signal propagation at THz frequencies. An ALOHA-like MAC protocol with a BO before transmission is considered for an AGV in the shop floor. The RL algorithm learns the network dynamics and selects the BO for the AGV to access the network, without any prior information about the industrial layout. The RL algorithm implemented on the AGV is the A2C. The developed model is compared against an ideal benchmark, a random BO selection, and a BEB model accepting a different number of collisions. It can be seen in the results, that the implemented algorithm highly outperforms the non-ideal benchmarks, demonstrating an improvement in the success probability up to 34.6% and guaranteeing a minimum reward enhancement of 40.1%. Furthermore, the RL model obtains exceptional performance, highly comparable with those of the oracle, attaining a difference in terms of reward and success probability of 2.39% and 0.51%, respectively, while guaranteeing a minimum success probability of 99.46%.

VI. Acknowledgement

This work has been performed as part of the HORIZON-JU-SNS-2022 project TIMES, co-funded by the European Union,

TABLE III
SIMULATION PARAMETERS

Layer	Parameter	Notation	Value
Scenario	Shop Floor Size	F_xxF_yxF_z	50x40x10 m^3
	Production Line Size	P_xxP_yxP_z	10x5x3 m^3
	Deposit Size	D_xxD_yxD_z	5x5x3 m^3
	Obstacle Size	O_xxO_yxO_z	2x2x2 m^3
	AGV Size	V_xxV_yxV_z	1x1x1 m^3
	Number of Sensors	S	100
	Number of Workers	W	4
Channel Model	BS Transmitted Power	$P_{tx,BS}$	30 dBm
	AGV Transmitted Power	$P_{tx,AGV}$	15 dBm
	Sensors Transmitted Power	$P_{tx,sens}$	0 dBm
	Workers Transmitted Power	$P_{tx,work}$	0 dBm
	BS Transmission and Reception Gains	$G_{tx,BS} = G_{rx,BS}$	10 dB
	AGV Transmission and Reception Gains	$G_{tx,AGV} = G_{rx,AGV}$	5 dB
	Sensors Transmission and Reception Gains	$G_{tx,sens} = G_{rx,sens}$	0 dB
	Workers Transmission and Reception Gains	$G_{tx,work} = G_{rx,work}$	0 dB
	Carrier Frequency	f_c	300 GHz
	Bit Rate	R_b	50 Gbit/s
	SNR of Threshold	SNR_{th}	-70 dBm
MAC	ACK Size	P_{ACK}	10 B
	Data Size	P_{DATA}	20 B
	BEB2 CW Size	C_{BEB2}	125
	BEB3 CW Size	C_{BEB3}	62
	BEB4 CW Size	C_{BEB4}	31
	Simulation Time	T_S	0.00005 s
	Workers Transmission Periodicity	T_w	$3.94 \cdot 10^{-7}$ s
	Slow Modality Transmission Periodicity	T_s	$7.04 \cdot 10^{-7}$ s
	Fast Modality Transmission Periodicity	T_f	$3.82 \cdot 10^{-7}$ s
	Fast and Slow Modality Probabilities	p, q	0.5
	Initial Nodes Probability to be in Fast Mode	p_{in}	0.5
	Maximum BO Duration in Ticks	$T_{BO,max}$	500 ticks
	Tick Duration	t'	$1.6 \cdot 10^{-9}$ s
RL	Neural Network	NN	32x32
	Learning Rate	η	0.001
	Discount Factor	γ	0.5
	Episode Duration	T_{ep}	17
	Number of Episodes	N_{ep}	500

within the DETERMINISTIC6G project under grant agreement No. 101096504, and under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART"). We express gratitude to Franco Fuschini and Andrea Panebianco.

REFERENCES

[1] 5G-ACIA, "5G for Connected Industries and Automation," *White Paper*, vol. 2nd Edition, 2019.

[2] D. C. Nguyen et al., "6G Internet of Things: A Comprehensive Survey," *IEEE Internet of Things J.*, vol. 9, no. 1, pp. 359–383, 2022.

[3] C. Chaccour et al., "Seven Defining Features of Terahertz (THz) Wireless Systems: A Fellowship of Communication and Sensing," *IEEE Commun. Surv. & Tut.*, vol. 24, no. 2, pp. 967–993, 2022.

[4] C. Buratti, L. Mesini, and R. Verdone, "Comparing MAC Protocols for Industrial IoT Using Terahertz Communications," in *Proc. Int. Symp. on Personal, Indoor and Mobile Radio Commun. (PIMRC)*, 2020, pp. 1–7.

[5] S. Cavallero et al., "Applying Carrier Sense Multiple Access to Industrial IoT at Terahertz Frequencies," *IEEE Internet of Things J.*, vol. 11, no. 7, pp. 11 986–11 999, 2024.

[6] M. Yu et al., "Joint Scheduling and Power Allocation for 6G Terahertz Mesh Networks," in *Proc. Int. Conf. on Comput., Netw. and Commun. (ICNC)*, 2020, pp. 631–635.

[7] J. Hastad, T. Leighton, and B. Rogoff, "Analysis of backoff protocols for multiple access channels," in *Proc. of the Nineteenth Annu. ACM Symp. on Theory of Comput.* New York, NY, USA: ACM, 1987, p. 241–253.

[8] Z. Haas and J. Deng, "On optimizing the backoff interval for random access schemes," *IEEE Trans. on Commun.*, vol. 51, no. 12, pp. 2081–2090, 2003.

[9] S. Misra and M. Khatua, "Semi-Distributed Backoff: Collision-Aware Migration from Random to Deterministic Backoff," *IEEE Trans. on Mobile Comput.*, vol. 14, no. 5, pp. 1071–1084, 2015.

[10] M. A. Jadoon et al., "Deep Reinforcement Learning for Random Access in Machine-Type Communication," in *Proc. IEEE Wireless Commun. and Netw. Conf. (WCNC)*, 2022, pp. 2553–2558.

[11] T. Lee and O. Jo, "Learning Backoff: Deep Reinforcement Learning-Based Wireless Channel Access," *IEEE Syst. J.*, vol. 18, no. 1, pp. 351–354, 2024.

[12] Z. Zheng et al., "An adaptive backoff selection scheme based on Q-learning for CSMA/CA," *Wirel. Netw.*, vol. 29, no. 4, p. 1899–1909, 2023.

[13] M. Mohammed, Z. Aliouat, and Y. Slimani, "Intelligent CW Selection Mechanism Based on Q-Learning (MISQ)," *Ingénierie des Systèmes D Information*, vol. 25, pp. 803–811, 2020.

[14] 3GPP, "Study on channel model for frequencies from 0.5 to 100 GHz (Release 16)," *TS 38.901*, 2019.

[15] E. M. Vitucci et al., "Ray Tracing RF Field Prediction: An Unforgiving Validation," *Int. J. of Antennas and Propag.*, vol. 2015, pp. 1–11, 2015.

[16] F. Fuschini et al., "Ray tracing propagation modeling for future small-cell and indoor applications: A review of current techniques," *Radio Sci.*, vol. 50, no. 6, pp. 469–485, 2015.

[17] Recommendation ITU-R, "Effects of building materials and structures on radiowave propagation above about 100 MHz," *P.2040-2*, 09/2021.

[18] A. Tarozzi and R. Verdone, "ML-Based Channel Parameters Estimation For Sensing Applications in Industrial IoT Scenarios," in *2024 IEEE 100th Veh. Technol. Conf. (VTC2024-Fall)*, 2024, pp. 1–6.

[19] M. Morales, *Deep Reinforcement Learning*. Manning Publications Co., 2020.

[20] K. Arulkumaran et al., "A Brief Survey of Deep Reinforcement Learning," *IEEE Signal Process. Mag., Special Issue on Deep Learn. for image understanding*, 2017.

[21] V. Mnih et al., "Asynchronous Methods for Deep Reinforcement Learning," *CoRR*, vol. abs/1602.01783, 2016.

[22] B.-J. Kwak, N.-O. Song, and L. Miller, "Performance analysis of exponential backoff," *IEEE/ACM Trans. on Netw.*, vol. 13, no. 2, pp. 343–355, 2005.