QoS based Deep Reinforcement Learning for V2X Resource Allocation

Shubhangi Bhadauria*, Zohaib Shabbir†, Elke Roth-Mandutz*, Georg Fischer‡
*Fraunhofer IIS, Erlangen, Germany,
†Altran Technologies, Wolfsburg, Germany,
‡Friedrich Alexander University (FAU), Erlangen, Germany,
Email: [shubhangi.bhadauria, elke.roth-mandutz]@fraunhofer.iis.de, zohaib.shabbir@altran.com, georg.fischer@fau.de

***Abstract*—The 3rd generation partnership project (3GPP) standard has introduced vehicle to everything (V2X) communication in Long Term Evolution (LTE) to pave the way for future intelligent transport solutions. V2X communication envisions to support a diverse range of use cases for e.g. cooperative collision avoidance, infotainment with stringent quality of service (QoS) requirements. The QoS requirements range from ultra-reliable low latency to high data rates depending on the supported application. This paper presents a QoS aware decentralized resource allocation for V2X communication based on a deep reinforcement learning (DRL) framework. The proposed scheme incorporates the independent QoS parameter, i.e. priority associated to each V2X message, that reflects the latency required in both user equipment (UE) and the base station. The goal of the approach is to maximize the throughput of all vehicle to infrastructure (V2I) links while meeting the latency constraints of vehicle to vehicle (V2V) links associated to the respective priority. A performance evaluation of the algorithm is conducted based on system level simulations for both urban and highway scenarios. The results show that incorporating the QoS parameter (i.e. priority) pertaining to the type of service supported is crucial in order to meet the latency requirements of the mission critical V2X applications.**

Keywords*—V2X communication, QoS, DRL, resource allocation.

I. Introduction

Future intelligent transport systems (ITS) are evolving towards connected, cooperative and automated driving. V2X communication is an enabler to a large number of connectivity-related use cases encompassing the areas of safety, traffic efficiency, convenience, infotainment and autonomous driving. The use of cellular infrastructure for V2X was initiated by the introduction of direct communication between devices, the device-to-device (D2D) communication by 3GPP in release 12 [1]. In release 14, 3GPP introduced the LTE support for V2X communication [2], which is already capable of supporting use cases in the areas of active safety, traffic efficiency and infotainment. As 3GPP is evolving towards 5G in release 16, enhanced use cases including platooning, advanced driving, remote driving and extended sensors are supported. All the mentioned use cases need V2X messages to be exchanged via both cellular infrastructure on the Uu interface as well as using direct communication on the sidelink via the PC5 interface, e.g. V2V communication. The requirements in terms of reliability, latency and data are most stringent for these use cases. For example, advanced driving requires an end to end latency of 3ms and reliability of 99.999% [3].

Efficient radio resource management (RRM) is essential to fulfilling the QoS requirements for the V2X sidelink communication. V2X sidelink communication operates in in-coverage, partial coverage and out of coverage scenarios. When the vehicles are in in-coverage, resources for data transmission are typically scheduled by the base station, which is referred to as mode 3 in LTE [2]. While, when the vehicle is in out of coverage then UE decides on its resource allocation autonomously based on preceding sensing results. Autonomous resource allocation utilizes the unlicensed resource pools for data transmission. The dynamic nature of the vehicular environment such as fast varying wireless propagation channels and rapidly varying network topology makes it even more challenging to maintain the ultra-reliable low latency communication needed for mission critical use cases. These challenging conditions often result in sub-optimal solutions for radio resource management (RRM) which cannot meet the required QoS.

One way to achieve efficient RRM solutions and spectrum access is to use advanced approaches utilizing advancements in Artificial Intelligence (AI). With the hardware advancements and availability of computational power it is possible to apply Machine Learning (ML) in wireless communication. Reinforcement Learning (RL) is a category of ML in which an agent performs sequential decision over time. The foundation of RL problems is based on a Markov decision process (MDP). Hence, the problems related to RL are composed of an agent or of agents and an environment. The learning agent interacts with the environment and in turn gets a scalar reward based on which an optimal policy is developed. As the V2X environment is highly dimensional, a DRL can be utilized to form adaptive RRM.

II. Related Work

There exist several contributions on RRM for V2X communication. In [4], a resource allocation mechanism is proposed for D2D communications in cellular networks in which the overall network capacity is maximized by adjusting the transmit power level of the transmission and assigning the optimized spectrum. In [5], a centralized resource allocation procedure is used. Each vehicle has to report channel state information (CSI) and interference information to the base station. Based on the received information from the vehicles, the resource allocation procedure is formulated as an optimization problem in which QoS requirements are used as constraints. With increasing number of vehicles, transmission overhead increases rapidly, resulting in decreased throughput. In [6], a decentralized resource allocation approach based on DRL for V2V communications is proposed, in which an agent corresponding to the V2V link is interacting with the environment and learning to make an optimal decision in order to find the optimum spectrum and power level for transmission. However, the proposed solution did not consider the 3GPP defined priority and latency indicator ProSe per packet priority (PPPP) of each vehicular UE (V-UE). In this paper, we have proposed a DRL

978-1-7281-7127-2/20/$31.00 ©2020 IEEE

based resource allocation procedure which takes into account the PPPP of each V2X message, in order to ensure that users with high priority safety messages are given priority and transmitted within the latency constraint. The state space, action space and reward function are formulated by taking into account the associated priority where the penalty for each V-UE is proportional to the QoS requirement. We have verified our proposed solution for both urban and highway scenario based on our simulation.

III. System Model

The vehicular network consists of $\mathbf{I} := [1, 2, 3...., I]$ V2I links and $\mathbf{V} := [1, 2, 3...., V]$ pairs of V2V links, as shown in Figure 1. The performance of V2X communication is analyzed for both urban and highway scenario based on system model defined in 3GPP Technical Report (TR) 36.885 [6]. In both scenarios, a V-UE is communicating to other V-UE's via PC5 interface and to the base station (eNB) via the Uu interface. Similar to [2], for efficient spectrum utilization it is assumed that orthogonal uplink resources for I V2I links are reused by V V2V links. The interference to V2I links consists of signal from V2V links and background noise. Thus, received signal-to-interference-plus-noise-ratio (SINR) for the i^{th} V2I link is given as :

$$SINR(i) = \frac{P_i.h_i}{N + \sum_{v \in \mathbf{V}} \rho_{i,v}.P_v.h_{v'}}; \quad (1)$$

P_i and h_i denote the transmission power of the i^{th} V2I transmitter and channel power gain from the i^{th} V2I transmitter to the eNB respectively. N represents the noise. The transmission power of the v^{th} V2V transmitter is represented by P_v. The interference channel power gain between the v^{th} V2V transmitter and the eNB is represented as $h_{v'}$.The resource occupancy is indicated by $\rho_{i,v}$. If v^{th} V2V transmitter reuses the spectrum of the i^{th} V2I link then $\rho_{i,v}$ is equal to 1 otherwise it will be 0 and vice versa. Hence, the capacity of i^{th} V2I link is given as:

$$C(i) = Blog_2[1 + SINR(i)](bits/second); \quad (2)$$

where B is the bandwidth.

Similarly, received SINR of v^{th} V2V UE can be expressed as:

$$SINR(v) = \frac{P_v.h_v}{N + \sum_{v \in \mathbf{V}} \rho_{v,i}.P_v.h_{v'} + H_v}; \quad (3)$$

P_v and h_v is the transmission power and channel gain of v^{th} V2V user respectively. H_v is the interference power of V2I link and overall interference power from all V2V links that are sharing the same resource block (RB). The resource occupancy is denoted by $\rho_{v,i}$. It is equal to 1 if v^{th} V2V transmitter reuses the spectrum of the i^{th} V2I link else vice versa it is 0. The capacity of v^{th} V2V link is given as:

$$C(v) = Blog_2[1 + SINR(v)](bits/second); \quad (4)$$

In this paper, stringent requirements for latency are considered for V2V links as defined by 3GPP [3] for LTE V2X. The latency is defined by the packet delay budget (PDB) of the application the V-UE needs to support. The PPPP is enabling QoS differentiation across various traffic streams corresponding to different sidelink logical channels. It consists of eight values, representing the priority of the associated traffic over the sidelink channel. The eight values are denoted by PPPP1 up to PPPP8, where PPPP1 stands for the highest priority. These constraints are explicitly considered in the reward function of DRL. When these constraints are violated, it results in the negative reward depending on the priority level associated to the V-UE. The V2V links will be selecting resources relying on local observations as eNB has no information about the V2V link. It is assumed that V-UEs are capable of maintaining unicast links with other V-UE's in proximity, i.e. within a minimum communication range. Each V-UE link is an agent and other V-UEs are part of the environment whose behavior is uncontrolled. Therefore, actions of the V-UEs, i.e., sub-channel and transmission power selection, are considered part of the environment. In order to characterize the whole environment, V-UEs are updated asynchronously. At every time slot V-UEs, which are within the minimum communication range update their selected actions. For example, for a communication scenario of cooperative collision avoidance between V-UEs, the requirement of the minimum communciation range is limited to 350m as defined in [3]. Hence, the number of V-UEs that update their selected actions at every time slot are limited. Thereby, each agent can observe the effect of other agents on the environment with limited overhead because of the decentralized setting. It is also assumed that the highest priority packet choses the maximum transmission power, i.e. 23 dBm.

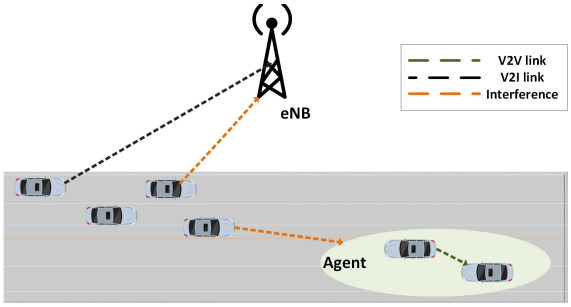

Figure 1. System model of V2X Communication in a highway scenario

IV. Resource Allocation based on Deep Reinforcement Learning

In this section, resource allocation for V-UEs is formulated based on DRL. The overall structural representation of the DRL approach is shown in Figure 2. As mentioned in the previous section, each V-UE link is an agent which interacts with the environment. The agent stores its experiences in the replay buffer based on experience replay. The mini-batch corresponds to the fixed sample size from the replay buffer which is used to train the neural network. Further details of the algorithm can be found in sub-section B. In Figure 2, at each time interval t, the agent observes a state s_t from the state space S where $s_t \in S$. The state of the environment includes the following parameters:

- channel state information (CSI) of corresponding V2V link(G_t),
- previous interference to V2V link (I_{t-1}),
- CSI of V2I link (H_t),
- sub-channel of neighbors in previous time slot (N_{t-1}),
- remaining load to transmit (L_t), and
- remaining time to fulfil the latency constraint (U_t) including the priority of previous transmissions (P_{t-1}).

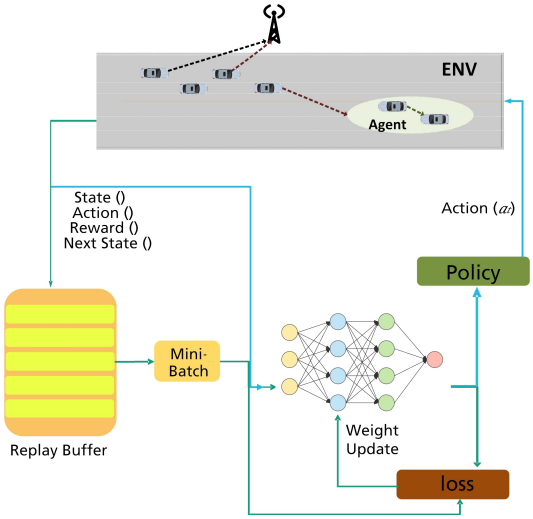

Figure 2. DRL approach: Agent and Environment Interaction

After observing the state, the agent takes an action a_t from the action space A where $a_t \in A$. The action a_t is comprised of selecting sub-channel and a power level for transmission based on the decision policy π. The V-UE selects the transmission power from three discretized levels of 5 dBm, 10 dBm and 23 dBm. Therefore, dimension of action space is $3 \times N_{RB}$.The policy π is the agent's behavior at each time interval and it maps state s_t to action a_t. The policy is purely deterministic and defined as $\pi : s_t \rightarrow a_t$. The agent aims to maximize the expected return, where the policy is $\pi(s, a) \in \prod$. Expected return is also called value function V that is represented as $V^{\pi}(s) : S \rightarrow \mathbb{R}$. Value function $V^{\pi}(s)$ can be represented as [7]:

$$V^{\pi}(s) = \mathbb{E}\left(\sum_{k=0}^{\infty} \gamma^{k} r_{t+k} | s_t = s, \pi\right); \tag{5}$$

where, γ is the discount factor such that $\gamma \in (0, 1)$. After the action taken by the agent, the environment transits to a new state denoted by s_{t+1} based on the policy. The state transition from s_t to s_{t+1} is stochastic and follows the MDP. The state transition which includes transition on the channels, interference, and remaining messages to transmit is generated by the 3GPP compliant V2X environment simulator.

The agent based on the action collects reward r_t from the environment. As shown in Figure 2, the reward is a scalar feedback signal and indicates the performance of agent in time interval t. The reward function includes capacities of V2V, V2I links and latency constraint including the associated packet priority. Mathematically, it can be represented as:

$$r_{t+1} = \lambda_a \sum_{i \in \mathbf{I}} C(i) + \lambda_b \sum_{v \in \mathbf{V}} C(v) - \lambda_p [T_o - U_t] \tag{6}$$

λ_a and λ_b are the positive weights of the reward equation. The weight of the latency condition λ_p is proportional to the PPPP associated to the packet. This implies, a higher penalty is imposed when a higher PPPP packet's latency constraint is violated. The term $[T_o - U_t]$ is the transmission time where T_o is tolerable latency for V2V links depending upon the type of application supported by the V2V UE. U_t is the remaining time to transmit the packet. Both, T_o and U_t are assumed to be provided by the application layer depending on the PDB of the supported service and is a V-UE implementation issue. Instead of eight, three priority levels are assumed here in the paper in order to accelerate the generation of simulation results. PPPP1 and PPPP2 are mapped to the high priority level where the associated service has maximum tolerable latency of 20ms. Similarly, PPPP3, PPPP4 and PPPP5 are mapped to medium priority where the associated service has maximum tolerable latency of 60ms. Finally, PPPP6, PPPP7 and PPPP8 are mapped as low priority where the associated service has maximum tolerable latency of 100ms. The previously mentioned latency requirements are taken from 3GPP TR 36.885. The main objective of RL is to maximize the cumulative discounted rewards as shown in the equation (7) and the extended form of equation is $r_{t+1} + \gamma.r_{t+2} + \gamma^2.r_{t+3}$...., where γ is the discount rate ranging between 0 and 1. The reward received at each state is represented as r_{t+k+1}.

$$G_t = \sum_{k=0}^{\infty} \gamma^{k}.r_{t+k+1}. \tag{7}$$

A. Designing of Deep Q-Network

The Q-learning is a value based learning method which falls under the category of model free RL algorithm. The Q-learning algorithm performs well when state and action space is small using a Q-table to capture the updated rules. When the state and action space becomes enormous then a Q-table cannot be used to capture the updated rules because many states may be visited on rare basis. Therefore, the corresponding Q-values will be updated on an infrequent basis, resulting in increased convergence time. One such complex system is V2X communication where the user density is high thereby leading to a high dimensional environment. This results in large state space and action space; therefore Deep Q-Network (DQN) needs to be introduced [8]. In DQN, the Q-learning table is replaced by a fully connected DNN that determines Q-values $Q(s, a, \theta)$, where s, a and θ are the state, action and weights respectively.

The state is applied as input of the DQN as shown in Figure 2. The weights of the neural network have been determined to get the output Q values $Q(s_t, a_t)$ of the DQN. The number of neurons in the hidden layers are kept to 500, 250, and 120 respectively. Initial evaluations were done by keeping the hidden layers to 3 as in [6]. Afterwards, for more stringent latency constraint of 20 ms the DQN was made denser and hidden layers were increased to 4 with 50 neurons in the respective layer. Rectified linear unit (ReLU) is used as an activation function to introduce non-linearity. The main objective to use an activation function is to help neural network to learn how to process complex data to make an accurate prediction.

The policy taken by the agent π has been determined by a Q function $Q(s_t, a_t, \theta)$ where θ is the parameter of Q function determined by DRL. Q-function has been determined to find an improved version of policy by taking an action such as $a_t = max_{a \in A} Q(s_t, a)$. The update rule of Q-learning can be used to get an optimal policy without the knowledge of system dynamics as shown in equation (8):

$$\begin{aligned} Q'(s_t, a_t) &= Q_o(s_t, a_t) + \alpha[r_{t+1} \\ &+ \gamma max_a Q_o(s, a_t) - Q_o(s_t, a_t)]; \end{aligned} \tag{8}$$

where,

- α and γ is the learning rate and discount factor,
- $r_{t+1} + \gamma max_a Q_o(s_t, a_t)$ is the target value and,
- $r_{t+1} + \gamma max_a Q_o(s, a_t) - Q_o(s_t, a_t)$ is the temporal difference error respectively.

B. Deep Q-Learning Algorithm

The Deep Q-Learning (DQL) algorithm has been split into two phases called training and testing phase similarly to other ML algorithms. The training and test data has been generated from the interaction with environment simulator and the agents. The environment simulator includes the V-UEs and the generated channels of the V-UEs including the CSI based on the V-UE dropping model as defined in 3GPP TR 36.885 [2]. Training and test data set contains $40,000$ samples. A batch size of 2000 samples has been considered which defines the number of samples that has been worked through before occurrence of an update to internal model parameters. The weights of DQN are updated after each batch of 2000 samples.

Each training sample used for the optimization of DQN contains the states (s_t), state transitions (s_{t+1}), actions (a_t) and rewards (r_t). At each iteration, DQN updates its randomly initialized weights (θ) in order to minimize the loss function. After forward propagation, an output value is generated called prediction value. To calculate the error, prediction value is compared to the actual output value. Loss function is used to calculate the error value and given as shown in the equation (9):

$$L(\theta) = \sum_{(s_t,a_t)\in D} [y - Q(s_t, a_t, \theta)]^2 \tag{9}$$

where D is the data set and s_t, a_t, θ are state, action at time interval t and weight, respectively. The expression for y is given in equation(10):

$$y = r_t + maximum_{a\in A} Q_{old}(s_t, a_t, \theta); \tag{10}$$

where r_t is the corresponding reward. The ADAM optimizer is used to minimize the loss function [9]. During the training process of the DQN, errors are minimized while weights are being updated. The choice of the learning rate follows the general ML trend of choosing hyperparameters of similarly solved problems. Hence, the learning rate of 0.01 is used motivated by the results in [6], which determines the update rate of the weights during the DQN training process. Another important aspect of DQN, called experience replay, is used to store the agent's experience in a replay buffer [8], as shown in Figure 2. The trajectories sampled from the environment are temporally correlated. Using temporally correlated trajectories in order to train the network can result in overfitting. The temporal correlations of data points during training will be broken down by storing the experience of the agent at each time interval t into a replay buffer. The agent's experience contains (s_t, a_t, r_t, s_{t+1}). At every instant, a mini-batch according to the batch size of 2000 samples is extracted from the replay buffer and is used to train the neural network to learn efficiently. The exploration and exploitation strategy used by the agent for collecting the experience is balanced by utilizing the ϵ-greedy policy. The policy is updated by choosing the action with maximum Q-value.

In the test stage, actions are selected, which yield the maximum Q-value given by the trained Q-network. Next, the environment simulator is updated by the selected actions. Finally, the evaluation results are generated, which provide the capacity of the V2I links as well as the probability of satisfied V2V links.

V. Simulation Results

In this section, the simulation results are presented, analyzed and evaluated to validate the proposed QoS-based DRL for V2X communication. We built our system level simulator based on the evaluation methodology for urban and highway case described in Annex A of 3GPP TR 36.885 [2]. V-UEs are dropped on the lane based on a spatial Poisson distribution. In this paper, the simulator was setup with 80 vehicles and a minimum communication range of the V-UE of 3 times the vehicle length. The number of lanes was set in each direction to two for the urban and to three for highway case. The channel model used is the V2V channel model and V2I channel model is taken from [2]. In our implementation a CPU based evaluation was performed thereby increasing the complexity and thus, the required time period for generating results. Similar implementation with GPU 1080 Ti can be limited to 2.4×10^{-4} seconds as per [6]. The simulation parameters are shown in the Table I.

Table I. Overall Simulation Parameters [2]

Parameter	**Value**
Bandwidth	10 MHz
Carrier frequency	2 GHz
No. of RBs	20
BS transmitter antenna height	25 m
UE drop and mobility model	Urban and Freeway case of A.1.2 [2]
Antenna pattern	Omni 2D
BS receiver antenna height	1.5 m
BS antenna gain	8 dBi
Vehicle transmitter antenna height	1.5 m
Vehicle receiver antenna height	1.5 m
Vehicle antenna gain	3 dBi
Vehicle receiver noise figure	9 dB
Vehicle speed in urban case	56 km/h
Vehicle speed in freeway case	70 km/h and 140 km/h
Latency constraints for V2V links	20 ms, 60 ms and 100 ms
List for V2V transmission power	$23, 10, 5$ dBm
Noise floor (N)	-114 dBm

A. Urban Scenario

For the urban scenario, firstly random resource allocation was evaluated for a stringent latency constraint of $20ms$ at 56 km/h. We evaluated the DQL first with 3 hidden layers. The obtained result was less optimum compared to the findings in [6], the reason being the larger latency budget of 100 ms considered in this paper. As a first improvement, we made the DQN denser by including one extra hidden layer with 50 neurons, resulting in a slightly improved performance of the algorithm. However, in a realistic scenario of V2X communication the required PDB by the UE's application will not be fixed to a single value. Hence, we analyzed as a next step the algorithm for a more realistic random scenario, where the PDB's vary for different applications. We evaluated the DQL by associating the reward with the respective PPPP. As shown in Figure 3, this leads to a significant improvement in the capacity of the V2I links by 7.034% comapred to DQL with 4 hidden layers. Also, compared to the random method the capacity of V2I links is improved by 26.16%

Similarly, in Figure 4, it is seen that by introducing the PPPP levels in the reward equation, an agent learned to satisfy the stringent latency constraint on V2V links while preserving enough resources. The DQL with priority levels has higher probability to meet the latency constraints compared to DQL with 4 hidden layers by 1.33%. Also, performance of DQL with PPPP as compared to the random method is improved by 17.73%.

B. Highway Scenario

Similar to the urban scenario, the throughput of the V2I links and the probability of V2V links satisfying the latency constraints are evaluated for the highway case at both low

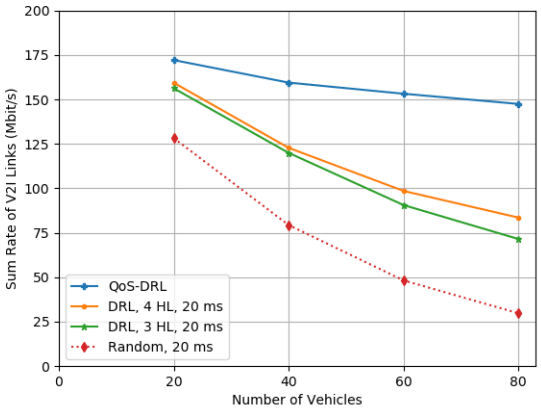

Figure 3. Sum rate of V2I links in urban scenario at 56 km/h

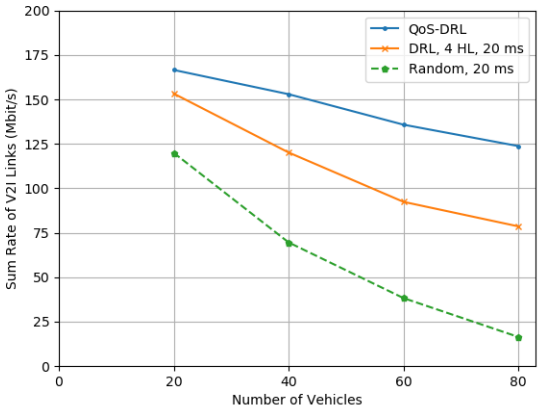

Figure 5. Sum Rate of V2I links in highway scenario at 70 km/h

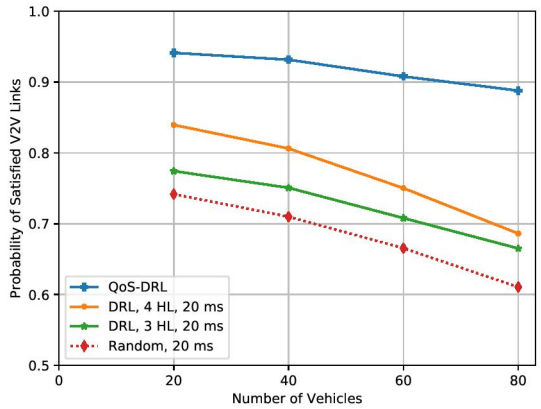

Figure 4. Probability of users meeting the latency constraint of V2V links in urban scenario at 56 km/h

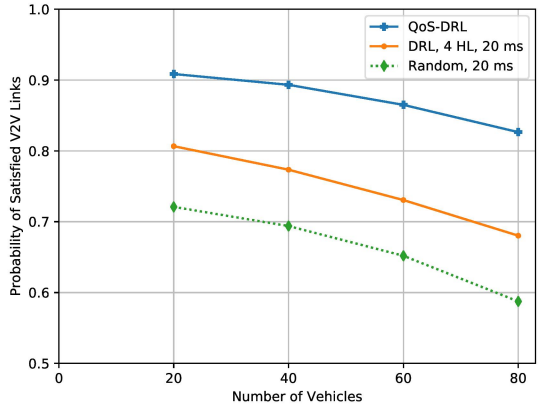

Figure 6. Probability of users meeting latency constraint of V2V links in highway scenario at 70 km/h

speed (70 km/h) and high speed (140 km/h). We evaluated for the highway scenario the DQN with 4 hidden layers. To best of our knowledge we did not find any previous work to compare our obtained results with. It is noted from the Figure 5 that for 70 km/h obtained capacity of V2I links is slightly reduced by 2.67% compared to the urban scenario in Figure 3. However, the DQL with priority levels is still outperforming the random method by 28%. It is to be noted that compared to urban scenario in Figure 3, the capacity of V2I links is only reduced by 2.64% in case of DQL with PPPP.

The probability of V-UE's meeting the latency constraints is increased with DQL with PPPP by 19.7% comapred to random method as seen in figure 6. Also, compared to DQL with 4 hidden layers the performance with QoS-based approach is increased by 11.2%. Similarly, compared to the urban scenario in Figure 4 the performance of DQL with PPPP is reduced by 2.17%.

A major impact on performance is seen at high speed of 140 km/h. As seen in Figure 7 the capacity of V2I links is significantly lowered compared to the urban scenario in Figure 3.The capacity of V2I links based on DQL with PPPP is reduced by 52.47% in comparision to the urban scenario. It is presuambly because of the fast variation of the channel gains at high speed.

Similarly, as seen in Figure 8, the probability of meeting the latency constraints of the V2V links is also reduced by 3.98% compared to the urban scenario in Figure 4. The reason is that the reward equation includes both the capacities of the V2I and V2V links as well the latency constraints. If the capacity of the V2I links is reduced the overall reward for the agent is also reduced. However, with QoS-based DRL approach still a significant improvement is seen in the probability of meeting the latency constraints in both the urban and highway scenario.

VI. Conclusion

In this paper, we have proposed a QoS-based DRL for V2X resource allocation. Each V2V link as an agent is deciding autonomously the transmission power and optimal sub-channel for transmission. The need of global information is eliminated because of the decentralized approach. From the simulation results, it is shown that for the urban scenario the agent can satisfactorily learn how to satisfy the V2V latency constraint while minimizing the interference to V2I communication. Using QoS-based DRL, the capacity for V2I improves on average compared to 28% compared to random based resource allocation method. However, for the highway scenario, the performance of the agent in terms of minimizing the interference is comparatively reduced by 52.4% compared to the urban scenario. This is assumed to be caused by the fast variation of the channel gains at high speed. Concluding, our proposed QoS-based DRL approach is able to maintain the probability of V2V links to satisfy the latency constraint close to 90%. Overall, the application of DL over

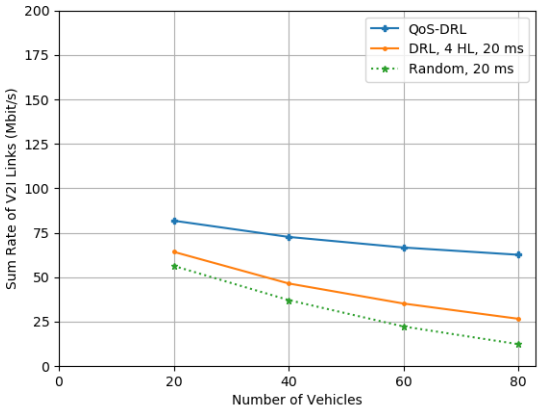

Figure 7. Sum Rate of V2I links in highway scenario at 140 km/h

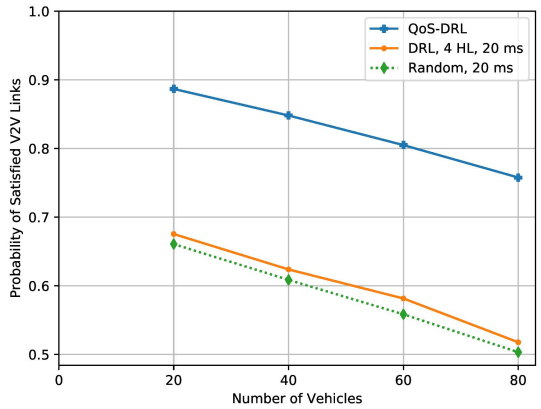

Figure 8. Probability of users meeting latency constraint of V2V links in highway scenario at 140 km/h

the classical methods provides a signifcant improvement to meet the required capacity and latency constraints. As next steps, we plan to analyze how machine learning approaches perform during more challenging realistic network scenarios, e.g. congestion, and in heterogeneous network setups.

ACKNOWLEDGMENT

We would like to thank Martin Leyh and Dariush Mohammad Soleymani for their insightful comments.

REFERENCES

[1] 3GPP, "TS 23.303: Group Services and System Aspects, Proximity-based services (ProSe), V 15.1.0, Release 13," 2016.

[2] 3GPP, "TR 36.885: Study on LTE-based V2X Services,V 14.0.0, Release 14 ," 2016.

[3] 3GPP, "TS 22.186: Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio transmission and reception, V 14.3.0, Release 14 ," 2017.

[4] D. Feng, L. Lu, Y. Yuan-Wu, G. Y. Li, S. Li, and G. Feng, "Device-to-device communications in cellular networks," *IEEE Communications Magazine*, vol. 52, no. 4, pp. 49–55, April 2014.

[5] H. Ye, L. Liang, G. Y. Li, J. Kim, L. Lu, and M. Wu, "Machine learning for vehicular networks: Recent advances and application examples," *IEEE Vehicular Technology Magazine*, vol. 13, no. 2, pp. 94–101, June 2018.

[6] H. Ye, G. Y. Li, and B. F. Juang, "Deep reinforcement learning based resource allocation for V2V communications," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3163–3173, April 2019.

[7] V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, "An introduction to deep reinforcement learning," *Foundations and Trends® in Machine Learning*, vol. 11, no. 3-4, pp. 219–354, 2018. [Online]. Available: https://rlseminar.github.io/static/files/intro-drl.pdf

[8] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013. [Online]. Available: https://arxiv.org/pdf/1312.5602.pdf

[9] S. Ruder, "An overview of gradient descent optimization algorithms," *CoRR*, vol. abs/1609.04747, 2016. [Online]. Available: http://arxiv.org/abs/1609.04747