Deep Reinforcement Learning based Distributed Resource Allocation for V2V Broadcasting

Hao Ye and Geoffrey Ye Li

School of Electrical and Computer Engineering
Georgia Institute of Technology
Email: yehao@gatech.edu and liye@ece.gatech.edu

***Abstract*—In this article, we exploit deep reinforcement learning for joint resource allocation and scheduling in vehicle-to-vehicle (V2V) broadcast communications. Each vehicle, considered as an autonomous agent, makes its decisions to find the messages and spectrum for transmission based on its local observations without requiring or having to wait for global information. From the simulation results, each vehicle can effectively learn how to ensure the stringent latency constraints on V2V links while minimizing the interference to vehicle-to-infrastructure (V2I) links.**

***Index Terms*—Deep Reinforcement Learning, V2V Communication, Resource Allocation**

I. Introduction

Vehicle-to-vehicle (V2V) communications allows cooperation and information sharing among vehicles in close proximity and play an essential role in intelligent transportation and road safety [1]–[4]. There are usually stringent quality-of-service (QoS) constraints on latency and reliability for the V2V communications. Since device-to-device (D2D) communications [5] have a potential to satisfying the QoS requirements for V2V applications, the Third Generation Partnership (3GPP) has supported V2V services based on D2D communications [6].

The spectrum used in D2D communications may underlay with the cellular users and effective resource allocation schemes are needed to minimize the interference between D2D and cellular users [5]. The high mobility of vehicular networks causes rapid changing wireless channels and makes it impossible to collect or track full channel state information (CSI) in a short timescale. Therefore, it is hard to use the traditional resource management approaches in the V2V communications since the full CSI is mostly assumed to be known in these approaches. To deal with the new challenges in V2V communications brought by the high mobility vehicles, centralized resource allocation schemes have been designed only based on the slowly varying large-scale fading information and the sum V2I ergodic capacity is optimized with V2V reliability guaranteed [7], [8].

Centralized resource allocation schemes, even if usually have good performance, require a large communication overhead to collect global information and therefore are not scalable to large networks. Recently, distributed resource management in V2V communications has been investigated in several related works. In [9], a distributed approach has been proposed to allocate sub-bands to the V2V link by exploiting the position information. Based on the similarities of positions and loads, the V2V links are grouped into different clusters and the resource blocks are allocated to each cluster. The assignments are then adjusted iteratively by swaps among V2V links within each cluster. The low-complexity algorithm in [10] optimizes outage probabilities for V2V communications based on bipartite matching. In [11], a deep reinforcement learning approach has been proposed to manage the resources based on local observations so that the V2V links have the least interference to the V2I links while the latency constraints on the V2V links are satisfied.

The above approaches are designed for unicast communications in vehicular networks. Nevertheless, to share the safety information among vehicles efficiently, sometimes broadcasting the messages is more appropriate than unicast communications since the targets for the traffic safety messages are usually for the vehicles within the surrounding area. However, broadcast is more challenging than the unicast in the vehicular system as multiple receivers must receive each broadcasting message within the latency constraint. To improve the broadcast reliability, some vehicles may need to rebroadcast the messages that have been received. However, blindly rebroadcasting the messages may cause package collisions during the transmission, which has been referred to as the broadcast storm [12], [13]. In [13], several forwarding node selection algorithms have been proposed based on the distance to the nearest sender, of which p-persistence provides the best performance.

In this paper, we will use deep reinforcement learning to handle resource allocation and the broadcast scheduling jointly. Recently, deep learning has made great successes in computer vision [14], speech recognition [15], and wireless communications [16]. By combining the advanced deep learning techniques with reinforcement learning, impressive improvement has been shown in many applications, including playing Go games [17], playing video games [18], and job scheduling in computing clusters [19]. In our proposed

This work was supported in part by a research gift from Intel Corporation and the National Science Foundation under Grants 1443894 and 1731017.

978-1-5386-2070-0/18/$31.00 ©2018 IEEE

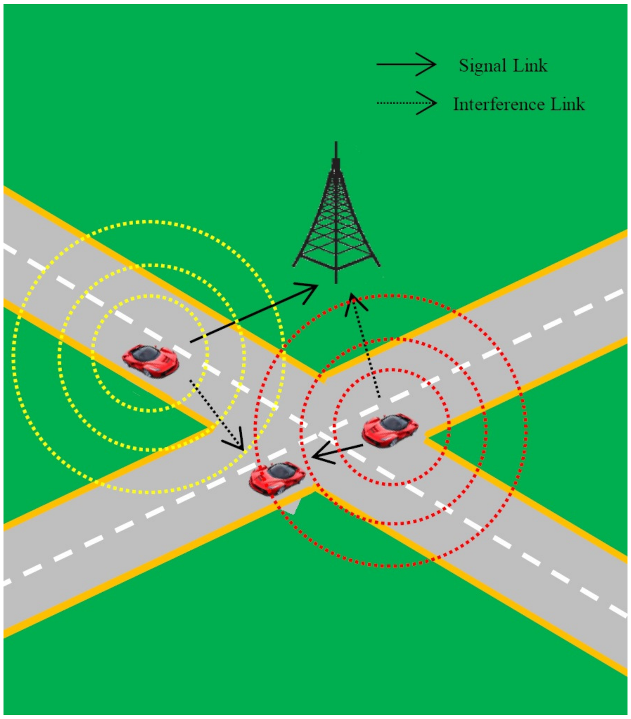

Fig. 1. An illustrative structure of vehicular communication networks.

method, multi-agent deep reinforcement learning is utilized to learn the mapping from the local observations to the joint resource management and scheduling solutions. Each vehicle, regarded as an agent, selects the spectrum and message for transmission according to the local information, including local CSI and the interference information. In general, the agent can effectively learn to balance between meeting the stringent latency requirement imposed on V2V transmission and minimizing the interference from the V2V links to the V2I links.

The main contribution of this article is using deep reinforcement learning to develop a joint distributed resource allocation and scheduling mechanism for in V2V broadcast communication, where the stringent constraints on latency can be directly addressed. According to the simulation results, our deep reinforcement learning based resource allocation scheme can effectively learn to minimize the interference to the V2I links while ensuring the latency constraint.

II. System Model

In this section, the system model and problem of joint resource allocation and scheduling are presented.

As shown in Fig. 1, the vehicular network contains M cellular users (CUEs) denoted by $\mathcal{M} = \{1, 2, ..., M\}$ demanding V2I links for high capacity communications. At the same time, there are K V2V users (VUEs) denoted by $\mathcal{K} = \{1, 2, ..., K\}$ demanding V2V links for broadcasting the safety messages to nearby vehicles, where each message is transmitted from one vehicle to a group of receivers within a surrounding area. To achieve a high spectrum utilization efficiency, the orthogonally allocated uplink spectrum for the V2I links is shared by the V2V links.

The interference to the V2I links comes from the background noise and the signals from the VUEs that share the same sub-band. Thus the signal-to-interference-plus-noise ratio (SINR) of V2I link can be expressed as

$$\gamma^c_m = \frac{P^c h_m}{\sigma^2 + \sum_{k\in\mathcal{K}} \rho_{m,k} P^d \tilde{h}_k}, \tag{1}$$

where P^c and P^d are the transmission powers of CUE and VUE, respectively, σ^2 is the noise power, h_m is the channel gain of the channel corresponding to the mth CUE, $\tilde{h}_k$ is the interference power gain of the kth VUE, and $\rho_{m,k}$ is the spectrum allocation indicator with $\rho_{m,k} = 1$ if the kth VUE reuses the spectrum of the mth CUE and $\rho_{m,k} = 0$ otherwise. Hence the capacity of the mth CUE can be expressed as

$$C^c_m = W \cdot \log(1 + \gamma_m), \tag{2}$$

where W is the bandwidth.

Similarly, for the jth receiver of the kth VUE, the SINR is

$$\gamma^d_{k,j} = \frac{P^c h_m}{\sigma^2 + G_c + G_d}, \tag{3}$$

with

$$G_c = \sum_{m\in\mathcal{M}} \rho_{m,k} P^c \tilde{g}_{m,k}, \tag{4}$$

and

$$G_d = \sum_{m\in\mathcal{M}} \sum_{k'\in\mathcal{K} k\neq k'} \rho_{m,k}\rho_{m,k'} P^d \tilde{g}^d_{k',k,j}, \tag{5}$$

where g_k is the power gain of kth VUE, $\tilde{g}_{k,m}$ is the interference power gain of the mth CUE, and $\tilde{g}^d_{k',k,j}$ is the interference power gain of the k'th VUE. The capacity for the jth receiver of the kth VUE can be expressed as

$$C^d_{k,j} = W \cdot \log(1 + \gamma^d_{k,j}). \tag{6}$$

In order to increase the reliability of V2V broadcast communications, some vehicles need to serve as relays and rebroadcast messages that have been received so that more receivers are able to get the messages within the latency requirement. However, the broadcast storm will occur if there are excessive redundant rebroadcast messages in the vehicular network. To remedy this problem, approaches of selecting messages to rebroadcast need to be carefully designed so that the messages can be disseminated to more vehicles while bringing little superfluous rebroadcasting to the vehicular network.

In the decentralized schemes, each vehicle will select the spectrum and messages to broadcast and these decisions are made based on local observations and are independent of the V2I communications. Therefore, given the resource allocation of the V2I links, the objective of the proposed autonomous scheme is to ensure that the latency constraints for the V2V links are met and the interference to the V2I links are minimized.

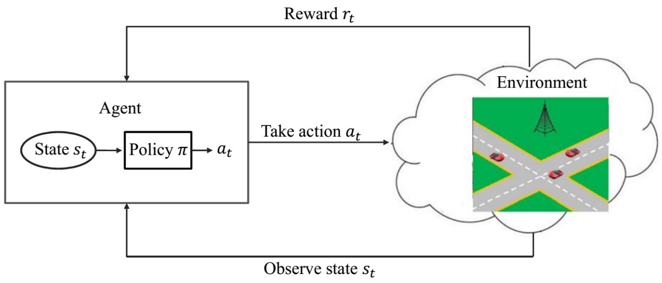

Fig. 2. Deep reinforcement learning for V2V communications

III. Distributed Resource Allocation

In this section, the framework on deep reinforcement learning for joint resource allocation and scheduling in V2V broadcast communications is introduced. The key parts are presented in detail and algorithms to train the deep Q-networks are shown as the proposed solution.

A. Reinforcement Learning Framework

The structure of reinforcement learning for resource allocation and scheduling in V2V broadcast communications is shown in Fig. 2, where an agent, corresponding to a vehicle, interacts with the environment. In this case, the environment is considered to be everything beyond the vehicle.

Each vehicle is regarded as an agent in our system. At each time t, the agent observes a state, s_t, from the state space, $\mathcal{S}$, and accordingly takes an action, a_t, from the action space, $\mathcal{A}$, selecting sub-band and messages based on the policy, π. With the actions taken by the agents, the environment transits to a new state s_{t+1} and the agent receives a reward, r_t, determined by the capacities of the V2I and V2V links and the latency constraints of the corresponding V2V message.

The state observed by each vehicle for each received message consists of several parts: the instant channel interference power to the link, $\mathbf{I_{t-1}}$, the channel information of the V2I link, e.g., from the V2V transmitter to the BS, $\mathbf{H_t}$, the selection of sub-channel of neighbors in the previous time slot, $\mathbf{N_{t-1}}$, the number of received times, O_t, the minimum distance to the vehicles that have broadcast the message, D_t, and the remaining time to meet the latency constraints, U_t. In summary, the state can be expressed as $s_t = [\mathbf{I_{t-1}}, \mathbf{H_t}, \mathbf{N_{t-1}}, O_t, D_t, U_t]$.

At each time, the agent takes an action at $a_t \in \mathcal{A}$, which includes determining the massages for broadcasting and the sub-channel for transmission. For each message, the dimension of action space is the $N_{RB}+1$, where N_{RB} is the number of resource blocks. If the agent takes an action from the first N_{RB} actions, the message will be broadcast immediately in the corresponding sub-channel. Otherwise, if the agent takes the last action, the message will not be broadcast at this time.

The objective of reinforcement learning is to minimize the interference to the V2I links with the latency constraints for VUEs guaranteed. In order to reach this objective, the frequency band and messages selected by each vehicle should have small interference to all V2I links as well as other VUEs and it also needs to meet the requirement of latency constraints. Therefore, the reward function consists of three parts, the capacity of V2I links, the capacity of V2V links, and the latency condition. To suppress the redundant rebroadcasting, only the capacities of receivers that have not received the message are taken into consideration. Therefore, no capacity of V2V links is considered if the rebroadcasting message has already been received by all the targeted receivers before. The latency condition is represented as a penalty if the message has not been received by all the targeted receivers, which increases linearly as the remaining time U_t decreases. Therefore, the reward function can be expressed as,

$$r_t = \lambda_c \sum_{m \in \mathcal{M}} C^c_m + \lambda_d \sum_{k \in \mathcal{K}, j \notin E\{k\}} C^d_{k,j} - \lambda_p (T_0 - U_t), \quad (7)$$

where T_0 is the constraint time and λ_c, λ_d, and λ_p are weights of the three parts, respectively, and $E\{k\}$ represents the set of the targeted receivers that have received the transmitted message.

The state transition and reward are stochastic and follow the Markov decision process (MDP), where the state transition probabilities and rewards depend only on the state of the environment and the action taken by the agent. The transition from s_t to s_{t+1} with reward r_t when action a_t is taken can be characterized by the conditional transition probability, $p(s_{t+1}, r_t | s_t, a_t)$. It should be noted that the agent can only control its own actions and has no prior knowledge on the transition probability matrix $\mathbf{P} = \{p(s_{t+1}, r_t | s_t, a_t)\}$, which is determined by the environment. The goal of reinforcement learning is to maximize the return defined as the expected cumulative discounted rewards,

$$G_t = \mathbb{E}[\sum_{n=0}^{\infty} \beta^n r_{t+n}], \quad (8)$$

where β is the discount factor.

B. Q-Learning

At each time, the agent takes an action, a_t, according to a policy, π, based on the observed state, s_t. As indicated before, the action, $a_t \in \mathcal{A}$, corresponds to how to select messages and spectrum given a state s_t described above in our problem.

In our proposed method, Q-learning is employed to find the optimal policy for the joint spectrum allocation and scheduling problem in the V2V broadcast communications. The Q-value for a given state-action pair, $Q(s_t, a_t)$, of policy π is defined as the expected accumulated discounted rewards when taking an action $a_t \in \mathcal{A}$ and following policy π thereafter. Given Q-values, $Q(s_t, a_t)$, an improved policy, π, can be easily constructed by taking the action that maximizes the long-term accumulated rewards, i.e.,

$$a_t = \arg\max_{a \in \mathcal{A}} Q(s_t, a). \quad (9)$$

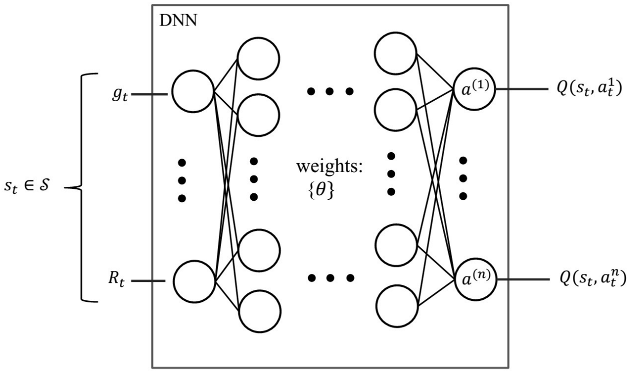

Fig. 3. Structure of Deep Q-networks

The optimal Q-values Q^* can be obtained based on iteratively updating,

$$\begin{aligned} Q_{new}(s_t, a_t) =& Q_{old}(s_t, a_t) + \alpha[r_{t+1} + \\ & \beta \max_{s \in \mathcal{S}} Q_{old}(s, a_t) - Q_{old}(s_t, a_t)], \end{aligned} \tag{10}$$

It has been proven that the Q-values will ultimately converge to the optimal $Q*$ with probability 1 under the assumption each action has been tried infinite times under each state. The optimal policy, π^*, can be found once the optimal Q-value, Q^*, is determined.

When applying Q-learning for V2V communications, the resource allocation and scheduling can be performed only based on the local observation, s_t, once optimal Q^* is known. The optimal Q^* can be obtained by computer simulation.

C. Deep Q Networks

The Q-learning works effectively when the state-action space is small, where a look-up table can be maintained for the update of the Q-value. However, it is impossible to apply the Q-learning with look-up tables when the state-action space becomes very large, as the joint resource management and scheduling problem. In this situation, many states may be rarely visited, thus the corresponding Q-values are seldom updated, leading to a much longer time to converge. As shown in Fig. 3, deep Q-network combines Q-learning with a deep neural network (DNN). The basic idea behind deep Q-network is to estimate the Q-values by a DNN function approximator with weights $\{\theta\}$ as a Q-network [20]. Once $\{\theta\}$ is determined, Q-values, $Q(s_t, a_t)$, will be the outputs of the DNN. The DNN can address sophisticate mappings between the channel information and the desired output based on a large amount of training data, which will be used to determine Q-values.

The Q-network updates its weights, θ, at each iteration to minimize the following loss function derived from the same Q-network with old weights on a data set D,

$$Loss(\theta) = \sum_{(s_t, a_t) \in D} (y - Q(s_t, a_t, \theta))^2, \tag{11}$$

where

$$y = r_t + \max_{a \in \mathcal{A}} Q_{old}(s_t, a, \theta), \tag{12}$$

where r_t is the corresponding reward.

D. Training and Testing Algorithms

The proposed deep Q-network is trained with a large amount of simulated data, which are generated from interactions of agents and an environment simulator. Each sample includes s_t, s_{t+1}, a_t, and r_t. Our simulator consists of VUEs and CUEs and their channels, where the vehicles are randomly dropped and the channels for CUEs and VUEs are generated based on the positions of the vehicles. In the training stage, we follow the deep Q-learning with experience replay [20] to suppress the temporal correlation in the generated data, where the generated data are saved in a storage called *memory*. As shown in Algorithm 1, the mini-batch data used for updating the Q-network is sampled from the *memory* in each iteration. The policy used in each vehicle is random at the beginning and is gradually improved with the updated Q-networks.

Since each vehicle is considered as an agent, the actions of other vehicles are unknown if they update their actions simultaneously and independently. As a result, the states that each agent observes cannot fully characterize the whole environment. To mitigate this problem, the vehicles are set to be updated asynchronously. At each time slot, only one or a small proportional of vehicles will update their selections of actions. In this way, for each agent, the environmental changes due to other agents' actions can be observed.

Algorithm 1 Training for Deep Reinforcement Learning

```
procedure TRAINING
Input: Q-network structure, environment simulator.
Output: Q-network
Start:
    Random initialize the policy $\pi$
    Initialize the model
    Start environment simulator, generate vehicles, VUE, CUE.
Loop:
    Random sample a vehicle in the system.
    Determine the messages and spectrum for transmission based on policy $\pi$
    Collect and save the data item {state, reward, action, post-state} into memory.
    Sample a mini-batch of data from the memory.
    Train the deep Q-network using the mini-batch data.
    Update the policy $\pi$: chose the action with maximum Q-value.
End Loop
Return: Return the deep Q-network
```

IV. Simulation Results

In this section, we present simulation results to demonstrate the performance of the proposed method. A single cell outdoor

TABLE I
SIMULATION PARAMETERS

Parameter	Value
Carrier frequency	2 GHz
Bandwidth	10 MHz
BS antenna height	25m
BS antenna gain	8dBi
BS receiver noise figure	5dB
Vehicle antenna height	1.5m
Vehicle antenna gain	3dBi
Vehicle receiver noise figure	9dB
Vehicle speed	36 km/h
Number of lanes	3 in each direction (12 in total)
Latency constraints for V2V links T_0	100 ms
V2V transmit power	23 dBm
SINR threshold	1 dB
Noise power σ^2	-114 dBm
[λ_c,λ_d,λ_p]	[0.2, 0.8, 1]

system with the carrier frequency of 2 GHz is considered. The simulation setup follows the Manhattan case detailed in 3GPP TR 36.885 [21], where there are 9 blocks in all and with both line-of-sight (LOS) and non-line-of-sight (NLOS) channels.

The structure of deep Q-network used in the experiments consists of a five-layer fully connected neural network with three hidden layers. The numbers of neurons in the three hidden layers are 500, 250 and 120, respectively. The Relu function is used as the activation function, defined as

$$f_r(x) = \max(0, x). \tag{13}$$

The learning rate is 0.01 at the beginning and decreases exponentially. We also utilize ϵ-greedy policy to balance the exploration and exploitation [20] and adaptive moment estimation method (Adam) for training [22]. The detail parameters can be found in Table 1.

The baseline method used for comparison consists of two parts, the scheduling protocol of broadcasting message selection and spectrum selection. The p-persistence is employed as the broadcasting protocol [13] and distributed grouping based resource allocation scheme from [9] is utilized as the baseline approach. The p-persistence protocol is that the probability of broadcasting is determined by the distance to the nearest sender. The larger the distance is, the higher probability it will have to rebroadcast the received message. In the grouping based spectrum selection method, vehicles are first grouped by the similarities and then the sub-bands are allocated and adjusted iteratively to the V2V links in each group.

The message of the kth V2V link is regarded to be successfully received by the jth receiver if the SINR, $\gamma^d_{k,j}$, is above the SINR threshold. The V2V transmission is considered as successful if all the targeted receivers of the message have successfully received the message.

A. V2V Latency

Fig. 4 shows the probability that VUEs satisfy the latency constraint versus the number of vehicles. From the figure, the proposed method has a larger probability for VUEs to satisfy the latency constraint since it can effectively select the messages and sub-band for transmission.

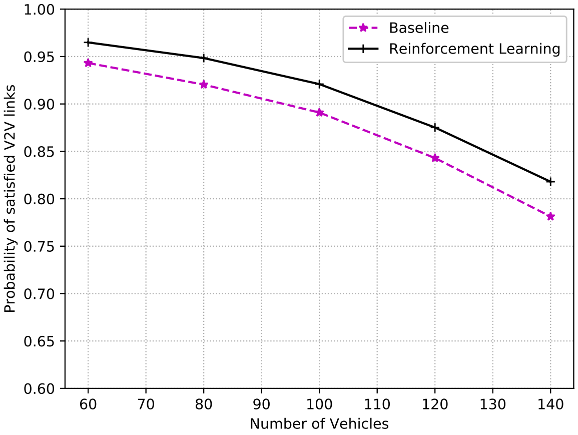

Fig. 4. Probability of satisfied VUEs versus the number vehicles.

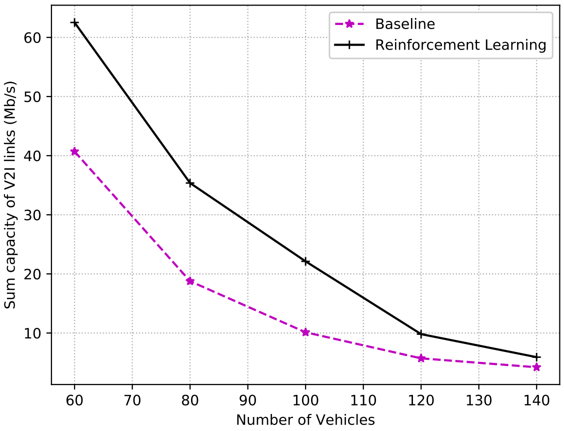

Fig. 5. Sum capacity versus the number of vehicles.

B. V2I Capacity

Fig. 5 shows the summation of V2I rate versus the number of vehicles. From the figure, the proposed method has a better performance to mitigate the interference of V2V links to the V2I communications.

V. CONCLUSION

In this article, a decentralized joint resource allocation and scheduling mechanism has been proposed for the V2V broadcast communications based on deep reinforcement learning. Each vehicle is regarded as an agent, making its own decisions to find optimal spectrum and messages for rebroadcasting. Since the proposed method is decentralized, the global information is not required for each agent to make its decisions, thus the transmission overhead is small. From the simulation results, each agent can learn how to satisfy the V2V constraints while minimizing the interference to V2I communications.

VI. Acknowledgment

The authors would like to thank Dr. May Wu, Dr. Satish C. Jha, Dr. Kathiravetpillai Sivanesan, Dr. Lu Lu, and Dr. JoonBeom Kim from Intel Corporation for their insightful comments, which have substantially improved the quality of this paper.

References

[1] L. Liang, H. Peng, G. Y. Li, and X. Shen, “Vehicular communications: A physical layer perspective,” *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 10647–10659, Dec. 2017.

[2] H. Peng, L. Liang, X. Shen, and G. Y. Li, “Vehicular communications: A network layer perspective,” *submitted to IEEE Trans. Veh. Technol. also in arXiv preprint arXiv:1707.09972*, 2017.

[3] H. Ye, L. Liang, G. Y. Li, J. Kim, L. Lu, and M. Wu, “Machine learning for vehicular networks,” *to appear IEEE Veh. Technol. Mag , also in arXiv preprint arXiv:1712.07143*, 2017.

[4] L. Liang, H. Ye, G. Y. Li, “Towards intelligent vehicular networks: A machine learning framework,” *submitted to IEEE Internet Things J., also in arXiv preprint arXiv:1804.00338*, 2018.

[5] D. Feng, L. Lu, Y. Yuan-Wu, G. Li, S. Li, and G. Feng, “Deviceto-device communications in cellular networks,” *IEEE Commun. Mag.*, vol. 52, no. 4, pp. 49–55, Apr. 2014.

[6] H. Seo, K. D. Lee, S. Yasukawa, Y. Peng, and P. Sartori. “LTE evolution for vehicle-to-everything services,” *IEEE Commun. Mag.*, vol. 54, no. 6, pp. 22–28, Jun. 2016.

[7] L. Liang, G. Y. Li, and W. Xu, “Resource allocation for D2D-enabled vehicular communications,” *IEEE Trans. Commun.*, vol. 65, no. 7, pp. 3186–3197, Jul. 2017.

[8] W. Sun, E. G. Strom, F. Brannstrom, K. C. Sou, and Y. Sui, “Radio resource management for D2D-based V2V communication,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, pp. 6636–6650, Aug. 2016.

[9] M. I. Ashraf, M. Bennis, C. Perfecto, and W. Saad, “Dynamic proximity-aware resource allocation in Vehicle-to-Vehicle (V2V) communications, ” in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2016, pp. 1–6.

[10] B. Bai, W. Chen, K. B. Letaief, and Z. Cao, “Low complexity outage optimal distributed channel allocation for vehicle-to-vehicle communications,” *IEEE J. Sel. Areas Commun*, vol. 29, no. 1, pp.161–172, Jan. 2011.

[11] H. Ye and G. Y. Li, “Deep reinforcement learning for resource allocation in V2V communications,” *to appear in IEEE ICC’ 18, also in arXiv preprint arXiv:1711.00968*, 2017.

[12] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, “The broadcast storm problem in a mobile ad hoc network,” in *Proc. ACM/IEEE MobiCom*, Aug. 1999, pp. 151–162

[13] O. Tonguz, N. Wisitpongphan, J. Parikh, F. Bai, P. Mudalige, V. Sadekar, “On the broadcast storm problem in ad hoc wireless networks”, in *Proc. BROADNETS*, Oct. 2006, pp. 1–11

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097-1105.

[15] C. Weng, D. Yu, S. Watanabe, and B. H. F. Juang, “Recurrent deep neural networks for robust speech recognition,” in *Proc. ICASSP*, May 2014, pp. 5532–5536.

[16] H. Ye, G. Y. Li, and B.-H. F. Juang, “Power of deep learning for channel estimation and signal detection in OFDM systems” *to appear in IEEE Wireless Commun. Lett.*, 2017.

[17] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershevlvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, “Mastering the game of go with deep neural networks and tree search,” *Nature*, vol. 529, no.7587, pp. 484–489, Jan. 2016.

[18] 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.

[19] H. Mao, M. Alizadeh, I. Menache, and S. Kandula, “Resource management with deep reinforcement learning,” in *Proc. of the 15th ACM Workshop on Hot Topics in Networks.* ACM, Nov. 2016, pp. 50–56.

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

[21] *3rd Generation Partnership Project: Technical Specification Group Radio Access Network: Study LTE-Based V2X Services: (Release 14)*, Standard 3GPP TR 36.885 V2.0.0, Jun. 2016.

[22] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014