



OPEN

Approximate reinforcement learning to control beaconing congestion in distributed networks

J. Aznar-Poveda[✉], A.-J. García-Sánchez, E. Egea-López & J. García-Haro

In vehicular communications, the increase of the channel load caused by excessive periodical messages (beacons) is an important aspect which must be controlled to ensure the appropriate operation of safety applications and driver-assistance systems. To date, the majority of congestion control solutions involve including additional information in the payload of the messages transmitted, which may jeopardize the appropriate operation of these control solutions when channel conditions are unfavorable, provoking packet losses. This study exploits the advantages of non-cooperative, distributed beaconing allocation, in which vehicles operate independently without requiring any costly road infrastructure. In particular, we formulate the beaconing rate control problem as a Markov Decision Process and solve it using approximate reinforcement learning to carry out optimal actions. Results obtained were compared with other traditional solutions, revealing that our approach, called SSFA, is able to keep a certain fraction of the channel capacity available, which guarantees the delivery of emergency-related notifications with faster convergence than other proposals. Moreover, good performance was obtained in terms of packet delivery and collision ratios.

Transportation is evolving in increasingly overpopulated cities due to a growing demand for goods and human transit. Unless current trends change, the number of vehicles on the road is predicted to triple by 2050¹, which will also contribute to increasing vehicle-related crashes, injuries, and death ratios. Under these circumstances, Intelligent Transportation Systems (ITS) and Vehicle-to-Vehicle (V2V) communications are expected to reduce these detrimental effects and extend the capabilities of numerous driver-assistance systems and driverless vehicles². The European Telecommunications Standards Institute (ETSI) defines V2V communications over the ITS-G5 radio channel, a 10 MHz control channel at the 5.9 GHz band of the IEEE 802.11p standard³. The ETSI Cooperative Awareness Service (CAS)⁴ transmits periodic broadcast single-hop messages, called beacons, throughout this control channel (Channel 172 in the US). Formally called Cooperative Awareness Messages (CAM) in Europe or Basic Safety Messages (BSM) in the US, beacons are responsible for disseminating status and environmental information among vehicles, where specific information such as position, speed, acceleration, direction, or vehicle dimension are employed to track and predict vehicle behavior. This broadcast information supports many safety applications and is crucial to reducing the risk of collision among vehicles or other undesired events⁵⁻⁷.

As the number of CAS beacons increases, the aggregated load can easily saturate the communication channel, compromising packet reception, and therefore endangering vehicle situation awareness. In this scenario, many safety applications⁵⁻⁷ based on beacons could receive outdated and inaccurate information. Furthermore, congestion negatively affects other services transmitted over the ITS-G5 radio channel, like Decentralized Environmental Notification (DEN), which notifies emergency services whenever an accident occurs on the road⁸. Not receiving such event-related messages, called DEN Messages (DENM), is of critical importance to the safety and health of road users (drivers, passengers, and pedestrians). To guarantee the delivery of these emergency-related messages (DENM), as well as to provide enough cooperation awareness (using CAM messages) to satisfy safety application requirements, a certain fraction of the channel capacity should be available. The upper limit of the channel load that can be dedicated to beaconing is usually called Maximum Beaconing Load (MBL). To satisfy the MBL constraint, the ETSI standard defines a Cross-Layer Decentralized Congestion Control (DCC) Management Entity⁹ to prevent the ITS-G5 radio channel from overloading.

Several transmission parameters can be tuned over time according to channel requirements and congestion. Keeping this in mind, the most widespread mechanism used to restrict congestion is decreasing the number of beacons transmitted per second. In¹⁰, authors proposed the LIMERIC method, in which each vehicle linearly updates its own rate (metric) depending on the total channel load, thus orienting rates toward a certain

Department of Information and Communications Technologies, Universidad Politécnica de Cartagena, 30202 Cartagena, Spain. ✉email: juan.aznar@upct.es

target value. This algorithm became so extended that even the standardization process included it in the DCC mechanism⁹. PULSAR¹¹ was another popular rate-based control algorithm using Additive Increase Multiplicative Decrease (AIMD) with feedback from 2-hop neighbors. Since the convergence of LIMERIC has not been tested when some vehicles are out of range with each other, PULSAR was used in combination with LIMERIC to solve this issue¹². With this combination, every vehicle sharing a link converges to the beaconing rate employed by the most congested link. The disadvantage of this solution is that it might unnecessarily decrease the beaconing rate of some vehicles, even though they are not congested. In¹³, authors proposed FABRIC, which dealt with congestion control as a Network Utilization Maximization (NUM) problem^{14,15} to optimally allocate beaconing rates. This approach allowed the design of simple algorithms with proven convergence. Note that these previous solutions are pure congestion controls in which only channel load is used to update beaconing rates. Conversely, other approaches found in the literature cope with the beaconing rate metric from some kind of prioritization. For instance, beacon inter-reception time is employed in¹⁶ to improve vehicle awareness. Similarly, other works adjust the beaconing rate in an attempt to minimize position tracking error with respect to other vehicles¹⁷. This was actually included in the US DCC standard¹⁸. Under this umbrella, the work in¹⁹, called EMBARC, is based on LIMERIC but integrates the tracking error algorithm of²⁰. Several works define some risk metrics related to vehicle dynamics and traffic situations, such as^{21–24}. In²¹, collision probability was employed to adjust the message or beaconing rate at intersections. The works^{22,23} employed tailgating collision risk to assign more resources to more dangerous vehicles. TTCC²⁴ extended FABRIC to transmit beacons using a more generic time-to-collision metric. Vehicle density was also used to fit beacon generation over time^{25,26}.

Most of the approaches mentioned above involve including additional information in beacon payloads. This implies that the congestion control procedure depends on beacon reception, which may disrupt congestion control performance in unfavorable channel conditions. As suggested in²⁷, tracking errors should be included in the congestion alleviation mechanisms. However, there is a more straightforward solution: not relying on neighbors' information to control congestion. These kinds of algorithms, commonly known as non-cooperative, are able to obtain a global change by means of individual actions. The most representative one is NORAC²⁸, a distributed beaconing rate control that employs game theory as its optimization core. As expected, NORAC does not involve exchanging control information, so each vehicle independently fits its beaconing rate according to the channel load measured. Despite the fact that some parameters can be used to adjust the behavior of NORAC, the MBL cannot be explicitly set. This leads to an insufficient or excessive channel load unless an appropriate combination of parameters is selected. However, non-cooperative proposals provide simple but effective resource allocation with very low computational cost, which results in a faster convergence speed to appropriate beaconing rates.

In this paper, we thoroughly discuss all these aforementioned aspects and conceive novel, non-cooperative congestion control capable of attaining an optimal MBL. Similar to game theory, we explore how decision theory and novel reinforcement learning (RL) techniques²⁹ can be applied to resolve a distributed optimization problem. As far as we know, most of the RL-based works require some kind of infrastructure; that is, they are designed for cellular networks^{30–32} or employ a more complex combination of parameters^{31,33–36}. However, none of them introduce simple, reliable, and fast beaconing rate control to alleviate congestion for V2V communications. We make use of a finite Markov Decision Problem (MDP) to formulate both the road environment and congestion control, which is later solved using approximate solution methods. In particular, we apply on-policy control with function approximation, which, unlike tabular solutions, allows us to generalize previous states to derive sensible decisions when new states are encountered. The resulting parameterized model can be applied by vehicles so the most appropriate beaconing rate is arrived at very efficiently in terms of runtime and computational cost, which is of great importance in congested scenarios. Results show that the policy, together with the model evaluated, called SSFA, successfully adjusts the channel load to an appropriate level. This means that road safety services, such as DEN, maintain a certain reserved bandwidth to guarantee the delivery of DENM notifications. Also, the proposed congestion alleviation mechanism does not require the installation of any costly infrastructure on the road (distributed) and does not depend on channel conditions to work properly (non-cooperative).

The remainder of this article is organized as follows. In “MDP formulation for congestion alleviation” section, we describe the resource (beaconing rate) allocation problem of V2V communications more thoroughly and introduce our proposal. Then, we validate it in “Results” section, comparing it with other algorithms and discussing the obtained results. Finally, “Conclusion” section summarizes the main conclusions.

MDP formulation for congestion alleviation

Excessive channel load might increase packet loss and hamper the operation of safety applications with outdated information, not to mention the fatal consequences of not receiving emergency notifications or DENMs. To overcome this problem, congestion control maintains the channel load near a certain target value, defined as the Maximum Beaconing Load (MBL). According to several works^{13,28,37,38}, the MBL is assumed to be around 60 or 70 percent of channel capacity (C), leaving the remaining percentage of the channel free to guarantee the delivery of DEN-related messages and other essential services. Since no a priori information or data about the (road) environment is available, we model the beaconing rate allocation problem as a finite Markov Decision Process (MDP), which is the basis of Reinforcement Learning (RL), to optimally satisfy this MBL constraint using discrete actions. In such a way, each vehicle takes actions, performs transitions among different states, and obtains different rewards depending on how well congestion is alleviated. This will be solved by means of approximated reinforcement learning techniques. The parameterized model resulting from these learning techniques can be easily evaluated by vehicles, causing the algorithm to converge significantly faster than other approaches^{13,28}.

MDPs are often employed to formulate optimization problems and later solve them by deriving optimal sequences of actions. This is particularly appropriate for complex environments that are partially random and difficult to predict. MDPs are mainly comprised of several entities. Firstly, *agents* are the learner entities that

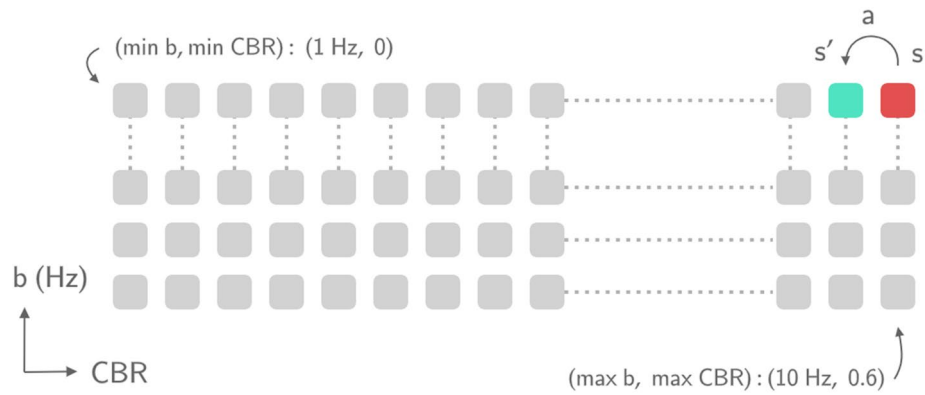


Figure 1. Two-dimensional space of states employed to model the beaconing rate allocation problem as an MDP. Axes represent each constituent element of the available states of the MDP: beaconing rate and CBR.

continuously seek for optimal behavior. In our case, vehicles evaluate policies to keep channel congestion under control. Secondly, the *environment* (road) is everything outside the agent (pedestrians, roads, or other agents) able to alter the agent *state*. The external environmental situation and the internal agent conditions are called *state*, usually defined as a vector $s \in \mathcal{S}$, with \mathcal{S} being the set of possible states. The agent is able to vary its state, from s to s' , by carrying out actions $a \in \mathcal{A}(s)$. Every time this happens, the environment is modified, and the agent obtains a *reward* r according to how appropriate the behavior of the agent has been. The agent acts over time in a bid to maximize the reward obtained, which can be modeled as a function of the state s and the action taken a , i.e., $r(s, a) = f(s, a) \in \mathbb{R}$.

The relationships among the different entities are usually determined by state-transition models, depicted by probabilities of transitioning among states. Nonetheless, in realistic scenarios with fast variations and partial information, MDP-solving algorithms employ a mapping between states and actions called policy; that is $\pi : \mathcal{S} \rightarrow \mathcal{A}$. Consequently, the main goal of the MDP-solving algorithm is to reach the optimal policy π^* that maximizes the accumulated sum of rewards during the entire training of the agent.

Particularization of actions and states. As mentioned above, the agents of the proposed MDP model, represented by vehicles, sense their environment to adequately adjust their beaconing rate, and thus reduce overall channel congestion. These changes are called *actions*, and they allow vehicles to reduce, maintain, or increase their current beaconing rate within the limits stated in the standard (1–10 Hz)⁴. The set of available discrete actions is called action space and is crucial to obtain a good training efficiency and later algorithm accuracy. For instance, too small actions (e.g. < 0.1 Hz) may lead to a more accurate solution but involving a huge state space, which takes much longer to be trained without incurring inaccuracies since many states may never be visited after a while. Once deployed, the convergence of our algorithm could also be affected if too many steps are required. In contrast, too big actions (e.g. > 1 Hz) simplify the training of the model at the expense of accuracy, so that the optimal value would rarely be reached over time. Note that the number of available actions can also make the dimension of the state space grow exponentially, to the detriment of effective training process. Therefore, an appropriate balance should be struck between training efficiency (directly related to the size of the state space) and proximity to the optimal value, in order to appropriately select the set of actions. In our particular case, the action space $\mathcal{A}(s) = \{0, \pm 0.5\}$ Hz was used.

Congestion is usually measured by using the Channel Busy Ratio (CBR), defined as the fraction of time (typically 1 s) during which the channel is busy due to transmissions or receptions. Another way of understanding the CBR is as the fraction of the channel load (sum of the neighbors' beaconing rates), over channel capacity. Note that this metric reflects external environment conditions. For instance, given a beaconing rate, a low measured CBR may be due (i) to a channel with high fading (lost packets results in a lower measured CBR) or (ii) because of having few neighbors. From the point of view of our solution, the action to perform would be the same irrespective of the real cause. In that sense, the CBR captures well the particularities of different scenarios. Because of this, the performance of our proposed solution is robust to variations in the channel model or radio propagation effects, as will be shown in the results section.

Taking this into account, let us define the states of the MDP model as the tuple comprised of the current beaconing rate and the CBR measured $s = (b, CBR)$. Up to 789 different CBR values (60% of the channel capacity in beacons per second) from 0 to 0.6 (MBL/C) are included in the MDP model, which results in 15,780 different tuple states. Every sensed state above or below these limits would result in a decrease or increase of the beaconing rate, respectively. As shown in Fig. 1, the proposed space of states can be illustrated on a two-dimensional plane, where the axes represent both the current beaconing rate and the CBR measured. When executing an action $a \in \mathcal{A}(s)$, the environment gives a new state s' back to the vehicle. The beaconing rate only applies the action value to the state. If, for instance, a lot of vehicles are transmitting at 10 Hz (beaconing rate) but suddenly they experience slight congestion and $a = -0.5$, they will decrease their beaconing rate to 9.5 Hz.

Reward function. With each transition among states, the agent obtains a reward $r(s, a) \in \mathbb{R}$, which denotes how much the current behavior differs from the desired behavior. In other words, maximizing the accumulated reward allows the agent to approach the optimal transmission parameters recommended by the optimal policy π^* . In our case, the desired behavior is to maintain the channel load around the MBL, typically between 60 and 70 percent of channel capacity. Note that higher loads may increase packet loss, jeopardizing vehicle context awareness and preventing emergency-related messages from being decoded. In contrast, a lower CBR would imply that the channel is underused, causing a loss of awareness. To overcome these drawbacks and thus achieve an optimal channel load, the reward is shaped according to the following function:

$$r(x) = -x \operatorname{sign}(x - \delta), \quad (1)$$

where sign is the signum function shifted by target value δ . In our case, the input x is the CBR, whereas the target value δ would be the MBL over channel capacity. Let us denote this upper CBR limit as the Maximum Beaconing Ratio ($MBR = MBL/C$). With this function, an increasing positive reward is obtained as the CBR approaches the target value MBR (0.6–0.7). However, if the CBR overruns this limit, a decreasing negative reward is obtained. These negative rewards speed up the learning process compared to using only positive ones, as proved in²⁹. In short, reaching the MBR limit not only allows us to reduce congestion and leave a certain fraction of the channel free to guarantee the delivery of emergency-related messages but also prevents channel underutilization.

Policy derivation. Once the states, actions, and rewards of the MDP have been defined, agents should learn the most advantageous policy; that is, to determine the sequence of actions for which the total reward is maximized. To this end, we employ Semi-gradient SARSA with Function Approximation (hence the name of our congestion alleviation mechanism: SSFA). SARSA iteratively updates the policy to achieve as large a reward as possible over time. Its name comes from the fact that the action is updated in function of the current state (s), the action selected (a), the reward obtained (r), the new state (s'), and the next action selected by the agent (a') in the new state. In difference to the training environment described above which was defined with discrete MDP states, the state space is continuous in real conditions (or their simulated counterparts) while evaluating the policy. Making sensible decisions in these unknown states entails generalizing from previous states that are similar to the current one. To this end, we attempt to generalize using function approximation; that is, we approximate the state-action value function, $Q(s, a)$, as a parameterized function $\hat{Q}(s, a)$, as follows:

$$Q(s, a) \approx \hat{Q}(s, a) = \theta_0 + \theta_1 x_1(s, a) + \dots + \theta_n x_n(s, a), \quad (2)$$

where $\theta \in \mathbb{R}^n$ is an n -dimensional weight vector to be learned and $\bar{x} = \langle x_1, \dots, x_n \rangle$ the n -tuple ($n=5$) comprised of the following features:

- $x_1(s, a)$ represents an additional reward given whenever ideal behavior is reached. Therefore, it has a value of 1 if the vehicle senses a CBR = MBR and 0 otherwise.
- $x_2(s, a)$ and $x_3(s, a)$ are congestion indicators, which are useful to lead the vehicle to the desired behavior and to define whether the channel load is congested or not. They have a value of 1 if the vehicle has high or low congestion, respectively; that is, the CBR experienced is above or below the MBR, and 0 otherwise.
- $x_4(s, a)$ and $x_5(s, a)$ provide subtle information about how the algorithm should proceed in detail. They assess whether the associated action is approaching or moving away from the desired behavior. In particular, they have a value of 1 if the CBR measured after carrying out the action is closer or further from the MBR, respectively, and 0 otherwise.

As can be observed, these five functions are modeled to obtain the desired behavior so the algorithm does not depend so much on how the reward is shaped. In short, using function approximation not only allows better generalization when assessing the policy but also speeds up the learning process and eases reward tuning. It is worth noting that more sophisticated RL algorithms, suitable for continuous action spaces, could be used to directly predict the optimal beaconing rate. Nevertheless, these solutions entail tough reward modeling to obtain the desired behavior in the right way as well as longer training times and subtle hyperparameter tuning. In contrast, we provide a simple congestion alleviation mechanism that can be trained straightforwardly and is ready to be deployed in realistic scenarios. Furthermore, as will be seen in the next section, the results obtained are close to the optimal values proposed by baseline works, and more complex algorithms might not mean a significant improvement.

The complete environment and the solving algorithm of the MDP model proposed have been implemented in Python, using different classes, objects, and advanced libraries, like NumPy³⁹, to obtain efficient data processing. The environment is represented by a set of vehicles arbitrarily located on a two-dimensional plane, as would occur on realistic roads. The interactions between the agents and the environment, such as rewards and transitions among states, are also implemented. In this (training) environment, each vehicle includes its current state (CBR and beaconing rate), transmission power, spatial location, and the set of allowed actions. Note that the proposed congestion control algorithm is not influenced by channel model or propagation effects so we assume here a free-space channel model and sufficient transmission power to permit vehicles to be in coverage with each other. In this way, numerous congestion levels represented by different numbers of vehicles allow a policy able to respond quickly to each CBR measured, from 0 to 0.6, to be obtained. This CBR is controlled in a distributed fashion by each vehicle according to the way the reward is modeled.

The semi-gradient SARSA with function approximations described in Algorithm 1 was implemented in a different class from the environment. Firstly, the parameterized action-state pairs, or \hat{Q} -values, are initialized to

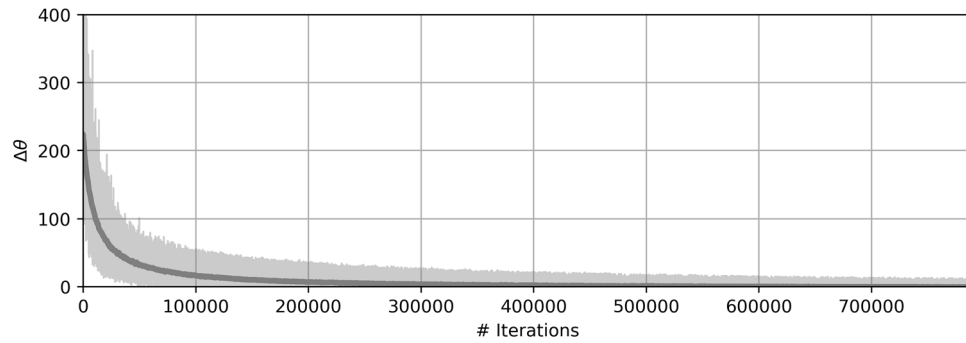


Figure 2. Biggest variation of consecutive θ values for each episode.

zero. For each episode, the environment generates as many vehicles as required to represent the CBR indicated in the current state and iteratively calculates the expected rewards and updated \hat{Q} -values. For instance, if the initial state is $s = (b, CBR) = (10\text{Hz}, 0.5)$, the environment will create as many vehicles transmitting at 10 Hz as possible to obtain a $CBR = 0.5$. The algorithm will recommend actions to every vehicle equally until reaching the optimal policy $\pi^* = f(\theta^*)$ and maximizing the accumulated reward during the training. Note that the policy is shared among vehicles and that overall channel congestion is successfully controlled in this training scenario. As will be shown in the following section, this shared policy will work appropriately even when all the vehicles are not in range of each other since individual contributions lead to the right overall channel load. This is the advantage of non-cooperative algorithms: they can obtain a global change by means of individual actions.

Algorithm 1 Pseudocode of semi-gradient SARSA with function approximation

Inputs

Step size $\alpha \in (0,1]$, small epsilon $\epsilon > 0$, discount factor $\gamma \in [0,1]$

Set of features $\bar{x} = (x_1, \dots, x_n)$

Initialize value-function weights $\theta \in \mathbb{R}^d$ arbitrarily

Loop for each episode

Initialize s and select action a

Loop for each step of episode

Act a

Observe reward r and new state s'

Select action a' using policy derived from $\hat{Q}(s', \cdot, \theta)$ (e.g., ϵ -greedy)

$\theta \leftarrow \theta + \alpha [r + \gamma \hat{Q}(s', a', \theta) - \hat{Q}(s, a, \theta)] \nabla \hat{Q}(s, a, \theta)$

$s \leftarrow s'$

$a \leftarrow a'$

End loop

End loop

It should be noted that attaining the optimal policy is not guaranteed. For this reason, we continued training until we achieved the desired behavior ($CBR \approx MBR$). To illustrate this, the learning curve of the proposed algorithm has been plotted in Fig. 2 through the biggest change of consecutive θ vectors, called $\Delta\theta$. This value was calculated as the sum of the difference between the elements of successive θ vectors. As can be observed, the biggest changes between consecutive θ values decrease as training moves forward, which implies better performance.

The most meaningful parameters of the environment as well as those employed in the MDP-solving algorithm have been summarized in Table 1. In the next section, the performance of the proposed congestion alleviation mechanism in different motorway and urban scenarios will be thoroughly assessed.

Results

In this section, a well-trained SSFA model is evaluated using different studies ranging from simple, theoretical evaluations to more complex and realistic simulations in urban and motorway scenarios. To this end, the resulting policy is loaded onto vehicles to execute the SSFA mechanism, as shown in Algorithm 2. Firstly, each vehicle measures the CBR and initializes the beaconing rate (10 Hz by default). Once the vehicles become aware of their state, the policy function modeled by the weights gives the optimal action back. Then, the policy is evaluated as many times as there are different available rates ($|\mathcal{B}|$) as a preemptive measure to avoid overlooking possible inaccuracies in the trained policy. This way, vehicles are led to reduce overall congestion in a distributed and non-cooperative fashion. Note that this is achieved thanks to the individual contributions of all the vehicles in

Parameter	Value
Discount factor (γ)	0.9
Step size (α)	0.1
Epsilon-greedy probability (ϵ)	0.1
Channel capacity (C)	1315.78 beacons/s
Maximum Beaconing Load (MBL)	789.47 beacons/s
Maximum Beaconing Ratio (MBR)	0.6
Transmission power	500 mW (27 dBm)
Min., Max. beaconing rate	1, 10 Hz
Number of available actions ($ \mathcal{A} $)	3
Number of available rates ($ \mathcal{B} $)	20
Number of available CBRs ($ \mathcal{L} $)	789
Total number of states ($ \mathcal{S} $)	$ \mathcal{B} \times \mathcal{L} $
Episodes	$(\mathcal{S} \times \mathcal{A})/MBR$
Steps of episode	100

Table 1. Training parameters and their values.

the network, which follow the same policy and act equally under similar states. Finally, according to the received action, SSFA (each vehicle) adjusts the beaconing rate that will be used until the next update.

Algorithm 2 Pseudocode of the non-cooperative SSFA mechanism

Inputs

Load trained policy $\pi := f(\theta)$

$s \leftarrow (b_o, CBR_o)$

Loop over time t

For each vehicle

Measure CBR

For $i = 1 \rightarrow |\mathcal{B}|$ do

$s \leftarrow (b, CBR)$

$b \leftarrow b + \theta(s)$

End For

End For

End loop

SSFA allocates beaconing rates without cooperation among vehicles and without relying on any base station or road infrastructure. Therefore, we compare it with two similar congestion control mechanisms found in the literature. The first solution in the comparison is NORAC²⁸, which employs game theory to allocate the beaconing rate depending on the measured CBR. Nevertheless, as discussed in “Introduction” section, no channel load limit can be explicitly set. In other words, the proper combination of parameters to reach a given MBL is, a priori, unknown, which means that the MBL constraint might not be met when traffic conditions vary. The second comparative solution is FABRIC¹³, which approaches the beaconing rate allocation as a Network Utilization Maximization (NUM) problem with proven convergence. Despite providing optimal allocation, FABRIC entails including Lagrange multipliers in the header of the transmitted messages (these multipliers, also known as prices, π , should not be confused with the policy that defines agent behavior). This means that vehicles require additional information about their neighbors, which may increase the convergence time. It is noteworthy that the authors of NORAC criticized FABRIC for piggybacking these prices. Nonetheless, we consider that adding a few extra bytes in the heading is not as serious a problem as longer convergence time. The comparison of (i) our non-cooperative approach based on decision theory (RL), (ii) FABRIC, which is a cooperative solution employing NUM, and (iii) NORAC, a non-cooperative solution based on game theory, is performed by making use of the following metrics:

- Channel Busy Ratio (CBR). The CBR is defined as the ratio between channel load and channel capacity. Furthermore, it can be interpreted as the fraction of busy time (typically 1 s) due to transmissions or receptions. As seen throughout this work, this metric represents how much of the channel is used (congested) so it is closely related to packet loss.
- Neighboring vehicles. Finally, together with the CBR, the number of neighbors detected provides valuable insight into the distribution of resources (and context awareness) among vehicles, which should also be considered when assessing the aforementioned algorithms.

Parameter	Value
Frequency band	5.9 GHz
Channel model	Nakagami-m
Carrier sense threshold	-92 dBm
Noise floor	-110 dBm
SNIR threshold	4 dB
Data rate	6 Mbps
Transmission power	500 mW (27 dBm)
Beacon size	4288 bits
Channel capacity (C)	1315.78 beacons/s
Maximum beaoning load (MBL)	789.47 beacons/s
Maximum beaoning ratio (MBR)	0.6
Min., Max. beaoning rate	1, 10 Hz
FABRIC parameter	
α	1
β	$2.8e-7$
ω	1
π_0	0.001252
NORAC parameter	
u_i	5
p_{c_i}	0.2

Table 2. OMNeT + + simulation parameters.

- Packet Delivery Ratio (PDR). The PDR is usually defined as the sum of successfully decoded packets with respect to the number of packets transmitted in the network^{40,41}. In our particular case, we employ a transmitter-centric approach in which the PDR is measured as the transmitted packets that are successfully received at a certain distance over the total number of packets transmitted. More to the point, the PDR is calculated every 50 m from the source vehicle.
- Packet Collision Ratio (PCR). We assume the PCR as the number of packets lost due to a collision (n_c) between the packets successfully decoded (n_s) and those lost due to a collision, $n_c/(n_c + n_s)$, as suggested in³⁶.

The scenarios simulated below are not only conducted using different channel conditions and environments, but also a variable number of vehicles. In the following sections, we simulate 650 static vehicles under training conditions, 400 static vehicles under realistic conditions, and up to 400 moving vehicles gradually introduced in a realistic urban scenario. For all these scenarios, a beacon size of 536 bytes and a fixed data rate of 6 Mbps were employed. According to the standard³, this results in a total PHY packet duration of 760 μ s and channel capacity of $C = 1315.78$ beacons per second. The whole set of simulation parameters are depicted in Table 2.

Evaluation under training conditions. As a first step, we evaluate the proposed congestion control using the Python environment, maintaining most of the training conditions. Therefore, resource allocation is performed theoretically, without sending messages or considering any interference phenomena, headers, or further MAC or PHY protocols. However, unlike the training, in which vehicles were randomly distributed and transmission power was high enough to reach every other vehicle, we now introduce a limited communication range of 400 m. We employ a single row of 650 vehicles evenly spaced along 2000 m. Every algorithm under comparison was run for 100 successive iterations. This evaluation is useful to check whether our proposed mechanism obtains the desired overall congestion (CBR) even when vehicles act in a non-cooperative way. As shown in Fig. 3, the beaoning rate obtained by evaluating the resulting policy (SSFA) is similar to the optimal response of FABRIC. NORAC has a rougher, oscillating shape, although the allocation tends toward the same limit as SSFA and FABRIC by the middle of the row of vehicles. Concerning the CBR, the three algorithms perform well in terms of reaching the MBR constraint, which is satisfied except in the transition to the edges, around 350 and 1600 m. The evolution over time for a vehicle located in the middle of the row was also obtained to study the convergence time. In this case, our proposed SSFA algorithm converges faster (around 15 iterations) than NORAC (40 iterations) and FABRIC (30 iterations).

Realistic uniformly spaced vehicles. In this subsection, we rigorously assess the performance of each comparative algorithm with the well-known discrete event simulator of networks OMNeT + + 5.3⁴². The INET 3.5 library⁴³ was used to implement the IEEE 802.11p standard as well as realistic channel, propagation, and interference models. To observe whether these realistic conditions affect the process of how resources are allocated, we deploy a similar scenario to the previous subsection. In particular, a row of 400 static vehicles uniformly spaced along 2000 m is simulated for 30 s. As illustrated in Fig. 4, SSFA obtains reliable beaoning rate

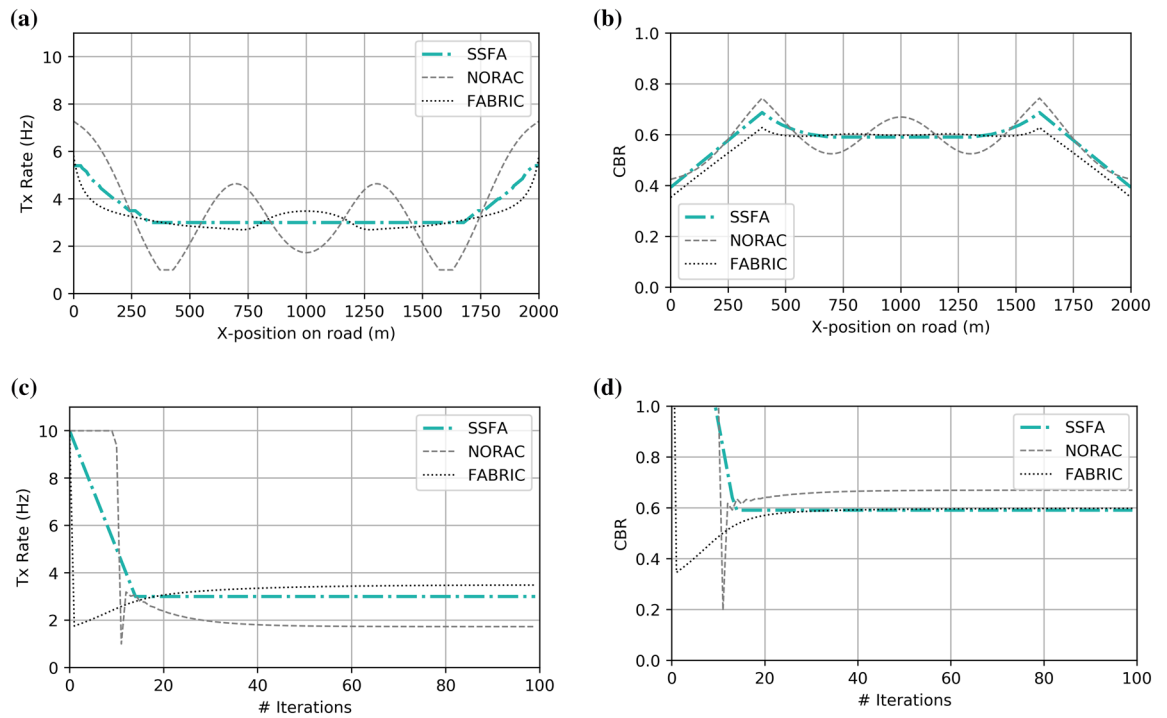


Figure 3. Theoretical comparison (implemented in Python) of the proposed congestion control approach with FABRIC and NORAC. (a) Recommended beaoning rate and (b) CBR measured for a row of vehicles versus their position on the road; (c) Evolution of the beaoning rate and (d) CBR of a vehicle located in the middle area of the road over time.

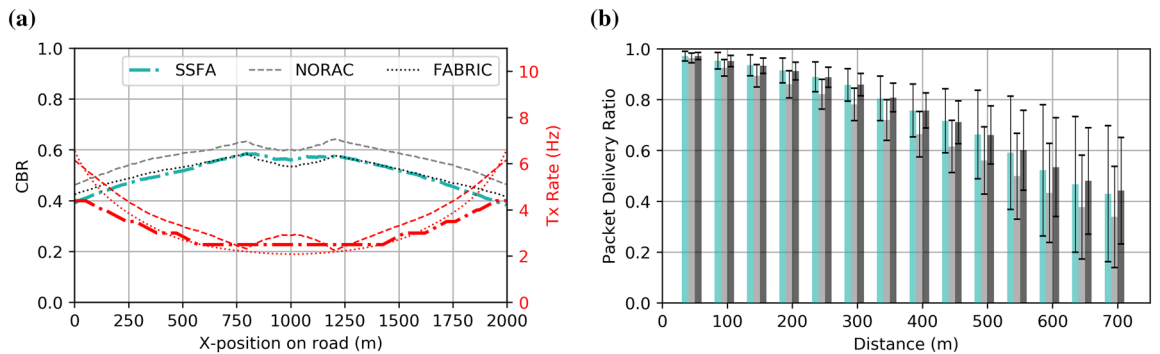


Figure 4. Realistic simulation (OMNeT++) of our proposed congestion control approach compared to FABRIC and NORAC for an evenly spaced row of vehicles. (a) Beaoning rate and CBR measured versus the vehicles' position on the road; (b) Packet Delivery Ratio over different distances.

allocation, comparable to the optimal rate proposed by FABRIC, and similar also to NORAC. Regarding the CBR measured, SSFA and FABRIC provide identical responses, whereas NORAC slightly exceeds the MBR limit. Recall that NORAC does not allow the upper CBR limit to be specified. Therefore, not reaching the desired CBR of 0.6 exactly means that the parameters selected were probably not optimal. This non-compliance brings interesting outcomes in terms of packet delivery ratio, decoded packets, and packet collision ratio, as shown in Fig. 4b and Table 3, respectively. As expected, SSFA and FABRIC reach a similar packet delivery ratio, but that of NORAC is reduced. Such an effect highlights the importance of congestion control and proves that the MBR used (0.6) is the value which allows us to obtain the best performance, as studied in numerous works^{13,28,37,38}. Since NORAC surpasses the MBR limit, the beaoning rate, and thus the number of decoded packets, is slightly higher than with the FABRIC and SSFA algorithms. However, the packet collision ratio is significantly higher than in the SSFA and FABRIC approaches. So far, not only does our proposal improve the convergence time but it also obtains excellent results in terms of PDR and PCR. Moreover, it is important to note that, unlike FABRIC, which is based on exchanging prices among neighboring vehicles, SSFA does not depend on channel conditions or packet delivery to operate properly.

Algorithm	PCR ± std	Decoded packets
Realistic uniformly spaced vehicles		
SSFA	0.1115 ± 0.0886	6,706,167
NORAC	0.1530 ± 0.1040	6,851,859
FABRIC	0.1144 ± 0.0902	6,736,288
Realistic urban scenario		
SSFA	0.1341 ± 0.0620	1,368,140
NORAC	0.1344 ± 0.0619	1,286,087
FABRIC	0.1536 ± 0.0649	1,457,903

Table 3. Packet Collision Ratio and total number of decoded packets.

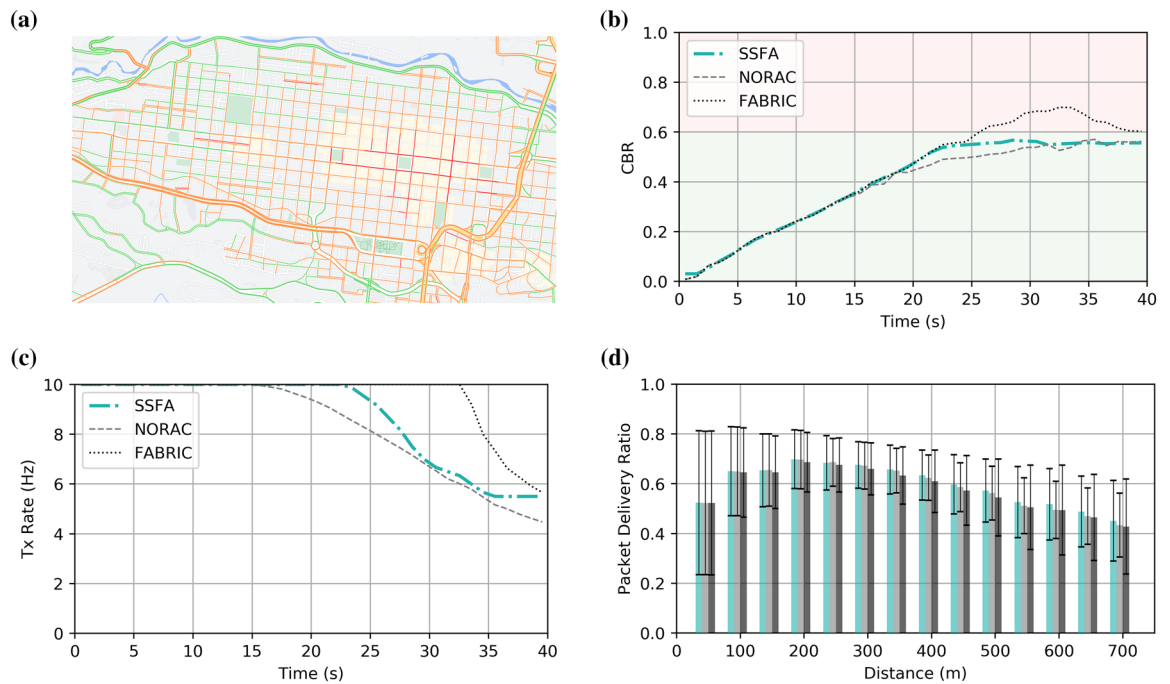


Figure 5. Realistic urban simulation (OMNeT ++ and SUMO) of the proposed congestion control approach compared to FABRIC and NORAC. **(a)** Traffic map (Map data ©2021 Google) of the city of Pereira (Risarcaldá, Colombia), used in the simulations, illustrating different levels of congestion (from low; green, to high; red) during the peak period (4 p.m.); **(b)** CBR measured and **(c)** allocated beaconing rate of a sample vehicle over time; **(d)** average packet delivery ratio for different vehicles over distance.

Realistic urban scenario. Finally, we evaluate SSFA in a realistic urban scenario where, unlike in the previous subsections, vehicles are not uniformly spaced. This will put the non-cooperative scheme to the test because the requirements of the vehicles might differ significantly among neighboring vehicles. For instance, some vehicles could experience congestion while stopped in a traffic jam or at a traffic light, whereas others could flow at higher speeds in a secondary street. In short, this realistic scenario will show how well SSFA performs under congested and stressful conditions considering rapid variations. To this end, in addition to OMNeT ++ and INET, we use Simulation of Urban MObility (SUMO)⁴⁴, a traffic simulation package designed for large networks. The OSM web wizard of SUMO allowed us to select a geographic region and specify traffic mode and demand. We imported the traffic map of the city of Pereira, Colombia. The different levels of congestion in this city (from low; green, to high; red) during the peak period (4 p.m.) are illustrated in Fig. 5a. To simulate this congestion, we sequentially introduce up to 400 vehicles, which will be randomly traveling around the city during the whole simulation time (40 s). The behavior of the first vehicle introduced in the network will be studied so that the congestion experienced by this vehicle increases over time. The goal of this experiment is to observe whether the SSFA approach performs well in terms of convergence time as well as to obtain an adequate CBR compared to other solutions. As can be shown in Fig. 5b, SSFA and NORAC cause the CBR to be the set MBR. However, FABRIC maintains the maximum beaconing rate even after experiencing congestion (Fig. 5c) so the CBR takes longer to converge and to reach the target MBR. This is a disadvantage of cooperative schemes under varying conditions. Concerning the PDR (Fig. 5d), SSFA obtains higher values than its counterparts in almost every

distance measured, from 0 to 700 m. It should be noted that the location of vehicles is now random around the simulated urban environment so the PDR has a different distribution than in the previous scenario. The PCR obtained, shown in Table 3, is also similar to or lower than that of FABRIC and NORAC. Therefore, not limiting the CBR properly may lead the algorithms to transmit pointless beacons that not only will be lost but that will impede the reception of DENM messages. In other words, we want to transmit only those messages that will be successfully decoded (Table 3) in order to not congest the channel. Finally, note that vehicles are now introduced gradually so that the total number of decoded packets is smaller with respect to the previous scenario.

Conclusions and future work

V2V communications are based on transmitting periodic messages (beacons) which support most safety applications and driver assistance systems. However, the associated channel load stemming from beacons should be controlled since it might saturate the channel and hamper the appropriate operation of these applications and services. For this reason, congestion control algorithms aimed at maintaining a given fraction of the channel free are of great importance to preserve the safety of road users, especially by guaranteeing the delivery of emergency-related notifications (DENMs). In this work, we introduce innovative beaconing rate control to alleviate congestion. We make use of approximate reinforcement learning, which allows vehicles to take sensible actions with low computational cost and converge in a short period of time. Our proposal, called SSFA, restricts the channel load by adapting the beaconing rate in a non-cooperative way. Since no additional information from neighbors is required and vehicles work independently, the algorithm is robust even in unfavorable conditions in which packet losses are significant. Moreover, SSFA operates in a distributed manner, thus no pre-installed infrastructure is required for its operation. Results reveal that SSFA successfully maintains channel usage at the desired level, leaving channel capacity free enough for successful DENM reception. Also, a higher packet delivery ratio and a lower number of collisions than other related mechanisms are achieved. In future works, we will focus on the design of algorithms with improved learning capabilities while driving in real implementations.

Received: 31 July 2021; Accepted: 16 December 2021

Published online: 07 January 2022

References

- Sitty, G. & Taft, N. *What will the global light-duty vehicle fleet look like through 2050?* (2016).
- Liang, L., Peng, H., Li, G. Y. & Shen, X. Vehicular communications: A physical layer perspective. *IEEE Trans. Veh. Technol.* **66**, 10647–10659 (2017).
- ITS. EN 302 663 - V1.3.0 - *Intelligent Transport Systems (ITS); ITS-G5 Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band.* (2019).
- ITS. EN 302 637-2 - V1.4.1 - *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service.* (2019).
- ITS. TS 101 539-3 - V1.1.1 - *Intelligent Transport Systems (ITS); V2X Applications; Part 3: Longitudinal Collision Risk Warning (LCRW) application requirements specification.* (2013).
- ITS. TS 101 539-2 - V1.1.1 - *Intelligent Transport Systems (ITS); V2X Applications; Part 2: Intersection Collision Risk Warning (ICRW) application requirements specification.* (2018).
- ITS. *Intelligent Transport Systems (ITS); V2X Applications; Part 1: Road Hazard Signalling (RHS) application requirements specification.* (2013).
- Standard, E. EN 302 637-3 - V1.2.1 - *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service.* (2014).
- ITS. TS 103 175 - V1.1.1 - *Intelligent Transport Systems (ITS); Cross Layer DCC Management Entity for operation in the ITS G5A and ITS G5B medium.* (2015).
- Kenney, J. B., Bansal, G. & Rohrs, C. E. LIMERIC: A linear message rate control algorithm for vehicular dsrc systems. in *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM 21–30* (2011). doi:<https://doi.org/10.1145/2030698.2030702>
- Tielert, T., Jiang, D., Chen, Q., Delgrossi, L. & Hartenstein, H. Design methodology and evaluation of rate adaptation based congestion control for vehicle safety communications. *IEEE Veh. Netw. Conf. VNC 116–123* (2011). doi:<https://doi.org/10.1109/VNC.2011.6117132>
- Kenney, J. B. Controlling Channel Congestion using CAM Message Generation Rate. (2013).
- Egea-Lopez, E. & Pavon-Mariño, P. Distributed and fair beaconing rate adaptation for congestion control in vehicular networks. *IEEE Trans. Mob. Comput.* **15**, 3028–3041 (2016).
- Kelly, F. Charging and Rate Control for Elastic Traffic. *Eur. Trans. Telecommun.* **8**, 33–37 (1997).
- Mo, J. & Walrand, J. Fair end-to-end window-based congestion control. *IEEE/ACM Trans. Netw.* **8**, 556–567 (2000).
- Son, S. & Park, K.-J. BEAT: Beacon inter-reception time ensured adaptive transmission for vehicle-to-vehicle safety communication. *Sensors* **19**, 3061 (2019).
- Nguyen, H. H. & Jeong, H. Y. Mobility-adaptive beacon broadcast for vehicular cooperative safety-critical applications. *IEEE Trans. Intell. Transp. Syst.* **19**, 1996–2010 (2018).
- SAE. J2945/1A: On-Board System Requirements for V2V Safety Communications - SAE International. Available at: https://www.sae.org/standards/content/j2945/1_202004/. (Accessed: 7th July 2021)
- Bansal, G., Lu, H., Kenney, J. B. & Poellabauer, C. EMBARC: Error model based adaptive rate control for vehicle-to-vehicle communications. *VANET 2013 - Proc. 10th ACM Int. Work. Veh. Inter-NETworking, Syst. Appl.* 41–50 (2013). doi:<https://doi.org/10.1145/2482967.2482972>
- Fallah, Y. P., Huang, C. L., Sengupta, R. & Krishnan, H. Analysis of information dissemination in vehicular ad-hoc networks with application to cooperative vehicle safety systems. *IEEE Trans. Veh. Technol.* **60**, 233–247 (2011).
- Joerer, S. *et al.* Enabling situation awareness at intersections for IVC congestion control mechanisms. *IEEE Trans. Mob. Comput.* **15**, 1674–1685 (2016).
- Lyu, F. *et al.* ABC: Adaptive beacon control for rear-end collision avoidance in VANETs. *2018 15th Annu. IEEE Int. Conf. Sensing, Commun. Networking, SECON 2018 1–9* (2018). doi:<https://doi.org/10.1109/SAHCN.2018.8397130>
- Hajjaghajani, F. & Qiao, C. Tailgating risk-aware beacon rate adaptation for distributed congestion control in VANETs. *2019 IEEE Glob. Commun. Conf. GLOBECOM 2019 - Proc.* (2019). doi:<https://doi.org/10.1109/GLOBECOM38437.2019.9013608>

24. Aznar-Poveda, J., Egea-Lopez, E., Garcia-Sanchez, A.-J. & Pavon-Marino, P. Time-to-collision-based awareness and congestion control for vehicular communications. *IEEE Access* **7**, 2 (2019).
25. Sospeter, J., Wu, D., Hussain, S. & Tesfa, T. An Effective and Efficient Adaptive Probability Data Dissemination Protocol in VANET. *Data* **2019**, Vol. 4, Page 1 4, 1 (2018).
26. Li, W., Song, W., Lu, Q. & Yue, C. Reliable congestion control mechanism for safety applications in urban VANETs. *Ad Hoc Netw.* **98**, 2 (2020).
27. Bolufé, S. *et al.* Dynamic control of beacon transmission rate and power with position error constraint in cooperative vehicular networks. *Proc. ACM Symp. Appl. Comput.* <https://doi.org/10.1145/3167132.3167356> (2018).
28. Goudarzi, F. & Asgari, H. Non-cooperative beacon rate and awareness control for VANETs. *IEEE Access* **5**, 16858–16870 (2017).
29. Sutton, R. S. & Barto, A. G. Reinforcement Learning: An Introduction Second edition, in progress.
30. Ghadimi, E., Calabrese, F. D., Peters, G. & Soldati, P. A Reinforcement Learning Approach to Power Control and Rate Adaptation in Cellular Networks. *IEEE Int. Conf. Commun.* (2016).
31. Ye, H., Li, G. Y. & Juang, B. H. F. Deep reinforcement learning based resource allocation for V2V communications. *IEEE Trans. Veh. Technol.* **68**, 3163–3173 (2019).
32. Sharma, S. & Singh, B. Context aware autonomous resource selection and Q-learning based power control strategy for enhanced cooperative awareness in LTE-V2V communication. *Wirel. Netw.* **26**, 4045–4060 (2020).
33. Nguyen, K. K., Duong, T. Q., Vien, N. A., Le-Khac, N. A. & Nguyen, M. N. Non-cooperative energy efficient power allocation game in D2D communication: A multi-agent deep reinforcement learning approach. *IEEE Access* **7**, 100480–100490 (2019).
34. He, C., Hu, Y., Chen, Y. & Zeng, B. Joint power allocation and channel assignment for NOMA with deep reinforcement learning. *IEEE J. Sel. Areas Commun.* **37**, 2200–2210 (2019).
35. Aznar-Poveda, J., Garcia-Sanchez, A. J., Egea-Lopez, E. & Garcia-Haro, J. MDPRL: A Q-learning approach for the joint control of beaconing rate and transmission power in VANETs. *IEEE Access* **9**, 10166–10178 (2021).
36. Cho, B. M., Jang, M. S. & Park, K. J. Channel-aware congestion control in vehicular cyber-physical systems. *IEEE Access* **8**, 73193–73203 (2020).
37. Huang, C. L., Fallah, Y. P., Sengupta, R. & Krishnan, H. Adaptive intervehicle communication control for cooperative safety systems. *IEEE Netw.* **24**, 6–13 (2010).
38. Subramanian, S. *et al.* Congestion control for vehicular safety: Synchronous and asynchronous MAC algorithms. *VANET'12 - Proc. 9th ACM Int. Work. Veh. Inter-NETworking, Syst. Appl.* 63–72 (2012). doi:<https://doi.org/10.1145/2307888.2307900>
39. Harris, C. R. *et al.* Array programming with NumPy. *Nat.* **2020** 5857825 **585**, 357–362 (2020).
40. Malik, S. & Sahu, P. K. A comparative study on routing protocols for VANETs. *Heliyon* **5**, e02340 (2019).
41. Choudhury, B., Shah, V. K., Dayal, A. & Reed, J. H. Joint Age of Information and Self Risk Assessment for Safer 802.11p based V2V Networks. (2020).
42. Varga, A. OMNeT++. *Model. Tools Netw. Simul.* 35–59 (2010). doi:https://doi.org/10.1007/978-3-642-12331-3_3
43. INET Framework - INET Framework. Available at: <https://inet.omnetpp.org/>. (Accessed: 16th July 2021)
44. Lopez, P. A. *et al.* Microscopic Traffic Simulation using SUMO. *IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC 2018-November*, 2575–2582 (2018).

Acknowledgements

This research has been supported by the projects AIM, ref. TEC2016-76465-C2-1-R, ARISE2 “Future IoT Networks and Nano-networks (FINE)” ref. PID2020-116329GB-C22, ONOFRE-3, ref. PID2020-112675RB-C41 [Agencia Estatal de Investigación (AEI), European Regional Development Fund (FEADER), European Union (EU)], ATENTO, ref. 20889/PI/18 (Fundación Séneca, Región de Murcia), and LIFE [Fondo SUPERA Covid-19, funded by Agencia Estatal Consejo Superior de Investigaciones Científicas (CSIC), Universidades Españolas and Banco Santander]. J.A.P. thanks the Spanish MECED for an FPI grant ref. BES-2017-081061. Finally, the authors acknowledge Laura Wettersten for her contribution in reviewing the grammar and spell of the manuscript.

Author contributions

Conceptualization, J.A.P.; methodology, J.A.P., A.J.G.S., E.E.L.; software, J.A.P. and E.E.L.; validation, J.A.P. and E.E.L.; formal analysis, J.A.P., A.J.G.S., E.E.L.; investigation, J.A.P., A.J.G.S., E.E.L.; writing—original draft preparation, J.A.P.; writing—review and editing, J.A.P., A.J.G.S., E.E.L., and J.G.H.; visualization, J.A.P.; supervision, A.J.G.S., E.E.L., and J.G.H.; project administration, A.J.G.S., E.E.L., and J.G.H.; funding acquisition, A.J.G.S., E.E.L., and J.G.H. All authors have read and agreed to the published version of the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to J.A.-P.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2022